THÈSE

en vue de l'obtention du grade de Docteur, délivré par

l'ÉCOLE NORMALE SUPÉRIEURE DE LYON

École Doctorale N° 512

École Doctorale en informatique et Mathématiques de Lyon

Discipline : Informatique

Soutenue publiquement le 23/09/2024, par :

Rémi DI GUARDIA

Identity of Proofs and Formulas using Proof-Nets in Multiplicative-Additive Linear Logic

Identité des Preuves et Formules par l'usage des Réseaux de Preuve en Logique Linéaire Multiplicative-Additive

Devant le jury composé de :

Claudia	FAGGIAN	Chargée de Recherche	CNRS & Université Paris Cité	Examinatrice
Willem	HEIJLTJES	Senior Lecturer	University of Bath	Rapporteur
Olivier	LAURENT	Directeur de Recherche	CNRS & ÉNS de Lyon	Directeur
Damiano	MAZZA	Directeur de Recherche	CNRS & Université Paris-Nord	Rapporteur
Samuel	MIMRAM	Professeur des Universités	École Polytechnique	Examinateur

Abstract

This study is concerned with the equality of proofs and formulas in linear logic, with in particular contributions for the multiplicative-additive fragment of this logic. In linear logic, and as in many other logics (such as intuitionistic logic), there are two transformations on proofs: cut-elimination and axiom-expansion. One often wishes to identify two proofs related by these transformations, as it is the case semantically (in a categorical model for instance). This situation is similar to the one in the λ-calculus where terms are identified up to β-reduction and η-expansion, operations that, through the prism of the Curry-Howard correspondence, are related respectively to cut-elimination and axiom-expansion. It is widely conjectured by the community that this identification corresponds exactly to identifying proofs up to rule commutation, a third well-known operation on proofs which is easier to manipulate. This result is even implicitly assumed by many papers, in particular those about the proof-net syntax we will discuss just after. Nonetheless, this important property was still not proved at the beginning of this thesis. We demonstrate it for multiplicative-additive linear logic, following a method that should extend to full linear logic.

Not only proofs but also formulas can be identified up to cut-elimination and axiom-expansion. Two formulas are isomorphic if there are proofs between them whose compositions yield identities, still up to cut-elimination and axiom-expansion. These formulas are then really considered to be the same, and every use of one can be replaced with one use of the other. It is expected that all isomorphisms arise from a few well-known given ones, and such a result was proven in multiplicative linear logic by Balat and Di Cosmo: they gave an equational theory characterizing exactly isomorphic formulas in this part of the logic. We extend their result to the wider multiplicative-additive linear logic, whose characterizing equational theory is more involved. A generalization of an isomorphism is a retraction, which intuitively corresponds to a pair of formulas where the first can be replaced by the second – but not necessarily the other way around, contrary to an isomorphism. Studying retractions is more complicated and, as opposed to isomorphisms, there is no conjecture on what retractions are expected to be. We characterize retractions to an atom in the multiplicative fragment of linear logic.

When studying the two previous problems, the usual syntax of proofs from sequent calculus seems ill-suited because we consider proofs up to rule commutation. Part of linear logic can be expressed in a better adapted syntax in this case: proof-nets, which are graphs representing proofs quotiented by rule commutation. This syntax was an instrumental tool for the characterization of isomorphisms and retractions. Unfortunately, proof-nets are not (or badly) defined with units: there is no known adequate definition if one wants the canonical property to represent proofs up to all rule commutations, even if there exist some notions corresponding to quotienting by unit-free rule commutations. Concerning our issues, this restriction leads to a study of the unit-free case by means of proof-nets with the crux of the demonstration, preceded by a work in sequent calculus to handle the units.

Besides, this thesis also develops part of the theory of proof-nets by providing a simple proof of the sequentialization theorem, which relates the two syntaxes of proof-net and sequent calculus, substantiating that they describe the same underlying objects. This new demonstration is obtained as a corollary of a generalization of Yeo's theorem. This last result is fully expressed in the theory of edge-colored graphs, and allows to recover proofs of sequentialization for various definitions of proof-nets. Finally, we also formalized proof-nets for the multiplicative fragment of linear logic in the proof assistant Coq, with notably an implementation of our new sequentialization proof.

Keywords: Multiplicative-additive linear logic, Cut elimination, Rule commutation, Type Isomorphisms, Retraction, Sequent calculus, Proof-nets, Sequentialization, Edge-colored graph theory, Formalization, Coq.

Résumé

Cette thèse s'intéresse à l'égalité des preuves et des formules en logique linéaire, avec des contributions en particulier dans le fragment multiplicatif-additif de cette logique. En logique linéaire, et dans de nombreuses autres logiques (telle que la logique intuitionniste), on dispose de deux transformations sur les preuves : l'élimination des coupures et l'expansion des axiomes. On souhaite très souvent identifier deux preuves reliées par ces transformations, étant donné qu'elles le sont sémantiquement (dans un modèle catégorique par exemple). Cette situation est similaire à celle du λ-calcul où les termes sont identifiés à β-réduction et η-expansion près, opérations qui, par le prisme de la correspondance de Curry-Howard, se rapportent respectivement à l'élimination des coupures et à l'expansion des axiomes. Une conjecture largement acceptée dans la communauté est que cette identification des preuves correspond exactement à l'identification des preuves à commutation de règle près, qui est une troisième opération sur les preuves bien connue et plus facile à manipuler. Ce résultat est même supposé implicitement par de nombreux articles, en particulier ceux traitant de la syntaxe des réseaux de preuve à laquelle nous nous intéresserons juste après. Néanmoins, cette importante propriété n'était toujours pas prouvée au commencement de cette thèse. Nous la démontrons ici pour la logique linéaire multiplicative-additive, suivant une méthode qui devrait permettre d'étendre la preuve à la logique linéaire dans son entièreté.

Non seulement des preuves peuvent être identifiées à élimination des coupures et expansion des axiomes près, mais aussi des formules. Deux formules sont isomorphes si elles sont reliées par des preuves dont les compositions donnent l'identité, toujours à élimination des coupures et expansion des axiomes près. Ces formules sont alors réellement considérées comme les mêmes, et toute utilisation de l'une peut être remplacée par une utilisation de l'autre. Il est attendu que tous les isomorphismes peuvent être obtenus à partir de quelques isomorphismes bien connus, ce qui a d'ailleurs été prouvé par Balat et Di Cosmo pour la logique linéaire multiplicative : ils ont donné une théorie équationnelle caractérisant exactement les formules isomorphes dans cette partie de la logique. Nous étendons leur résultat dans le cadre plus large de la logique linéaire multiplicative-additive, dont la théorie équationnelle est plus complexe. Un problème généralisant les isomorphismes est celui des rétractions, qui intuitivement correspondent aux paires de formules où la première peut être remplacée par la seconde – mais pas nécessairement la seconde par la première, contrairement aux isomorphismes. Étudier les rétractions est bien plus complexe et, contrairement aux isomorphismes, il n'y a pas de conjecture donnant les paires de formules qui devraient être des rétractions. Nous avons caractérisé les rétractions des atomes dans le fragment multiplicatif de la logique linéaire.

Pour l'étude des deux problèmes précédents, la syntaxe usuelle des preuves du calcul des séquents semble mal adaptée, car on considère des preuves à commutation de règle prés. Une partie de la logique linéaire possède une meilleure syntaxe dans ce cas : les réseaux de preuve, qui sont des graphes représentant des preuves quotientées par les commutations de règle. Cette syntaxe fut un outil indispensable pour caractériser isomorphismes et rétractions. Malheureusement, les réseaux de preuve ne sont pas (ou mal) définis en présence des unités : il n'y a pas de définition adéquate pour des réseaux canoniques représentant les preuves à toutes les commutations de règle près, même si il existe des notions correspondant à un quotient par les commutations de règles sans unité. Pour nos problèmes, cette restriction conduit à une étude du cas sans unité dans les réseaux avec le cœur de la démonstration, précédée d'un travail en calcul des séquents pour prendre en compte les unités.

Cette thèse développe par ailleurs une partie de la théorie des réseaux de preuve en fournissant une preuve simple du théorème de séquentialisation, qui relie les deux syntaxes des réseaux de preuve et du calcul des séquents, justifiant qu'elles décrivent les mêmes objets sous-jacents. Cette nouvelle démonstration s'obtient comme corollaire d'une généralisation du théorème de Yeo. Ce dernier résultat s'exprime entièrement dans la théorie des graphes aux arêtes colorées, et permet de déduire des preuves de sequentialisation pour différentes définitions de réseaux de preuve. Enfin, nous avons aussi formalisé les réseaux de preuve du fragment multiplicatif de la logique linéaire dans l'assistant de preuve Coq, avec en particulier une implémentation de notre nouvelle preuve de séquentialisation.

Mot-clefs : Logique linéaire multiplicative-additive, Élimination des coupures, Commutation de règle, Isomorphisme de type, Rétraction, Calcul des séquents, Réseaux de preuve, Séquentialisation, Théorie des graphes aux arêtes colorées, Formalisation, Coq.

Remerciements

Merci en premier lieu à Olivier Laurent, mon directeur de thèse, pour m'avoir non seulement pris comme étudiant mais aussi pour m'avoir fait découvrir le domaine où j'ai effectué ma thèse. Merci de plus pour ton attention tout au long de ces trois années, tes réponses à mes nombreuses questions et les propres questions que tu as soulevées et auxquelles je n'aurais pas pensé. Merci finalement pour ta relecture attentive de ce manuscrit, et tes recommandations.

Je voudrais aussi remercier Claudia Faggian, Willem Heijltjes, Damiano Mazza et Samuel Mimram pour avoir accepté de participer à mon jury de thèse. Je suis particulièrement reconnaissant à Willem et Damiano pour avoir lu et rapporté cette thèse dont je m'excuserais de la longueur si je n'avais eu autant à dire ni des preuves parfois aussi complexes. Merci encore à Willem pour les nombreuses corrections qui ont découlé de cette relecture, et qui devraient rendre la lecture de cette thèse plus digeste. J'en profite pour aussi remercier Samuel une deuxième fois, cette fois pour sa participation à mes comités de suivi durant mes trois années de thèse, aux côtés de Bora Uçar.

Merci à Lionel Vaux Auclair et Lorenzo Tortora de Falco pour avoir co-écrit avec moi et Olivier un papier sur ce qui ne devait être qu'une simple extension d'une preuve de séquentialisation de MLL avec mix à MALL, et qui a fini en discussions architecturales sur comment appeler une structure avec deux arches et une pile autrement qu'un pont, en passant par des considérations sans fin sur les boucles.

Merci aussi aux membres de Plume pour la très agréable ambiance de travail durant ma thèse, les différentes discussions que nous avons eues, les exposés des GdTs et les Plume Beers. Je remercie notamment Tito (Lê Thành Dũng Nguyễn) pour son impact direct sur l'importance de la partie graphe de cette thèse, qui n'aurait pas été du tout pareil sans ses connaissances précises de la littérature dans des domaines variés. Similairement, merci à Damien Pous et Cyril Cohen pour leur aide en Coq qui m'a fait gagner un temps considérable grâce à leur expérience et leur connaissance de librairies que j'utilise et qu'ils ont développées. J'aimerais aussi remercier Marie Narducci pour sa disponibilité et son aide constante pour l'administratif, y compris lors de la découverte de Nautilus.

Je remercie de plus Guy McCusker, Willem Heijltjes et James Laird pour m'avoir chaleureusement accueilli à Bath, ainsi que pour les discussions que nous y avons eues. J'aimerais remercier les nombreuses personnes avec qui j'ai discuté durant de multiples Chocolas, à l'occasion de conférences ou au CIRM, mais ne citerai pas de nom car j'en oublierais toujours trop.

Je remercie aussi mes amis de lycée de Rennes, pour nos chaleureuses réunions de parties bis-annuelles, bien qu'on soit maintenant bien dispersés en France. Merci aussi à toutes celles et ceux que j'ai croisés durant les clubs jeux de société de Centrale puis de l'ENS, qui ont bien égayé mes semaines. Je remercie tout particulièrement le club de cartes Baston de l'ENS, pour les belles campagnes qu'on y a jouées, et notamment Jean pour m'avoir initié à ce jeu qu'il a inventé.

Enfin, merci à toute ma famille, à qui je n'ai malheureusement jamais réussi à expliquer correctement mon sujet de thèse. Merci tout particulièrement à mon frère Hugo pour avoir lu jusqu'au bout cette thèse très éloignée de ses lectures habituelles !

Contents

Introduction

This thesis is encompassed within the field of *proof theory*, namely the study of mathematical proofs as formal objects. This sub-field of logic allows to analyze proofs as any other mathematical objects, permitting for instance to show that a statement is not provable. There are many logics in proof theory, the most famous ones surely being classical and intuitionistic logics. Here is considered another well-known one: *linear logic* [Gir87]. This logic is a refinement of classical logic, and is as powerful as the latter but with a view of formulas as resources that can be consumed, or used, exactly once – hence its denomination. Such a viewpoint is useful in plenty of settings, from quantitative considerations to describe objects with measures instead of only yes/no questions, to the analysis of programs through the Curry-Howard correspondence. The latter shows links between logic and programming: formulas correspond to types while proofs are like programs, or functions. The view of formulas as resources also corresponds more to what happens with real systems: when buying butter, one does not expect to get the butter while keeping the money, which is kind of what happens in classical logic.

As many other logics, linear logic can be described under the form of a *sequent calculus*: this proof system is the data of *formulas*, that compose statements we call *sequents*, and of *inference rules* that permit to deduce the veracity of a sequent from the veracity of some others – possibly none in the case of an axiom. This way, a proof is a tree with the statement one wants to prove at the bottom, above which are rules implying that this statement is true if others are, all the way up to basic facts that are trivially true. This inductive definition has the main advantage to be easily manipulated, and can therefore be implemented in a computer, yielding *proof assistants* such as Coq [Coq] where one can code a proof that is computer-checked. Notwithstanding this great quality, sequent calculus has one major drawback: identity of proofs is complicated.

Indeed, one does not want to identify only proofs that are syntactically equal, *i.e.* whose successive rules are exactly the same. In linear logic, and in most sequent calculi, there is a *cut-rule* that allows computation, called *cut-elimination*, and one wishes for two proofs related by such a computation to be the same. Looking at programs, even if $2 + 2$ and $(1 + 2) + 1$ are not the same term, because one has two symbols $+$ while the other has only one, it is still very sensible to identify them, at least from a computational approach. Another notion of identity on proofs is *axiom-expansion*, corresponding to the fact that a function f should be considered the same as $x \longmapsto f(x)$, the function that takes an input x and returns $f(x)$, which is again a very natural identification to ask for. Hence, one wants to see proofs as equivalent up to these two relations, which in general is either a trivial nonsense or a combinatorial nightmare. For instance of the former, in classical logic any two proofs of a given statement are related computationally [GLT89, Appendix B.1], which is as a matter of fact an easy characterization. Unfortunately, this means when seeing proofs as programs that there is only one element of each type, so one integer or one boolean, which is usually not desirable. Considering the combinatorial nightmare, one simply has to look at the numerous tables at the end of Chapter 1, which contain what is conjectured as the core identification needed to check if two proofs should be identified – in other words the "easy" way to get all proofs equivalent to the one under consideration. More formally, cut-elimination is *Church-Rosser* modulo rule commutation. Henceforth, sequent calculus appears to be pretty inefficient for identifying proofs.

Fortunately, linear logic has since its creation another syntax: *proof-nets* [Gir87]. A proof-net is a representation of a proof not under the form previously described of sequent calculus, but as a *graph*, *i.e.* a collection of vertices with arcs linking couples of vertices together [Die05; JU76]. This unusual presentation is well-behaved for the identification of proofs: two proofs are computationally equal if and only if they share the same graph, the same proof-net [HG16]. This means proof-nets are the *canonical* syntax for proof-theory: we have only one object for what we want to identify, instead of plenty of proofs related through some relations; like having only 4 instead of $2 + 2$ and $(1 + 2) + 1$. Such a syntax solves our problem of identification of proofs in one go, and is wonderful where it works. Unfortunately, a proof-net syntax does not exist for the whole of linear logic, which thus has to bear with sequent calculus. A part of linear logic admitting proof-nets is *multiplicative-additive linear logic* [HG05] (almost: only its units break proof-nets), which is the principal part of linear logic this thesis will be concerned with. Still, we will introduce the full logic to explain what fails outside of this sub-system, or to conjecture extensions of our results to the full logic. A main result in the theory of proof-nets is the *sequentialization theorem*: it indicates that proof-nets correspond exactly to proofs. Another major result is that two (normal) proofs correspond to a same proof-net if and only if they are identified, *i.e.* are related by cut-elimination. As an aside, proof-nets are far from the only tool to study the identity of proofs. Such a problem is studied at length in deep inference, see for instance [Brü04], as well as focusing [And92], coherent spaces [Gir87] and game semantics with full abstraction (*e.g.* [AJM00]).

Lastly, one may not only identify proofs related by computation, but also formulas that behave the same. As an example, one intuitively sees that the formula $A \wedge B$ – "A and B" – should behave in most ways as the formula $B \wedge A$ – "B and A". Another well-known example in programming is Currying: types $A \to (B \to C)$ and $(A \wedge B) \to C$ are identified in functional programming, with an easy way to turn a function of the first type into one of the second type, and vice-versa. The notion formalizing this intuition is the one of *isomorphism* that identifies formulas that cannot be distinguished computationally. Two formulas A and B are isomorphic when there are a proof of B when supposing A, and one of A when supposing B, whose compositions are identities up to computation. In layman's terms, one can encode an object of A as an object of B, and decode it back as an object of A without loss, and vice-versa. A related notion is the one of *retraction*, when this lossless encoding holds in only one direction. This corresponds to a natural form of sub-typing in programming.

Outline of the thesis This thesis is divided into three parts: Part I is about sequent calculus, Part II considers proof-nets, and Part III studies isomorphisms and retractions.

The first part is composed of two chapters. Chapter 1 defines the sequent calculus of linear logic, as well as the concepts explained beforehand: the computations on proofs that are cut-elimination and axiom-expansion, as well as *rule commutation*, a related concept to identify proofs equal up to these relations in sequent calculus. This chapter sets the framework of the thesis, with nothing novel there excepted for a clear definition of these three relations in full linear logic. Chapter 2 scrutinizes the properties of these transformations, and proves that rule commutations identify proofs up to axiom-expansion and cut-elimination, *i.e.* is the "easy" way to see if two proofs are in fact one and the same. This property is hardly a surprise for the reader familiar with linear logic, but is a blind spot in the literature that had not been proved previously. Despite that such a result is in the end nothing more than a "basic property" of the sequent calculus, its proof is complex and requires a precise analysis of the rules.

We then move on to the study of proof-nets in the second part, starting with Chapter 3 on graph theory and edge-coloring of graphs. It not only gives the background definitions needed to handle proof-nets, but also a proof of a new generalization of *Yeo's theorem* [Yeo97]. This result will be of use in Chapter 4 that defines proof-nets and proves they correspond to proofs of sequent calculus. This last demonstration is at its heart a corollary of our version of Yeo's theorem. We conclude the second part with Chapter 5 on a formalization of proof-nets in the proof assistant *Coq*, translating on computer our main results of the previous chapter in the simple case of multiplicative linear logic instead of multiplicative-additive linear logic.

Then comes the third and last part with Chapter 6 on isomorphisms. There is given an *equational theory*, *i.e.* building blocks, characterizing exactly isomorphisms of multiplicative-additive linear logic. This implies a corresponding result in category theory, with a characterization of $\star$-autonomous categories with finite products, which is the class of categories corresponding to multiplicative-additive linear logic. Lastly, we consider retractions of multiplicative linear logic in Chapter 7, with a characterization of all retractions to an atom, among other results for the general case such as proving the *Cantor-Bernstein-Schröeder* property, or that units do not give more retractions than their unitality isomorphisms. Both these chapters use results from the previous parts, and proof-nets are a key ingredient to get the characterizations of isomorphisms and retractions.

Related publications This thesis arises from results published during the corresponding PhD, that it subsumes and extends.

- Properties of cut-elimination as a rewriting system, in relation with rule commutation, were first given for multiplicative-additive linear logic in [DL23] (a journal paper extending this conference paper has currently been submitted). Chapter 2 simplifies part of the proofs, and also consider the *mix*-rules.

  Chapter 6 extends the main contribution of [DL23], expanding the characterization of isomorphisms in presence of the mix_2-rule and with different intermediate lemmas that are reused for the study of retractions in Chapter 7.

- The main contributions in Chapters 3 and 4 mostly broaden [Di+23] to incorporate graph results needed for sequentialization of additives. It also presents the main result in the more general framework of graph theory. A journal paper with the same authors is in preparation with these results, as well as more details in the simpler case of multiplicative proof-nets, where this yields to a very simple and short proof of the sequentialization theorem. This proof of sequentialization in multiplicative linear logic is also a simpler version of some known ones, including a previous paper of the author [DL22]. A conference paper about this new sequentialization proof for multiplicative linear logic has been submitted.

Part I

Sequent Calculus

This part studies the Sequent Calculus of Linear Logic. The framework of sequent calculus is often the one used to define logics, and the first definition of linear logic by Girard in [Gir87] was no exception. We give in Chapter 1 the definition of linear logic in sequent calculus, as well as those of the usual proof transformations: axiom-expansion, cut-elimination and rule commutation. In a second time, we study properties of this calculus in terms of normalization and confluence. We prove in Chapter 2 that, while the strongest properties we could wish for – strong normalization and confluence – do not hold for cut-elimination, we still have weaker corresponding properties – weak normalization and strong normalization of part of the system, as well as Church-Rosser modulo rule commutation. These results are proved in the multiplicative-additive fragment of linear logic, and conjectured for the full proof system.

Chapter 1

Sequent Calculus of Linear Logic

In this first chapter, we introduce linear logic with its sequent calculus syntax. The first definition of this logic can be found in [Gir87]. Linear logic plays a key part in the framework of the Curry-Howard correspondence, which links proofs and programs, and gave rise to many concepts thanks to its view of formulas as *resources*, that must be used exactly once in a proof. This is a very rich logic that can be seen as a refinement of classical logic, allowing to better understand the connectives and units of the latter: the conjunction (logical "and"), the disjunction (logical "or"), true and false. Each connective and unit has two versions in this logic: a multiplicative version and an additive one. Moreover, two modalities are introduced to speak of unlimited resources: these are the exponential connectives, which correspond to the structural rules, usually not associated to any connective. This yields a fairly complex system with ten connectives and units, against the four of classical logic.

While most of our work in the next chapters focuses only on a sub-system of it, we give here the full logic so as to explain when possible which generalizations of our results are conjectured, or on the contrary which rule breaks such extensions. Furthermore, considering the full logic is also an opportunity to describe precisely some concepts which are folklore but scarcely defined precisely in the general case – typically rule commutation.

Outline The sequent calculus of linear logic is described with its formulas and rules, as well as sub-systems of this logic (Section 1.1). Then are given some well-known elementary concepts of this logic (size of a formula, slice, ...) (Section 1.2). Finally, the three transformations on proofs that will be of interest in this thesis are defined: axiom-expansion, cut-elimination and rule commutation (Section 1.3).

1.1 Sequent Calculus

We define here the sequent calculus of linear logic [Gir87]. As written before, this logic contains three classes of propositional connectives: multiplicative, additive and exponential ones. The multiplicative and additive families provide two copies of each classical propositional connective: two copies of conjunction ($\otimes$ and $\&$), of disjunction ($\parr$ and $\oplus$), of true (1 and $\top$) and of false ($\bot$ and 0). The exponential family is constituted of two modalities $!$ and $?$, and is used to deal with the structural rules, and in particular to indicate formulas that can be duplicated or erased. We study a general setting with also (second order) quantifiers and possibly some additional rules: the mix_2- and mix_0-rules – often appearing when considering proof-nets (see for instance [FR94; Ham04; Ngu20], or the usual proof-nets for multiplicative-exponential linear logic [LL22]) – and the $\cup$ and $\emptyset$-rules – used in differential linear logic under the respective names $+$ and 0 [Ehr18], and the latter rule under the name ✠ in ludics [Gir01].

1.1.1 Formulas and Rules

Assume given a countable set $\mathcal{X}$ of **atoms**.[1] Formulas are given by the following grammar, where X is an atom:

$$\begin{aligned}
A, B ::= &\mid X^+ \mid X^- && \text{(atom)}\\
&\mid A \parr B \mid A \otimes B \mid \bot \mid 1 && \text{(multiplicative)}\\
&\mid A \mathbin{\&} B \mid A \oplus B \mid \top \mid 0 && \text{(additive)}\\
&\mid ?A \mid !A && \text{(exponential)}\\
&\mid \forall X A \mid \exists X A && \text{(quantifier)}
\end{aligned}$$

We call **positive atoms** those formulas of the shape X^+, **negative atoms** those of the kind X^-, and **signed atoms** all positive or negative atoms. Some $X \in \mathcal{X}$ is called an **unsigned atom** to remove any ambiguity. The **connectives** are the $\otimes$, $\parr$, $\oplus$, $\&$, $!$, $?$, $\forall$ and $\exists$ symbols, the four first being binary and the others unary connectives. Meanwhile, the **units** are 1, $\bot$, 0 and $\top$. The **leaves** consist of the signed atoms and the units.

We define on formulas a function $(\cdot)^\perp$ called **orthogonality**, also named negation or duality, through the following inductive definition:

$$\begin{aligned}
(X^+)^\perp &= X^- & (X^-)^\perp &= X^+\\
(A \parr B)^\perp &= B^\perp \otimes A^\perp & (A \otimes B)^\perp &= B^\perp \parr A^\perp\\
\bot^\perp &= 1 & 1^\perp &= \bot\\
(A \mathbin{\&} B)^\perp &= B^\perp \oplus A^\perp & (A \oplus B)^\perp &= B^\perp \mathbin{\&} A^\perp\\
\top^\perp &= 0 & 0^\perp &= \top\\
(?A)^\perp &= !A^\perp & (!A)^\perp &= ?A^\perp\\
(\forall X A)^\perp &= \exists X A^\perp & (\exists X A)^\perp &= \forall X A^\perp
\end{aligned}$$

It can be easily checked that orthogonality is an involution: $A^{\perp\perp} = A$. One may recognize the *non-commutative* De Morgan's laws in the definition of the dual of a binary connective. The non-commutative version is the good notion, as shown in the context of cyclic linear logic where this leads to planar proof-nets [AM98]. When speaking about proof-nets, this choice will often result in planar graphs on our illustrations, with links not crossing each others; hence to more readable figures. Everything written in this thesis still holds when taking the commutative De Morgan's laws instead, up to some minor fixes when considering dual formulas.

Another often used connective in linear logic is the **linear implication**, denoted by $\multimap$. It can be defined as $A \multimap B = A^\perp \parr B$. This corresponds to the definition of implication in classical logic, with $A \implies B = \neg A \vee B$.

[1] One often assume $\mathcal{X}$ to be countable, for we need an unbounded number of variables, but can only use a finite subset (of arbitrary size). This allows us to consider fresh atoms at any point.

Sequents are sets of occurrences of formulas written in the form $\vdash A_1, \ldots, A_n$. This means that, as is usual, we consider only *one-sided* sequents. Everything in this thesis should expand trivially to *two-sided* sequents; however, the two-sided framework has twice the number of rules, and thus has even more cases in our proofs when we look carefully at each rule.

Sequent calculus rules of linear logic are given on Figure 1.1. In these rules, A and B stand for arbitrary formulas, Γ and Δ for contexts (*i.e.* sets of occurrences of formulas). The notation $?\Gamma$ means that each formula of this context is a ?-formula, *i.e.* $?\Gamma = ?A_1, \ldots, ?A_n$. The notation $[B/X]$ is the usual substitution of an atom X by a formula B: each occurrence of X^+ is replaced by B, and each occurrence of X^- by $B^\perp$ (see Section 1.2 for more details). The side condition in the application of the $\forall$-rule, X not free (or X fresh) in Γ, is the usual one when handling universal quantifiers: the atom X should not appear in formulas of Γ, whether as a positive or negative atom, except possibly below a quantifier on X, namely $\forall X$ or $\exists X$. This condition can always be satisfied up to α-conversion.

Remark there is no rule associated to 0, there are two associated with $\oplus$, $\oplus_1$ and $\oplus_2$, and three associated with ?, the dereliction $?d$, the contraction $?c$ and the weakening $?w$. Otherwise, there is exactly one rule for each other unit and connective. As the $\oplus_1$ and $\oplus_2$-rules are really similar, we will often talk about a $\oplus_i$-rule, which is $\oplus_1$ or $\oplus_2$ according to the value of $i \in \{1;2\}$. A well-informed reader may remark the absence of an exchange rule; some explanations on this subject are given in Section 1.1.3.

1.1.2 Sub-systems

One often does not want to consider the full sequent calculus defined previously, but only a part of it; moreover, each of the optional rules may or may not be added to it. We call this a **sub-system** of the logic. As one always want to keep the atoms as well as the *ax*- and *cut*-rules, one also has to keep dual connectives and units, so that orthogonality $(\cdot)^\perp$ stays well-defined in this restriction. Here are the sub-systems we will use in this thesis. They define not only sub-sets of rules, but also of formulas.

Multiplicative-additive linear logic, denoted by MALL, is made of the multiplicative and additive formulas and rules of linear logic. In more details, its formulas are given by the following grammar:

$$A, B ::= X^+ \mid X^- \mid A \otimes B \mid A \parr B \mid 1 \mid \perp \mid A \,\&\, B \mid A \oplus B \mid \top \mid 0$$

Its rules are ax, cut, $\parr$, $\otimes$, $\perp$, 1, $\&$, $\oplus_1$, $\oplus_2$, $\top$ and 0. Similarly, multiplicative linear logic, called MLL, has for formulas:

$$A, B ::= X^+ \mid X^- \mid A \otimes B \mid A \parr B \mid 1 \mid \perp$$

The rules of this deduction system are: ax, cut, $\parr$, $\otimes$, $\perp$ and 1. As other sub-systems, one can define in a similar fashion additive linear logic ALL, exponential linear logic ELL, or multiplicative-exponential linear logic MELL. Considering the full propositional linear logic, with all rules excepted for the quantifiers and optional rules, we simply denote it by LL.

By a **unit-free** sub-system, we mean this sub-system without the units and their associated rules. For instance, MALL_{uf} denotes unit-free multiplicative-additive linear logic, with as formulas:

$$A, B ::= X^+ \mid X^- \mid A \otimes B \mid A \parr B \mid A \,\&\, B \mid A \oplus B$$

Its rules are ax, cut, $\parr$, $\otimes$, $\&$, $\oplus_1$ and $\oplus_2$. Likewise, one defines MLL_{uf} as MLL without the units and so on.

Identity rules:

$$\frac{}{\vdash A^{\perp}, A}\,(ax) \qquad \frac{\vdash A^{\perp}, \Gamma \quad \vdash A, \Delta}{\vdash \Gamma, \Delta}\,(cut)$$

Multiplicative rules:

$$\frac{\vdash A, B, \Gamma}{\vdash A \parr B, \Gamma}\,(\parr) \qquad \frac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \qquad \frac{\vdash \Gamma}{\vdash \bot, \Gamma}\,(\bot) \qquad \frac{}{\vdash 1}\,(1)$$

Additive rules:

$$\frac{\vdash A, \Gamma \quad \vdash B, \Gamma}{\vdash A \,\&\, B, \Gamma}\,(\&) \qquad \frac{\vdash A, \Gamma}{\vdash A \oplus B, \Gamma}\,(\oplus_1) \qquad \frac{\vdash B, \Gamma}{\vdash A \oplus B, \Gamma}\,(\oplus_2) \qquad \frac{}{\vdash \top, \Gamma}\,(\top)$$

Exponential rules:

$$\frac{\vdash A, \Gamma}{\vdash ?A, \Gamma}\,(?d) \qquad \frac{\vdash ?A, ?A, \Gamma}{\vdash ?A, \Gamma}\,(?c) \qquad \frac{\vdash \Gamma}{\vdash ?A, \Gamma}\,(?w) \qquad \frac{\vdash A, ?\Gamma}{\vdash !A, ?\Gamma}\,(!)$$

Quantifier rules:

$$X \text{ not free in } \Gamma \; \frac{\vdash A, \Gamma}{\vdash \forall X A, \Gamma}\,(\forall) \qquad \frac{\vdash A[B/X], \Gamma}{\vdash \exists X A, \Gamma}\,(\exists)$$

Optional multiplicative rules:

$$\frac{\vdash \Gamma \quad \vdash \Delta}{\vdash \Gamma, \Delta}\,(mix_2) \qquad \frac{}{\vdash}\,(mix_0)$$

Optional additive rules:

$$\frac{\vdash \Gamma \quad \vdash \Gamma}{\vdash \Gamma}\,(\cup) \qquad \frac{}{\vdash \Gamma}\,(\emptyset)$$

Figure 1.1: Rules of the Sequent Calculus of Linear Logic

When adding 2 or 0 as exponents in a sub-system, we mean this sub-system with in addition the mix_2- or mix_0-rules. For instance, MALL^2 is the sub-system MALL to which is added the mix_2-rule; and $\mathsf{MLL}_{uf}^{0,2}$ is MLL_{uf} with both mix_2- and mix_0-rules. Similarly, one may add $\cup$ or $\emptyset$ as exponents to indicate the associated sub-system containing these rules, or $\forall$ to add both quantifier rules.

1.1.3 Sequents as sets & The exchange rule

Looking at the usual definitions of the above sequent calculus in the literature (for instance [Gir11; LL22]), there are two differences here: sequents are sets and not (ordered) lists of formulas, and there is no exchange rule. The goal of this section is to explain and justify our unusual presentation.

In case sequents are lists, the **exchange rule** ex is the following, where σ is a permutation:

$$\frac{\vdash \Gamma}{\vdash \sigma(\Gamma)}\ (ex)$$

In most papers, immediately after giving this definition, it is written that the ex-rule will be used implicitly in the proofs, and this rule is usually not even named afterwards. In this thesis, we look at some precise properties of the sequent calculus, on which the ex-rule adds plenty of complexity: this is generally the case when defining transformations of proofs (in Section 1.3), and particularly when studying rule commutation. As a representative example, when considering in a proof two successive non-ex-rules s and r, with s above r, we sometimes need to commute them, putting r above s. In the framework with an ex-rule, we would not consider two rules but a stack of them, which is harder to manipulate. Moreover, when sequents are lists, the rules given on Figure 1.1 – which are the usual ones – need some ex-rules below and above them so as to move the main formula to the head of the list. Everything becomes tedious and unclear very quickly when taking this approach, and saying that ex-rules are used implicitly cannot be done here as we look closely on successive rules, and adding and removing ex-rules can have non-trivial impacts – when defining a decreasing measure for the transformation of a proof for instance.

Another solution could be to take all rules "up to permutation", with an ex-rule incorporated in the conclusion of each rule – or equivalently, considering proofs with exactly one ex-rule below each non-ex-rule, "merging" successive ex-rules by composing the associated permutations. Still, when following this approach one needs to put "up to permutation" everywhere, for instance in all conclusion sequents of our tables from Section 1.3. What is more, changing a permutation in an above rule can be done only up to admissibility, not deriviability: to permute the conclusion sequent of a proof, one needs to modify the last rule of this proof, while one wants the proofs before and after to be equal.

An easy solution is simply to remove the ex-rule, and to consider sequents as finite multisets instead of lists. Still, one have to be careful: considering sequents as multisets is not possible, as doing so forsake the Curry-Howard correspondence. It is important to be able to follow occurrences of a formula across rules: in

$$\frac{\vdash A, A, A}{\vdash A \parr A, A}\ (\parr)$$

we need to know which of the A above comes from which of the A below: which one is the left sub-formula of the $\parr$? the formula occurrence not a sub-formula of the $\parr$? The problem otherwise is the following: take π_1, π_2, π_3 and π_4 arbitrary proofs of a sequent $\vdash \Gamma$, and consider the following proofs:

$$
\begin{prooftree}
\AxiomC{π_1}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\AxiomC{π_2}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot, \bot \& \bot$}
\AxiomC{π_3}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\AxiomC{π_4}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot, \bot \& \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot \& \bot, \bot \& \bot$}
\RightLabel{$(⅋)$}
\UnaryInfC{$\vdash \Gamma, (\bot \& \bot) ⅋ (\bot \& \bot)$}
\end{prooftree}
\qquad
\begin{prooftree}
\AxiomC{}
\RightLabel{(1)}
\UnaryInfC{$\vdash 1$}
\RightLabel{$(\oplus_1)$}
\UnaryInfC{$\vdash 1 \oplus 1$}
\AxiomC{}
\RightLabel{(1)}
\UnaryInfC{$\vdash 1$}
\RightLabel{$(\oplus_2)$}
\UnaryInfC{$\vdash 1 \oplus 1$}
\RightLabel{$(\otimes)$}
\BinaryInfC{$\vdash (1 \oplus 1) \otimes (1 \oplus 1)$}
\end{prooftree}
$$

When composing them by cut, and applying one cut-elimination step (a transformation defined in Section 1.3), one obtain the following proof:

$$
\begin{prooftree}
\AxiomC{π_1}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\AxiomC{π_2}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot, \bot \& \bot$}
\AxiomC{π_3}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\AxiomC{π_4}
\noLine
\UnaryInfC{$\vdash \Gamma$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot$}
\RightLabel{$(\bot)$}
\UnaryInfC{$\vdash \Gamma, \bot, \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot, \bot \& \bot$}
\RightLabel{$(\&)$}
\BinaryInfC{$\vdash \Gamma, \bot \& \bot, \bot \& \bot$}
\AxiomC{}
\RightLabel{(1)}
\UnaryInfC{$\vdash 1$}
\RightLabel{$(\oplus_1)$}
\UnaryInfC{$\vdash 1 \oplus 1$}
\RightLabel{(cut)}
\BinaryInfC{$\vdash \Gamma, \bot \& \bot$}
\AxiomC{}
\RightLabel{(1)}
\UnaryInfC{$\vdash 1$}
\RightLabel{$(\oplus_2)$}
\UnaryInfC{$\vdash 1 \oplus 1$}
\RightLabel{(cut)}
\BinaryInfC{$\vdash \Gamma$}
\end{prooftree}
$$

Assuming sequents are multisets, in the above proof we do not know when looking at the sequent $\vdash \Gamma, \bot \& \bot, \bot \& \bot$ on which occurrences of $\bot \& \bot$ are applied the &-rules, thus which occurrences of $\bot$-formulas there are below the π_i. Then, applying further cut-elimination steps one either gets the proof π_2 or π_3. As we cannot distinguish between the two cases, this means both results can be found. As we usually wish to identify proofs up to cut-elimination, this implies that if sequents are multiset then π_2 and π_3 should be the same, meaning all proofs (of a given sequent) are equal! In such a case, the computational aspects of the logic are utterly lost.

Henceforth, we choose to consider as sequents sets of occurrences of formulas. Formally, this can be done in several ways. A first one is to seriously consider the notion of occurrence of a formula, and to put identifiers on all formulas, so as to follow them in a proof. This can be done by putting numbers on leaves, and taking care that in a sequent the same number cannot appear twice, along with a naming convention when using a $?_c$-rule for the duplicated formulas. A second solution is to associate a so-called threading function to a proof, which for sequents in the application of a rule associates each formula below with the corresponding formula above. Nonetheless, these solutions are not perfect: in case of a formalization on a proof assistant, tedious work awaits as one would need to give the occurrences explicitly, as well as preserving the properties we asked for (no duplicate identifiers, ...). In the rest of this thesis, when we need to consider occurrences, we do so by adding indices to formulas. For instance, $\top_3 \oplus \bot_6$ is the same formula as $\top_8 \oplus \bot_2$, but not the same occurrence.

A last consequence of having occurrences instead of an *ex*-rule is about isomorphisms of proof-nets seen as graphs, concepts that will be developed in Part II. When one quotients by permutations of the conclusions, checking whether two proof-nets are one and the same amounts to deciding if they are isomorphic as graphs, problem which in general is not that easy. Seeing sequents as sets of occurrences solves this problem: the vertices considered belong to a given occurrence of a formula in the sequent, that we do not identify with another occurrence in the same sequent, so that we only need to check equality of edges, which can be done in quadratic time (we have to check whether two sets are equal or not).

1.2 Basic concepts

The *cut*-rule allows composing proofs. Given proofs π and ϕ of respective sequents $\vdash \Gamma, A$ and $\vdash A^\perp, \Delta$, we denote by $\pi \overset{A}{\bowtie} \phi$ the proof obtained by adding a *cut*-rule on A between these two proofs:[2]

$$\pi \overset{A}{\bowtie} \phi = \quad \dfrac{\overset{\pi}{\vdash \Gamma, A} \qquad \overset{\phi}{\vdash A^\perp, \Delta}}{\vdash \Gamma, \Delta}\ (cut)$$

There are neutral elements for this composition, up to cut-elimination (a transformation defined in the next section): proofs composed of a unique *ax*-rule. We set ax_A the proof of $\vdash A^\perp, A$ containing just an *ax*-rule on A:

$$ax_A = \quad \dfrac{}{\vdash A^\perp, A}\ (ax)$$

Given a rule, its **main formula** is the one it acts on. For instance, with the notations from Figure 1.1, the main formula of the $\otimes$-rule is (the occurrence of) the $A \otimes B$ formula, and the main formula of the !-rule is the $!A$ formula. Each rule has a main formula, except the optional ones.

Given a formula, its **main connective** is the symbol at the root of its syntactic tree. When speaking of a **positive** formula, we mean a formula with main connective $\otimes$ or $\oplus$, a unit 1 or 0, or a positive atom X^+. A **negative** formula is one with main connective $⅋$ or &, a unit $\perp$ or $\top$, or a negated atom X^-.

A formula A is a **sub-formula** of a formula B if the syntactic tree of the former is a sub-tree of the syntactic tree of the latter. In case $A \neq B$, we say A is a **strict** sub-formula of B.

Definition 1.1 (Size of a formula)**.** The **size** $s(A)$ of a formula A is defined inductively as follows:

$$\begin{array}{lllll}
s(X^+) & = & s(X^-) & = & 1 \\
s(A ⅋ B) & = & s(A \otimes B) & = & s(A) + s(B) + 1 \\
s(\perp) & = & s(1) & = & 1 \\
s(A \,\&\, B) & = & s(A \oplus B) & = & s(A) + s(B) + 1 \\
s(\top) & = & s(0) & = & 1 \\
s(?A) & = & s(!A) & = & s(A) + 1 \\
s(\forall X A) & = & s(\exists X A) & = & s(A) + 1
\end{array}$$

One can prove the followings by induction.

Fact 1.2. *For any formula A, $s(A^\perp) = s(A)$.*

Fact 1.3. *Take a formula A, and set n the number of (occurrences of) atoms in A. Then $s(A) \geq n$. If A is a formula of MALL_{uf}, then $s(A) = 2 \times n - 1$.*

More generally, one can easily prove that the size $s(A)$ of A is its number of leaves plus its number of connectives. In particular, $s(A) \in \mathbb{N}^*$.

When considering substitutions, we also consider the case where a unit is replaced by a formula, which will be needed later for replacing units by atoms.

[2] Formally, one should consider occurrences of A and $A^\perp$ to have $\pi \overset{A}{\bowtie} \phi$ well-defined. As in our uses of this definition, the occurrences to consider are obvious, we abuse notations here so as to not surcharge by writing $\pi \overset{A_i, A_j^\perp}{\bowtie} \phi$.

Definition 1.4 (Substitution)**.** A **substitution** σ is an application from unsigned atoms and units to formulas. When this application is the identity except on leaves $(X_i)_{i\in I}$, that have for images $(A_i)_{i\in I} = (\sigma(X_i))_{i\in I}$, we denote σ by $[(A_i/X_i)_{i\in I}]$.

Given a formula A, $\sigma(A)$ is the formula obtained from A by replacing each atom X^+ by $\sigma(X)$, each X^- by $\sigma(X)^\perp$ and each unit X by $\sigma(X)$ (up to α-renaming for bounded variables of quantifiers). Here again, one can denote $\sigma(A)$ by $A[(A_i/X_i)_{i\in I}]$.

In case a substitution acts only on atoms, it can be extended to proofs in the obvious way, up to some α-renaming for bounded variables of quantifiers.

The extension of proofs to substitution is well-defined thanks to the following.

Fact 1.5. *Given a formula A and a substitution σ acting only on atoms, $\sigma(A^\perp) = \sigma(A)^\perp$.*

Proof. By induction on the formula. ☐

The main difference between MALL and MLL is the &-rule, which introduces some sharing of the context Γ. From this comes the notion of a *slice* [Gir87; Gir96] which is a partial proof missing some additive component.

Definition 1.6 (Slice)**.** For π a proof, consider the (non-correct) proof tree obtained by deleting one of the two sub-trees of each &-rule of π (thus, in the new proof tree, &-rules are unary):

$$\frac{\vdash A, \Gamma}{\vdash A \mathbin{\&} B, \Gamma}\ (\&_1) \qquad \frac{\vdash B, \Gamma}{\vdash A \mathbin{\&} B, \Gamma}\ (\&_2)$$

The remaining rules form a **slice** of π. We denote by $\mathcal{S}(\pi)$ the set of slices of π.

MALL slices satisfy a linearity property (validated by proofs of MLL as well): any connective in the conclusion is introduced by at most one rule in a slice.

A rule has the **sub-sequent property** if every sequent at its premises is a sub-sequent of its conclusion sequent. All rules of linear logic (on Figure 1.1) have this property, excepted for the *cut*-, $?_c$- and $\exists$-rules. A rule has the **strict sub-sequent property** if every sequent at its premises is a strict sub-sequent of its conclusion sequent.

Fact 1.7. *Rules of* $\mathsf{MALL}^{0,2}$ *except cut have the sub-sequent property, and rules of* MALL^0 *except cut have the strict sub-sequent property.*

Remark optional rules are the only ones that can have both premise and conclusion sequents empty.

Fact 1.8. *A proof in which all sequents are empty is a proof made exclusively of mix_2-, mix_0-, $\cup$- and $\emptyset$-rules.*

1.3 Transformations of proofs

We define here the usual transformations of proofs of linear logic: axiom-expansion, cut-elimination and rule commutation. These transformations are *contextual* – they can be applied in a sub-proof, *i.e.* they are closed by context – and most of them are *local* – they modify only part of a proof, rules that are above or below stay the same. Properties of these transformations as rewriting systems are the object of the next chapter, Chapter 2. All tables giving these transformations can be found at the end of this chapter (from Page 33 to Page 52).

1.3.1 Axiom-expansion

As in many logics with an *ax*-rule, one may consider proofs with the *ax*-rule reduced to the case where it is applied on an atom, *i.e.* with $A = X^+$:

$$\dfrac{}{\vdash X^-, X^+}\ (ax)$$

Such proofs are called **atomic-axiom**, and correspond to normal forms for a rewriting procedure named axiom-expansion.

Definition 1.9. Axiom-expansion is the rewriting system denoted $\xrightarrow{\eta}$ and whose rules are described on Table 1.1.

In the axiom-expansion case $\forall - \exists$, we instantiate the $\exists$-rule with X^+, obtaining $A[X^+/X] = A$.

This rewriting corresponds to η-expansion in λ-calculus, hence we also call it η-expansion, and atomic-axiom proofs will also be called η-normal proofs (and there will be no ambiguity, for we will not speak of λ-calculus in this thesis). We use the notation $=_\eta$ for equality of proofs up to axiom-expansion.

1.3.2 Cut-elimination

In most systems with a *cut*-rule, one shows this rule is *admissible*, *i.e.* that the same sequents can be proved with the *cut*-rule than without it. The procedure turning a proof into a *cut*-free one is called *cut-elimination*, and introduces a notion of computation in the logic.

Definition 1.10. Cut-elimination is the rewriting system denoted $\xrightarrow{\beta}$ and whose rules are described on Tables 1.2 to 1.4, up to commuting the two branches of any *cut*-rule. (This means one should take each rule in these tables and also consider a version of this rule with the left and right premises of any *cut*-rule swapped. In particular, for the $cut - cut$ commutative case, there are in fact 4 rewriting rules.)

In the $?c - !$ and $?w - !$ exponential key cases, the notation of a doubled inference rule means we have to apply the corresponding rule a certain number of times, once on each formula of $?\Delta$. The order in which these rules are applied has no importance; in other words, this step is non-deterministic according to the order in which these rules are applied.

The $\forall - \exists$ key case is not local due to the substitution of X by B in the proof π_1. It is well-defined, for $A^\perp[B/X] = (A[B/X])^\perp$ (Fact 1.5).

In the $\forall - cut$ commutative case, one may need to α-rename the atom X into a new atom Y so as to respect Y not free in Γ, Δ.

Remark 1.11. Usually, when defining cut-elimination in linear logic, an additional $⅋ - \otimes$ key case is given, which is the following:

$$\dfrac{\dfrac{\dfrac{\pi_1}{\vdash A, \Gamma} \quad \dfrac{\pi_2}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\ (\otimes) \quad \dfrac{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash \Gamma, \Delta, \Sigma}\ (cut) \quad \xrightarrow{\beta} \quad \dfrac{\dfrac{\pi_2}{\vdash B, \Delta} \quad \dfrac{\dfrac{\pi_1}{\vdash A, \Gamma} \quad \dfrac{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash B^\perp, \Gamma, \Sigma}\ (cut)}{\vdash \Gamma, \Delta, \Sigma}\ (cut)$$

This case can be simulated using the given $⅋ - \otimes$ key case followed by a $cut - cut$ commutative case. Therefore, the properties we give in Chapter 2 also holds when adding this case to our rewriting system.

Remark that in Tables 1.2 to 1.4 defining cut-elimination, there are some absent cases. Namely there is:

- no $\top - 0$ key case as there is no rule for 0;
- no $ax - cut$ commutative case as the ax-rule has no context;
- no $1 - cut$ commutative case as the 1-rule has no context;
- no $0 - cut$ commutative case as there is no rule for 0;
- no general $! - cut$ commutative case as the !-rule has a constraint on its context;
- no $mix_0 - cut$ commutative case as the mix_0-rule has no context.

Following the Curry-Howard correspondence, cut-elimination corresponds to β-reduction in λ-calculus. Thence, we call β-reduction a step of the cut-elimination procedure, and β-normal proofs are the *cut*-free ones (and as for axiom-expansion, there will be no risk of confusion). As with axiom-expansion, $=_\beta$ denotes equality up to cut-elimination. We also use the notation $=_{\beta\eta}$ for equality of proofs up to cut-elimination and axiom-expansion.

When studying properties of cut-elimination, we will need to consider a bipartition of $\xrightarrow{\beta}$.

Definition 1.12. We denote by $\xrightarrow{\overline{\beta}}$ a $\xrightarrow{\beta}$ step other than a $cut - cut$ commutation. We call $\overset{c}{\vdash\!\dashv}$ the $cut - cut$ commutation, which is a symmetric relation.

Henceforth, $\xrightarrow{\beta} = \xrightarrow{\overline{\beta}} \cup \overset{c}{\vdash\!\dashv}$.

Fact 1.13. *As long as there exists a cut-rule, a* $\xrightarrow{\overline{\beta}}$ *cut-elimination step can be applied. Thus, cut-free proofs correspond to proofs in normal form for* $\xrightarrow{\overline{\beta}}$*, as well as proofs in normal form for* $\xrightarrow{\beta}$*.*

1.3.3 Rule commutation

A last relation on proofs is considered in this thesis, *rule commutation*. These commutations associate proofs which differ only by the order in which their rules are applied. In other words, this relation appears when one can apply two successive rules, say r and s, and both applying r then s or s then r would do the job. That is why rule commutation can be viewed as "bureaucracy" [Gir01], as both choices are adequate, and in fact are not really a choice as we would like to consider both proofs to be the same. But due to the definition of a proof, we must choose an arbitrary order and apply one of the rules first, for it is impossible to apply two rules at once in sequent calculus. As we will see in Chapter 2, rule commutation and cut-elimination are closely related, allowing us to make more precise the intuitions written just above.

Definition 1.14. Rule commutation is the symmetric relation denoted $\overset{r}{\vdash\!\dashv}$ and whose rules are described on Tables 1.5 to 1.20; all sub-proofs named on these tables must be cut-free.

The commutations from the given tables in Definition 1.14 are exactly those given by the following method. Consider a *cut*-rule c and all couples of non *cut*-rules that can be the premises of c, such that two commutative cut-elimination cases can be applied on c. Comparing the results of applying the left commutative step then the right, against applying the right then the left, yields the rule commutations. This is why there is no commutation with an ax-, 1-, 0- nor mix_0-rule for there is no associated commutative cut-elimination case for these rules.

Remark 1.15. In our definition of rule commutation, there is no commutation with an !-rule, because the commutative cut-elimination case associated to the !-rule must have the !-rule of the cut formula on one of its premises due to the constraints on the context of a !-rule. In particular, using our method to get the rule commutations, we cannot have two commutative cases on a same *cut*-rule where one of these cases is a $! - cut$ commutation. This is in shard contrast contrary to what usually appears in the literature where there are the following $! - ?_c$ and $! - ?_w$ commutations:

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, ?\Gamma}}{\vdash !A, ?B, ?B, ?\Gamma}\,(!)}{\vdash !A, ?B, ?\Gamma}\,(?_c) \equiv \dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, ?\Gamma}}{\vdash A, ?B, ?\Gamma}\,(?_c)}{\vdash !A, ?B, ?\Gamma}\,(!) \qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, ?\Gamma}}{\vdash !A, ?\Gamma}\,(!)}{\vdash !A, ?B, ?\Gamma}\,(?_w) \equiv \dfrac{\dfrac{\overset{\pi}{\vdash A, ?\Gamma}}{\vdash A, ?B, ?\Gamma}\,(?_w)}{\vdash !A, ?B, ?\Gamma}\,(!)$$

We think these two commutations do not have the same status as the ones given in Definition 1.14, because we conjecture them not to be included in equality up to cut-elimination. As an insight for the knowing reader, consider the usual proof-nets for $\mathsf{MELL}^{0,2}$. Cut-elimination in this syntax mimics the one of sequent calculus and is convergent. If the $! - ?_c$ and $! - ?_w$ commutations were included in β-equality, then two $\mathsf{MELL}^{0,2}$ cut-free proofs differing by such a commutation would have the same proof-net, which is not the case. Indeed, these two commutations from sequent calculus have corresponding ones in proof-nets for $\mathsf{MELL}^{0,2}$, contrary to the rule commutations of our definition.

Remark that for no rule other than optional and exponential ones can a commutation with a !-rule be defined, because such a rule has a main formula which is not a ? one. One could imagine commutations of the !-rule with the $?_d$-rule, or with the optional rules as follows.

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?\Gamma}}{\vdash !A, ?B, ?\Gamma}\,(!)}{\vdash !A, ??B, ?\Gamma}\,(?_d) \equiv \dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?\Gamma}}{\vdash A, ??B, ?\Gamma}\,(?_d)}{\vdash !A, ??B, ?\Gamma}\,(!)$$

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A, ?\Gamma}}{\vdash !A, ?\Gamma}\,(!) \qquad \overset{\pi_2}{\vdash ?\Delta}}{\vdash !A, ?\Gamma, ?\Delta}\,(mix_2) \equiv \dfrac{\dfrac{\overset{\pi_1}{\vdash A, ?\Gamma} \qquad \overset{\pi_2}{\vdash ?\Delta}}{\vdash A, ?\Gamma, ?\Delta}\,(mix_2)}{\vdash !A, ?\Gamma, ?\Delta}\,(!)$$

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A, ?\Gamma}}{\vdash !A, ?\Gamma}\,(!) \qquad \dfrac{\overset{\pi_2}{\vdash A, ?\Gamma}}{\vdash !A, ?\Gamma}\,(!)}{\vdash !A, ?\Gamma}\,(\cup) \equiv \dfrac{\dfrac{\overset{\pi_1}{\vdash A, ?\Gamma} \qquad \overset{\pi_2}{\vdash A, ?\Gamma}}{\vdash A, ?\Gamma}\,(\cup)}{\vdash !A, ?\Gamma}\,(!) \qquad \dfrac{}{\vdash !A, ?\Gamma}\,(\emptyset) \equiv \dfrac{\dfrac{}{\vdash A, ?\Gamma}\,(\emptyset)}{\vdash !A, ?\Gamma}\,(!)$$

Here again, we conjecture these commutations are not included in β-equality, *i.e.* preserve provability but not computational content.

Also, there is no $\emptyset - \emptyset$ rule commutation as it is the identity: $\dfrac{}{\vdash \Gamma}\,(\emptyset) \xrightarrow{c^{\emptyset}_{\emptyset}} \dfrac{}{\vdash \Gamma}\,(\emptyset)$

The definition we give of rule commutation corresponds to rule commutation in cut-free linear logic, *i.e.* with no *cut*-rules above the commutation, whereas there may be *cut*-rules below the commutation, in the context. In particular, in a $\top - \otimes$ (resp. $\& - \otimes$) commutation we assume the created or erased sub-proof (resp. duplicated or superimposed sub-proof) to be cut-free. This choice is more appropriate for our setting, as it has less cases and corresponds to the equivalence relation between normal forms, namely cut-free proofs. A more general theory of rule commutation exists.

Definition 1.16. Rule commutation with cuts is the symmetric relation denoted $\overset{rc}{\longmapsto\!\dashv}$ and whose rules are the union of those described on Tables 1.5 to 1.20 – no sub-proofs named on these tables is supposed be cut-free – with the *cut*-commutations, which are the symmetric closure of the commutative cases of cut-elimination on Tables 1.3 and 1.4.

See [HG16] for tables with the rule commutation with cuts in MALL^2_{uf}.

1.3.4 Rétoré transformations

There are some more transformations on proofs in linear logic, that one may or may not want to include in equality of proofs. Those concern the exponential rules, as well as the two pairs of optional rules: one may want (or not) that $?_w$ is neutral for $?_c$, mix_0 for mix_2 and $\emptyset$ for $\cup$.

Definition 1.17. Directed Rétoré transformation is the relation denoted $\overset{o}{\rightsquigarrow}$ and defined as the union of $\overset{oe}{\rightsquigarrow}$, $\overset{om}{\rightsquigarrow}$ and $\overset{oa}{\rightsquigarrow}$ whose rules are described on Table 1.21.

The first pair of rewriting rules, $\overset{oe}{\rightsquigarrow}$, are called the **?-Rétoré** transformations. The second pair, $\overset{om}{\rightsquigarrow}$, are the *mix*-**Rétoré** transformations. Finally, the last pair, $\overset{oa}{\rightsquigarrow}$, are the $\emptyset$-**Rétoré** transformations.

In the $\overset{oe}{\rightsquigarrow}$ transformation, $?A_1$ and $?A_2$ represent different occurrences of the same formula $?A$.

Definition 1.18. Rétoré transformation is the symmetric relation denoted $\overset{o}{\longmapsto\!\dashv}$ and defined as the symmetric closure of $\overset{o}{\rightsquigarrow}$, or equivalently as the union of $\overset{oe}{\longmapsto\!\dashv}$, $\overset{om}{\longmapsto\!\dashv}$ and $\overset{oa}{\longmapsto\!\dashv}$ which are respectively the symmetric closure of $\overset{oe}{\rightsquigarrow}$, $\overset{om}{\rightsquigarrow}$ and $\overset{oa}{\rightsquigarrow}$.

When considering $\overset{o}{\rightsquigarrow}$ and $\overset{o}{\longmapsto\!\dashv}$, one can choose to take (or not) any of the $\overset{oe}{\rightsquigarrow}$, $\overset{om}{\rightsquigarrow}$ and $\overset{oa}{\rightsquigarrow}$ rules, leading to $2^3 = 8$ different systems. As before, $=_{\beta\eta o}$ denotes equality up to cut-elimination, axiom-expansion and (one of the 8 sub-systems of) Rétoré transformations. We also use the notation $=_{\beta o}$ for equality of proofs up to cut-elimination and Rétoré transformations.

$\parr - \otimes$	$\dfrac{}{\vdash B^{\perp} \parr A^{\perp}, A \otimes B}\ (ax)$	$\xrightarrow{\eta}$	$\dfrac{\dfrac{\dfrac{}{\vdash A, A^{\perp}}\ (ax) \qquad \dfrac{}{\vdash B, B^{\perp}}\ (ax)}{\vdash B^{\perp}, A^{\perp}, A \otimes B}\ (\otimes)}{\vdash B^{\perp} \parr A^{\perp}, A \otimes B}\ (\parr)$
$\perp - 1$	$\dfrac{}{\vdash \perp, 1}\ (ax)$	$\xrightarrow{\eta}$	$\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \perp, 1}\ (\perp)$
$\& - \oplus$	$\dfrac{}{\vdash B^{\perp} \& A^{\perp}, A \oplus B}\ (ax)$	$\xrightarrow{\eta}$	$\dfrac{\dfrac{\dfrac{}{\vdash B, B^{\perp}}\ (ax)}{\vdash B^{\perp}, A \oplus B}\ (\oplus_2) \qquad \dfrac{\dfrac{}{\vdash A, A^{\perp}}\ (ax)}{\vdash A^{\perp}, A \oplus B}\ (\oplus_1)}{\vdash B^{\perp} \& A^{\perp}, A \oplus B}\ (\&)$
$\top - 0$	$\dfrac{}{\vdash \top, 0}\ (ax)$	$\xrightarrow{\eta}$	$\dfrac{}{\vdash \top, 0}\ (\top)$
$? - !$	$\dfrac{}{\vdash ?A^{\perp}, !A}\ (ax)$	$\xrightarrow{\eta}$	$\dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\ (ax)}{\vdash ?A^{\perp}, A}\ (?d)}{\vdash ?A^{\perp}, !A}\ (!)$
$\forall - \exists$	$\dfrac{}{\vdash \forall X A^{\perp}, \exists X A}\ (ax)$	$\xrightarrow{\eta}$	X not free in $\exists X A$ $\dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\ (ax)}{\vdash A^{\perp}, \exists X A}\ (\exists)}{\vdash \forall X A^{\perp}, \exists X A}\ (\forall)$

Table 1.1: Axiom-expansion

ax	$\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \overset{\pi}{\vdash A,\Gamma}}{\vdash A,\Gamma}\,(cut)$	$\xrightarrow{\beta}$	$\overset{\pi}{\vdash A,\Gamma}$
$\parr - \otimes$	$\dfrac{\dfrac{\overset{\pi}{\vdash B^\perp, A^\perp,\Gamma}}{\vdash B^\perp \parr A^\perp,\Gamma}\,(\parr) \qquad \dfrac{\overset{\phi}{\vdash A,\Delta} \quad \overset{\tau}{\vdash B,\Sigma}}{\vdash A\otimes B,\Delta,\Sigma}\,(\otimes)}{\vdash \Gamma,\Delta,\Sigma}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B^\perp,A^\perp,\Gamma} \quad \overset{\tau}{\vdash B,\Sigma}}{\vdash A^\perp,\Gamma,\Sigma}\,(cut) \qquad \overset{\phi}{\vdash A,\Delta}}{\vdash\Gamma,\Delta,\Sigma}\,(cut)$
$\bot - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash\Gamma}}{\vdash\bot,\Gamma}\,(\bot) \qquad \dfrac{}{\vdash 1}\,(1)}{\vdash\Gamma}\,(cut)$	$\xrightarrow{\beta}$	$\overset{\pi}{\vdash\Gamma}$
$\& - \oplus_1$	$\dfrac{\dfrac{\overset{\pi}{\vdash B^\perp,\Gamma}\quad\overset{\phi}{\vdash A^\perp,\Gamma}}{\vdash B^\perp \& A^\perp,\Gamma}\,(\&) \qquad \dfrac{\overset{\tau}{\vdash A,\Delta}}{\vdash A\oplus B,\Delta}\,(\oplus_1)}{\vdash\Gamma,\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\overset{\phi}{\vdash A^\perp,\Gamma}\quad\overset{\tau}{\vdash A,\Delta}}{\vdash\Gamma,\Delta}\,(cut)$
$\& - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash B^\perp,\Gamma}\quad\overset{\phi}{\vdash A^\perp,\Gamma}}{\vdash B^\perp \& A^\perp,\Gamma}\,(\&) \qquad \dfrac{\overset{\tau}{\vdash B,\Delta}}{\vdash A\oplus B,\Delta}\,(\oplus_2)}{\vdash\Gamma,\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi}{\vdash B^\perp,\Gamma}\quad\overset{\tau}{\vdash B,\Delta}}{\vdash\Gamma,\Delta}\,(cut)$
$?d - !$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp,\Gamma}}{\vdash ?A^\perp,\Gamma}\,(?d)\qquad\dfrac{\overset{\phi}{\vdash A,?\Delta}}{\vdash !A,?\Delta}\,(!)}{\vdash\Gamma,?\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi}{\vdash A^\perp,\Gamma}\quad\overset{\phi}{\vdash A,?\Delta}}{\vdash\Gamma,?\Delta}\,(cut)$
$?c - !$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A^\perp,?A^\perp,\Gamma}}{\vdash ?A^\perp,\Gamma}\,(?c)\qquad\dfrac{\overset{\phi}{\vdash A,?\Delta}}{\vdash !A,?\Delta}\,(!)}{\vdash\Gamma,?\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash ?A^\perp,?A^\perp,\Gamma}\quad \dfrac{\overset{\phi}{\vdash A,?\Delta}}{\vdash !A,?\Delta}\,(!)}{\vdash ?A^\perp,\Gamma,?\Delta}\,(cut)\qquad\dfrac{\overset{\phi}{\vdash A,?\Delta}}{\vdash !A,?\Delta}\,(!)}{\vdash\Gamma,?\Delta,?\Delta}\,(cut)}{\vdash\Gamma,?\Delta}\,(?c)$
$?w - !$	$\dfrac{\dfrac{\overset{\pi}{\vdash\Gamma}}{\vdash ?A^\perp,\Gamma}\,(?w)\qquad\dfrac{\overset{\phi}{\vdash A,?\Delta}}{\vdash !A,?\Delta}\,(!)}{\vdash\Gamma,?\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi}{\vdash\Gamma}}{\vdash\Gamma,?\Delta}\,(?w)$
$\forall - \exists$	X not free in Γ $\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp,\Gamma}}{\vdash\forall X A^\perp,\Gamma}\,(\forall)\qquad\dfrac{\overset{\phi}{\vdash A[B/X],\Delta}}{\vdash\exists X A,\Delta}\,(\exists)}{\vdash\Gamma,\Delta}\,(cut)$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi[B/X]}{\vdash A[B/X]^\perp,\Gamma}\quad\overset{\phi}{\vdash A[B/X],\Delta}}{\vdash\Gamma,\Delta}\,(cut)$

Table 1.2: Cut-elimination – Key cases

$\mathit{cut} - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B^\perp, \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B^\perp, \Gamma, \Delta}\,(\mathit{cut}) \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash \Gamma, \Delta, \Sigma}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B^\perp, \Gamma} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A^\perp, \Gamma, \Sigma}\,(\mathit{cut}) \quad \overset{\phi}{\vdash A, \Delta}}{\vdash \Gamma, \Delta, \Sigma}\,(\mathit{cut})$
$\parr - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, C, \Gamma}}{\vdash A^\perp, B \parr C, \Gamma}\,(\parr) \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B \parr C, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, C, \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B, C, \Gamma, \Delta}\,(\mathit{cut})}{\vdash B \parr C, \Gamma, \Delta}\,(\parr)$
$\otimes - \mathit{cut} - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma} \quad \overset{\phi}{\vdash C, \Delta}}{\vdash A^\perp, B \otimes C, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash A, \Sigma}}{\vdash B \otimes C, \Gamma, \Delta, \Sigma}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma} \quad \overset{\tau}{\vdash A, \Sigma}}{\vdash B, \Gamma, \Sigma}\,(\mathit{cut}) \quad \overset{\phi}{\vdash C, \Delta}}{\vdash B \otimes C, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - \mathit{cut} - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma} \quad \overset{\phi}{\vdash A^\perp, C, \Delta}}{\vdash A^\perp, B \otimes C, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash A, \Sigma}}{\vdash B \otimes C, \Gamma, \Delta, \Sigma}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi}{\vdash B, \Gamma} \quad \dfrac{\overset{\phi}{\vdash A^\perp, C, \Delta} \quad \overset{\tau}{\vdash A, \Sigma}}{\vdash C, \Delta, \Sigma}\,(\mathit{cut})}{\vdash B \otimes C, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\bot - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, \Gamma}}{\vdash A^\perp, \bot, \Gamma}\,(\bot) \quad \overset{\phi}{\vdash A, \Delta}}{\vdash \bot, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}\,(\mathit{cut})}{\vdash \bot, \Gamma, \Delta}\,(\bot)$
$\& - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma} \quad \overset{\phi}{\vdash A^\perp, C, \Gamma}}{\vdash A^\perp, B \,\&\, C, \Gamma}\,(\&) \quad \overset{\tau}{\vdash A, \Delta}}{\vdash B \,\&\, C, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma} \quad \overset{\tau}{\vdash A, \Delta}}{\vdash B, \Gamma, \Delta}\,(\mathit{cut}) \quad \dfrac{\overset{\phi}{\vdash A^\perp, C, \Gamma} \quad \overset{\tau}{\vdash A, \Delta}}{\vdash C, \Gamma, \Delta}\,(\mathit{cut})}{\vdash B \,\&\, C, \Gamma, \Delta}\,(\&)$
$\oplus_1 - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma}}{\vdash A^\perp, B \oplus C, \Gamma}\,(\oplus_1) \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B \oplus C, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, B, \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B, \Gamma, \Delta}\,(\mathit{cut})}{\vdash B \oplus C, \Gamma, \Delta}\,(\oplus_1)$
$\oplus_2 - \mathit{cut}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, C, \Gamma}}{\vdash A^\perp, B \oplus C, \Gamma}\,(\oplus_2) \quad \overset{\phi}{\vdash A, \Delta}}{\vdash B \oplus C, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^\perp, C, \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash C, \Gamma, \Delta}\,(\mathit{cut})}{\vdash B \oplus C, \Gamma, \Delta}\,(\oplus_2)$
$\top - \mathit{cut}$	$\dfrac{\dfrac{}{\vdash A^\perp, \top, \Gamma}\,(\top) \quad \overset{\pi}{\vdash A, \Delta}}{\vdash \top, \Gamma, \Delta}\,(\mathit{cut})$	$\xrightarrow{\beta}$	$\dfrac{}{\vdash \top, \Gamma, \Delta}\,(\top)$

Table 1.3: Cut-elimination – Commutative cases (Part 1/2)

$?d - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B, \Gamma}}{\vdash A^{\perp}, ?B, \Gamma}{\scriptstyle (?d)} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B, \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash B, \Gamma, \Delta}{\scriptstyle (cut)}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (?d)}$
$?c - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, ?B, ?B, \Gamma}}{\vdash A^{\perp}, ?B, \Gamma}{\scriptstyle (?c)} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, ?B, ?B, \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash ?B, ?B, \Gamma, \Delta}{\scriptstyle (cut)}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (?c)}$
$?w - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma}}{\vdash A^{\perp}, ?B, \Gamma}{\scriptstyle (?w)} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}{\scriptstyle (cut)}}{\vdash ?B, \Gamma, \Delta}{\scriptstyle (?w)}$
$! - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A^{\perp}, B, ?\Gamma}}{\vdash ?A^{\perp}, !B, ?\Gamma}{\scriptstyle (!)} \qquad \dfrac{\overset{\phi}{\vdash A, ?\Delta}}{\vdash !A, ?\Delta}{\scriptstyle (!)}}{\vdash !B, ?\Gamma, ?\Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A^{\perp}, B, ?\Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, ?\Delta}}{\vdash !A, ?\Delta}{\scriptstyle (!)}}{\vdash B, ?\Gamma, ?\Delta}{\scriptstyle (cut)}}{\vdash !B, ?\Gamma, ?\Delta}{\scriptstyle (!)}$
$\forall - cut$	X not free in $A^{\perp},\Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B, \Gamma}}{\vdash A^{\perp}, \forall XB, \Gamma}{\scriptstyle (\forall)} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash \forall XB, \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	X not free in Γ,Δ $\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B, \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash B, \Gamma, \Delta}{\scriptstyle (cut)}}{\vdash \forall XB, \Gamma, \Delta}{\scriptstyle (\forall)}$
$\exists - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B[C/X], \Gamma}}{\vdash A^{\perp}, \exists XB, \Gamma}{\scriptstyle (\exists)} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash \exists XB, \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, B[C/X], \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash B[C/X]\Gamma, \Delta}{\scriptstyle (cut)}}{\vdash \exists XB, \Gamma, \Delta}{\scriptstyle (\exists)}$
$mix_2 - cut - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash A^{\perp}, \Gamma, \Delta}{\scriptstyle (mix_2)} \qquad \overset{\tau}{\vdash A, \Sigma}}{\vdash \Gamma, \Delta, \Sigma}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Sigma}{\scriptstyle (cut)} \qquad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta, \Sigma}{\scriptstyle (mix_2)}$
$mix_2 - cut - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash A^{\perp}, \Delta}}{\vdash A^{\perp}, \Gamma, \Delta}{\scriptstyle (mix_2)} \qquad \overset{\tau}{\vdash A, \Sigma}}{\vdash \Gamma, \Delta, \Sigma}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A^{\perp}, \Delta} \qquad \overset{\tau}{\vdash A, \Sigma}}{\vdash \Delta, \Sigma}{\scriptstyle (cut)}}{\vdash \Gamma, \Delta, \Sigma}{\scriptstyle (mix_2)}$
$\cup - cut$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\phi}{\vdash A^{\perp}, \Gamma}}{\vdash A^{\perp}, \Gamma}{\scriptstyle (\cup)} \qquad \overset{\tau}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\tau}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}{\scriptstyle (cut)} \qquad \dfrac{\overset{\phi}{\vdash A^{\perp}, \Gamma} \qquad \overset{\tau}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}{\scriptstyle (cut)}}{\vdash \Gamma, \Delta}{\scriptstyle (\cup)}$
$\emptyset - cut$	$\dfrac{\dfrac{}{\vdash A^{\perp}, \Gamma}{\scriptstyle (\emptyset)} \qquad \overset{\pi}{\vdash A, \Delta}}{\vdash \Gamma, \Delta}{\scriptstyle (cut)}$	$\xrightarrow{\beta}$	$\dfrac{}{\vdash \Gamma, \Delta}{\scriptstyle (\emptyset)}$

Table 1.4: Cut-elimination – Commutative cases (Part 2/2)

$\parr - \parr$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, D, \Gamma}}{\vdash A \parr B, C, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr)$	$\xrightarrow{C^{\parr}_{\parr}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, D, \Gamma}}{\vdash A, B, C \parr D, \Gamma}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr)$
$\parr - \otimes - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr) \quad \overset{\phi}{\vdash D, \Delta}}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\otimes)$	$\xrightarrow{C^{\parr}_{\otimes}}$ $\xleftarrow[C^{\otimes}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma} \quad \overset{\phi}{\vdash D, \Delta}}{\vdash A, B, C \otimes D, \Gamma, \Delta}\,(\otimes)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\parr)$
$\parr - \otimes - 2$	$\dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, B, D, \Delta}}{\vdash A \parr B, D, \Delta}\,(\parr)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\otimes)$	$\xrightarrow{C^{\parr}_{\otimes}}$ $\xleftarrow[C^{\otimes}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \overset{\phi}{\vdash A, B, D, \Delta}}{\vdash A, B, C \otimes D, \Gamma, \Delta}\,(\otimes)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\parr)$
$\parr - \bot$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}\,(\parr)}{\vdash A \parr B, \bot, \Gamma}\,(\bot)$	$\xrightarrow{C^{\parr}_{\bot}}$ $\xleftarrow[C^{\bot}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, B, \bot, \Gamma}\,(\bot)}{\vdash A \parr B, \bot, \Gamma}\,(\parr)$
$\parr - \&$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr) \quad \dfrac{\overset{\phi}{\vdash A, B, D, \Gamma}}{\vdash A \parr B, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(\&)$	$\xrightarrow{C^{\parr}_{\&}}$ $\xleftarrow[C^{\&}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma} \quad \overset{\phi}{\vdash A, B, D, \Gamma}}{\vdash A, B, C \,\&\, D, \Gamma}\,(\&)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(\parr)$
$\parr - \oplus_1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr)}{\vdash A \parr B, C \oplus D, \Gamma}\,(\oplus_1)$	$\xrightarrow{C^{\parr}_{\oplus_1}}$ $\xleftarrow[C^{\oplus_1}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A, B, C \oplus D, \Gamma}\,(\oplus_1)}{\vdash A \parr B, C \oplus D, \Gamma}\,(\parr)$
$\parr - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, D, \Gamma}}{\vdash A \parr B, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \oplus D, \Gamma}\,(\oplus_2)$	$\xrightarrow{C^{\parr}_{\oplus_2}}$ $\xleftarrow[C^{\oplus_2}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, D, \Gamma}}{\vdash A, B, C \oplus D, \Gamma}\,(\oplus_2)}{\vdash A \parr B, C \oplus D, \Gamma}\,(\parr)$
$\parr - \top$	$\dfrac{}{\vdash A \parr B, \top, \Gamma}\,(\top)$	$\xrightarrow{C^{\parr}_{\top}}$ $\xleftarrow[C^{\top}_{\parr}]{}$	$\dfrac{\dfrac{}{\vdash A, B, \top, \Gamma}\,(\top)}{\vdash A \parr B, \top, \Gamma}\,(\parr)$
$\parr - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr)}{\vdash A \parr B, ?C, \Gamma}\,(?_d)$	$\xrightarrow{C^{\parr}_{?_d}}$ $\xleftarrow[C^{?_d}_{\parr}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A, B, ?C, \Gamma}\,(?_d)}{\vdash A \parr B, ?C, \Gamma}\,(\parr)$

Table 1.5: Rule commutation (Part 1/16)

$\parr - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, ?C, ?C, \Gamma}}{\vdash A \parr B, ?C, ?C, \Gamma}(\parr)}{\vdash A \parr B, ?C, \Gamma}(?_c)$	$\underset{C^{?_c}_{\parr}}{\overset{C^{\parr}_{?_c}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, ?C, ?C, \Gamma}}{\vdash A, B, ?C, \Gamma}(?_c)}{\vdash A \parr B, ?C, \Gamma}(\parr)$
$\parr - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}(\parr)}{\vdash A \parr B, ?C, \Gamma}(?_w)$	$\underset{C^{?_w}_{\parr}}{\overset{C^{\parr}_{?_w}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, B, ?C, \Gamma}(?_w)}{\vdash A \parr B, ?C, \Gamma}(\parr)$
$\parr - \forall$	X not free in $A \parr B, \Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, C, \Gamma}(\parr)}{\vdash A \parr B, \forall X C, \Gamma}(\forall)$	$\underset{C^{\forall}_{\parr}}{\overset{C^{\parr}_{\forall}}{\rightleftarrows}}$	X not free in A, B, Γ $\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, B, \forall X C, \Gamma}(\forall)}{\vdash A \parr B, \forall X C, \Gamma}(\parr)$
$\parr - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C[D/X], \Gamma}}{\vdash A \parr B, C[D/X], \Gamma}(\parr)}{\vdash A \parr B, \exists X C, \Gamma}(\exists)$	$\underset{C^{\exists}_{\parr}}{\overset{C^{\parr}_{\exists}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C[D/X], \Gamma}}{\vdash A, B, \exists X C, \Gamma}(\exists)}{\vdash A \parr B, \exists X C, \Gamma}(\parr)$
$\parr - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}(\parr) \quad \overset{\phi}{\vdash \Delta}}{\vdash A \parr B, \Gamma, \Delta}(mix_2)$	$\underset{C^{mix_2}_{\parr}}{\overset{C^{\parr}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A, B, \Gamma, \Delta}(mix_2)}{\vdash A \parr B, \Gamma, \Delta}(\parr)$
$\parr - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, B, \Delta}}{\vdash A \parr B, \Delta}(\parr)}{\vdash A \parr B, \Gamma, \Delta}(mix_2)$	$\underset{C^{mix_2}_{\parr}}{\overset{C^{\parr}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, B, \Delta}}{\vdash A, B, \Gamma, \Delta}(mix_2)}{\vdash A \parr B, \Gamma, \Delta}(\parr)$
$\parr - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}(\parr) \quad \dfrac{\overset{\phi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}(\parr)}{\vdash A \parr B, \Gamma}(\cup)$	$\underset{C^{\cup}_{\parr}}{\overset{C^{\parr}_{\cup}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma} \quad \overset{\phi}{\vdash A, B, \Gamma}}{\vdash A, B, \Gamma}(\cup)}{\vdash A \parr B, \Gamma}(\parr)$
$\parr - \emptyset$	$\dfrac{}{\vdash A \parr B, \Gamma}(\emptyset)$	$\underset{C^{\emptyset}_{\parr}}{\overset{C^{\parr}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash A, B, \Gamma}(\emptyset)}{\vdash A \parr B, \Gamma}(\parr)$
$\otimes - \otimes - 1$	$\dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, D, \Delta} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, D, \Delta, \Sigma}(\otimes)}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}(\otimes)$	$\underset{C^{\otimes}_{\otimes}}{\overset{C^{\otimes}_{\otimes}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \overset{\phi}{\vdash A, D, \Delta}}{\vdash A, C \otimes D, \Gamma, \Delta}(\otimes) \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}(\otimes)$

Table 1.6: Rule commutation (Part 2/16)

$\otimes - \otimes - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash D, \Sigma}}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$	$\overset{\xrightarrow{c^{\otimes}_{\otimes}}}{\xleftarrow[c^{\otimes}_{\otimes}]{}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\tau}{\vdash D, \Sigma}}{\vdash A, C \otimes D, \Gamma, \Sigma}\,(\otimes) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - \otimes - 3$	$\dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, D, \Sigma}}{\vdash A \otimes B, D, \Gamma, \Sigma}\,(\otimes)}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$	$\overset{\xrightarrow{c^{\otimes}_{\otimes}}}{\xleftarrow[c^{\otimes}_{\otimes}]{}}$	$\dfrac{\overset{\phi}{\vdash A, \Delta} \quad \dfrac{\overset{\pi}{\vdash C, \Gamma} \quad \overset{\tau}{\vdash B, D, \Sigma}}{\vdash B, C \otimes D, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - \bot - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \bot, \Gamma, \Delta}\,(\bot)$	$\overset{\xrightarrow{c^{\otimes}_{\bot}}}{\xleftarrow[c^{\bot}_{\otimes}]{}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \bot, \Gamma, \Delta}\,(\otimes)$
$\otimes - \bot - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \bot, \Gamma, \Delta}\,(\bot)$	$\overset{\xrightarrow{c^{\otimes}_{\bot}}}{\xleftarrow[c^{\bot}_{\otimes}]{}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, \Delta}}{\vdash B, \bot, \Delta}\,(\bot)}{\vdash A \otimes B, \bot, \Gamma, \Delta}\,(\otimes)$
$\otimes - \& - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, C, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes) \quad \dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash B, D, \Delta}}{\vdash A \otimes B, D, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \,\&\, D, \Gamma, \Delta}\,(\&)$	$\overset{\xrightarrow{c^{\otimes}_{\&}}}{\xleftarrow[c^{\&}_{\otimes}]{}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, C, \Delta} \quad \overset{\tau}{\vdash B, D, \Delta}}{\vdash B, C \,\&\, D, \Delta}\,(\&)}{\vdash A \otimes B, C \,\&\, D, \Gamma, \Delta}\,(\otimes)$
$\otimes - \& - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes) \quad \dfrac{\overset{\tau}{\vdash A, D, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, D, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \,\&\, D, \Gamma, \Delta}\,(\&)$	$\overset{\xrightarrow{c^{\otimes}_{\&}}}{\xleftarrow[c^{\&}_{\otimes}]{}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\tau}{\vdash A, D, \Gamma}}{\vdash A, C \,\&\, D, \Gamma}\,(\&) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \,\&\, D, \Gamma, \Delta}\,(\otimes)$
$\otimes - \oplus_1 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\oplus_1)$	$\overset{\xrightarrow{c^{\otimes}_{\oplus_1}}}{\xleftarrow[c^{\oplus_1}_{\otimes}]{}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_1) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\otimes)$
$\otimes - \oplus_1 - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, C, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\oplus_1)$	$\overset{\xrightarrow{c^{\otimes}_{\oplus_1}}}{\xleftarrow[c^{\oplus_1}_{\otimes}]{}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, C, \Delta}}{\vdash B, C \oplus D, \Delta}\,(\oplus_1)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\otimes)$
$\otimes - \oplus_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, D, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\oplus_2)$	$\overset{\xrightarrow{c^{\otimes}_{\oplus_2}}}{\xleftarrow[c^{\oplus_2}_{\otimes}]{}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_2) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\otimes)$
$\otimes - \oplus_2 - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, D, \Delta}}{\vdash A \otimes B, D, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\oplus_2)$	$\overset{\xrightarrow{c^{\otimes}_{\oplus_2}}}{\xleftarrow[c^{\oplus_2}_{\otimes}]{}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, D, \Delta}}{\vdash B, C \oplus D, \Delta}\,(\oplus_2)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\,(\otimes)$

Table 1.7: Rule commutation (Part 3/16)

$\otimes - \top - 1$	$\dfrac{}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\top)$	$\xrightarrow{c^{\otimes}_{\top}}$ $\xleftarrow[c^{\top}_{\otimes}]{}$	$\dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \overset{\pi}{\vdash B, \Delta}}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)$
$\otimes - \top - 2$	$\dfrac{}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\top)$	$\xrightarrow{c^{\otimes}_{\top}}$ $\xleftarrow[c^{\top}_{\otimes}]{}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{}{\vdash B, \top, \Delta}\,(\top)}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_d - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_d)$	$\xrightarrow{c^{\otimes}_{?_d}}$ $\xleftarrow[c^{?_d}_{\otimes}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_d) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_d - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, C, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_d)$	$\xrightarrow{c^{\otimes}_{?_d}}$ $\xleftarrow[c^{?_d}_{\otimes}]{}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash B, C, \Delta}}{\vdash B, ?C, \Delta}\,(?_d)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_c - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, ?C, ?C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_c)$	$\xrightarrow{c^{\otimes}_{?_c}}$ $\xleftarrow[c^{?_c}_{\otimes}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_c) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_c - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, ?C, ?C, \Delta}}{\vdash A \otimes B, ?C, ?C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_c)$	$\xrightarrow{c^{\otimes}_{?_c}}$ $\xleftarrow[c^{?_c}_{\otimes}]{}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash B, ?C, ?C, \Delta}}{\vdash B, ?C, \Delta}\,(?_c)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_w - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_w)$	$\xrightarrow{c^{\otimes}_{?_w}}$ $\xleftarrow[c^{?_w}_{\otimes}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_w) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - ?_w - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(?_w)$	$\xrightarrow{c^{\otimes}_{?_w}}$ $\xleftarrow[c^{?_w}_{\otimes}]{}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash B, \Delta}}{\vdash B, ?C, \Delta}\,(?_w)}{\vdash A \otimes B, ?C, \Gamma, \Delta}\,(\otimes)$
$\otimes - \forall - 1$	X not free in $A \otimes B, \Gamma, \Delta$ $\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \forall X C, \Gamma, \Delta}\,(\forall)$	$\xrightarrow{c^{\otimes}_{\forall}}$ $\xleftarrow[c^{\forall}_{\otimes}]{}$	X not free in A, Γ $\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, \forall X C, \Gamma}\,(\forall) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \forall X C, \Gamma, \Delta}\,(\otimes)$
$\otimes - \forall - 2$	X not free in $A \otimes B, \Gamma, \Delta$ $\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, C, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \forall X C, \Gamma, \Delta}\,(\forall)$	$\xrightarrow{c^{\otimes}_{\forall}}$ $\xleftarrow[c^{\forall}_{\otimes}]{}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad X \text{ not free in } B, \Delta \;\; \dfrac{\overset{\phi}{\vdash B, C, \Delta}}{\vdash B, \forall X C, \Delta}\,(\forall)}{\vdash A \otimes B, \forall X C, \Gamma, \Delta}\,(\otimes)$

Table 1.8: Rule commutation (Part 4/16)

$\otimes - \exists - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C[D/X], \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \exists XC, \Gamma, \Delta}\,(\exists)$	$\underset{c^{\exists}_{\otimes}}{\overset{c^{\otimes}_{\exists}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C[D/X], \Gamma}}{\vdash A, \exists XC, \Gamma}\,(\exists) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \exists XC, \Gamma, \Delta}\,(\otimes)$
$\otimes - \exists - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, C[D/X], \Delta}}{\vdash A \otimes B, C[D/X], \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \exists XC, \Gamma, \Delta}\,(\exists)$	$\underset{c^{\exists}_{\otimes}}{\overset{c^{\otimes}_{\exists}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, C[D/X], \Delta}}{\vdash B, \exists XC, \Delta}\,(\exists)}{\vdash A \otimes B, \exists XC, \Gamma, \Delta}\,(\otimes)$
$\otimes - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2)$	$\underset{c^{mix_2}_{\otimes}}{\overset{c^{\otimes}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash \Sigma}}{\vdash A, \Gamma, \Sigma}\,(mix_2) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2)$	$\underset{c^{mix_2}_{\otimes}}{\overset{c^{\otimes}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - mix_2 - 3$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2)$	$\underset{c^{mix_2}_{\otimes}}{\overset{c^{\otimes}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash B, \Delta, \Sigma}\,(mix_2)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - mix_2 - 4$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2)$	$\underset{c^{mix_2}_{\otimes}}{\overset{c^{\otimes}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\overset{\phi}{\vdash A, \Delta} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash B, \Gamma, \Sigma}\,(mix_2)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$
$\otimes - \cup - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \quad \dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta}\,(\cup)$	$\underset{c^{\cup}_{\otimes}}{\overset{c^{\otimes}_{\cup}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, \Delta} \quad \overset{\tau}{\vdash B, \Delta}}{\vdash B, \Delta}\,(\cup)}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)$
$\otimes - \cup - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \quad \dfrac{\overset{\tau}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta}\,(\cup)$	$\underset{c^{\cup}_{\otimes}}{\overset{c^{\otimes}_{\cup}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash A, \Gamma}}{\vdash A, \Gamma}\,(\cup) \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)$
$\otimes - \emptyset - 1$	$\dfrac{}{\vdash A \otimes B, \Gamma, \Delta}\,(\emptyset)$	$\underset{c^{\emptyset}_{\otimes}}{\overset{c^{\otimes}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash A, \Gamma}\,(\emptyset) \quad \overset{\pi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)$

Table 1.9: Rule commutation (Part 5/16)

$\otimes - \emptyset - 2$	$\dfrac{}{\vdash A \otimes B, \Gamma, \Delta}\,(\emptyset)$	$\underset{c^{\emptyset}_{\otimes}}{\overset{c^{\otimes}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{}{\vdash B, \Delta}\,(\emptyset)}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)$
$\bot - \bot$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot_1, \Gamma}\,(\bot_1)}{\vdash \bot_1, \bot_2, \Gamma}\,(\bot_2)$	$\xrightarrow{c^{\bot}_{\bot}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot_2, \Gamma}\,(\bot_2)}{\vdash \bot_1, \bot_2, \Gamma}\,(\bot_2)$
$\bot - \&$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot) \quad \dfrac{\overset{\phi}{\vdash B, \Gamma}}{\vdash B, \bot, \Gamma}\,(\bot)}{\vdash A \& B, \bot, \Gamma}\,(\&)$	$\underset{c^{\&}_{\bot}}{\overset{c^{\bot}_{\&}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Gamma}}{\vdash A \& B, \Gamma}\,(\&)}{\vdash A \& B, \bot, \Gamma}\,(\bot)$
$\bot - \oplus_1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(\oplus_1)$	$\underset{c^{\oplus_1}_{\bot}}{\overset{c^{\bot}_{\oplus_1}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, \bot, \Gamma}\,(\bot)$
$\bot - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash B, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(\oplus_2)$	$\underset{c^{\oplus_2}_{\bot}}{\overset{c^{\bot}_{\oplus_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_2)}{\vdash A \oplus B, \bot, \Gamma}\,(\bot)$
$\bot - \top$	$\dfrac{}{\vdash \top, \bot, \Gamma}\,(\top)$	$\underset{c^{\top}_{\bot}}{\overset{c^{\bot}_{\top}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash \top, \Gamma}\,(\top)}{\vdash \top, \bot, \Gamma}\,(\bot)$
$\bot - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash \bot, A, \Gamma}\,(\bot)}{\vdash \bot, ?A, \Gamma}\,(?_d)$	$\underset{c^{?_d}_{\bot}}{\overset{c^{\bot}_{?_d}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash ?A, \Gamma}\,(?_d)}{\vdash \bot, ?A, \Gamma}\,(\bot)$
$\bot - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash \bot, ?A, ?A, \Gamma}\,(\bot)}{\vdash \bot, ?A, \Gamma}\,(?_c)$	$\underset{c^{?_c}_{\bot}}{\overset{c^{\bot}_{?_c}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, \Gamma}\,(?_c)}{\vdash \bot, ?A, \Gamma}\,(\bot)$
$\bot - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot)}{\vdash \bot, ?A, \Gamma}\,(?_w)$	$\underset{c^{?_w}_{\bot}}{\overset{c^{\bot}_{?_w}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash ?A, \Gamma}\,(?_w)}{\vdash \bot, ?A, \Gamma}\,(\bot)$

Table 1.10: Rule commutation (Part 6/16)

$\bot - \forall$	$X \text{ not free in } \bot,\Gamma \quad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, A, \Gamma}\,(\bot)}{\vdash \bot, \forall X A, \Gamma}\,(\forall)$	$\overset{c^{\bot}_{\forall}}{\longrightarrow}$ $\underset{c^{\forall}_{\bot}}{\longleftarrow}$	$\dfrac{X \text{ not free in } \Gamma \quad \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash \forall X A, \Gamma}\,(\forall)}{\vdash \bot, \forall X A, \Gamma}\,(\bot)$
$\bot - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma}}{\vdash \bot, A[B/X], \Gamma}\,(\bot)}{\vdash \bot, \exists X A, \Gamma}\,(\exists)$	$\overset{c^{\bot}_{\exists}}{\longrightarrow}$ $\underset{c^{\exists}_{\bot}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma}}{\vdash \exists X A, \Gamma}\,(\exists)}{\vdash \bot, \exists X A, \Gamma}\,(\bot)$
$\bot - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot) \quad \overset{\phi}{\vdash \Delta}}{\vdash \bot, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{\bot}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{\bot}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,(mix_2)}{\vdash \bot, \Gamma, \Delta}\,(\bot)$
$\bot - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash \bot, \Delta}\,(\bot)}{\vdash \bot, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{\bot}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{\bot}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,(mix_2)}{\vdash \bot, \Gamma, \Delta}\,(\bot)$
$\bot - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot) \quad \dfrac{\overset{\phi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot)}{\vdash \bot, \Gamma}\,(\cup)$	$\overset{c^{\bot}_{\cup}}{\longrightarrow}$ $\underset{c^{\cup}_{\bot}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Gamma}}{\vdash \Gamma}\,(\cup)}{\vdash \bot, \Gamma}\,(\bot)$
$\bot - \emptyset$	$\dfrac{}{\vdash \bot, \Gamma}\,(\emptyset)$	$\overset{c^{\bot}_{\emptyset}}{\longrightarrow}$ $\underset{c^{\emptyset}_{\bot}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash \Gamma}\,(\emptyset)}{\vdash \bot, \Gamma}\,(\bot)$
$\& - \&$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&) \quad \dfrac{\overset{\tau}{\vdash A, D, \Gamma} \quad \overset{\mu}{\vdash B, D, \Gamma}}{\vdash A \mathbin{\&} B, D, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, C \mathbin{\&} D, \Gamma}\,(\&)$	$\overset{c^{\&}_{\&}}{\longrightarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\tau}{\vdash A, D, \Gamma}}{\vdash A, C \mathbin{\&} D, \Gamma}\,(\&) \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma} \quad \overset{\mu}{\vdash B, D, \Gamma}}{\vdash B, C \mathbin{\&} D, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, C \mathbin{\&} D, \Gamma}\,(\&)$
$\& - \oplus_1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, C \oplus D, \Gamma}\,(\oplus_1)$	$\overset{c^{\&}_{\oplus_1}}{\longrightarrow}$ $\underset{c^{\oplus_1}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_1) \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma}}{\vdash B, C \oplus D, \Gamma}\,(\oplus_1)}{\vdash A \mathbin{\&} B, C \oplus D, \Gamma}\,(\&)$
$\& - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma} \quad \overset{\phi}{\vdash B, D, \Gamma}}{\vdash A \mathbin{\&} B, D, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, C \oplus D, \Gamma}\,(\oplus_2)$	$\overset{c^{\&}_{\oplus_2}}{\longrightarrow}$ $\underset{c^{\oplus_2}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_2) \quad \dfrac{\overset{\phi}{\vdash B, D, \Gamma}}{\vdash B, C \oplus D, \Gamma}\,(\oplus_2)}{\vdash A \mathbin{\&} B, C \oplus D, \Gamma}\,(\&)$
$\& - \top$	$\dfrac{}{\vdash A \mathbin{\&} B, \top, \Gamma}\,(\top)$	$\overset{c^{\&}_{\top}}{\longrightarrow}$ $\underset{c^{\top}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \quad \dfrac{}{\vdash B, \top, \Gamma}\,(\top)}{\vdash A \mathbin{\&} B, \top, \Gamma}\,(\&)$
$\& - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, ?C, \Gamma}\,(?_d)$	$\overset{c^{\&}_{?_d}}{\longrightarrow}$ $\underset{c^{?_d}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_d) \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma}}{\vdash B, ?C, \Gamma}\,(?_d)}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)$
$\& - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma} \quad \overset{\phi}{\vdash B, ?C, ?C, \Gamma}}{\vdash A \mathbin{\&} B, ?C, ?C, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, ?C, \Gamma}\,(?_c)$	$\overset{c^{\&}_{?_c}}{\longrightarrow}$ $\underset{c^{?_c}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_c) \quad \dfrac{\overset{\phi}{\vdash B, ?C, ?C, \Gamma}}{\vdash B, ?C, \Gamma}\,(?_c)}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)$
$\& - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Gamma}}{\vdash A \mathbin{\&} B, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, ?C, \Gamma}\,(?_w)$	$\overset{c^{\&}_{?_w}}{\longrightarrow}$ $\underset{c^{?_w}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, ?C, \Gamma}\,(?_w) \quad \dfrac{\overset{\phi}{\vdash B, \Gamma}}{\vdash B, ?C, \Gamma}\,(?_w)}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)$
$\& - \forall$	$X \text{ not free in } A\&B,\Gamma \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \mathbin{\&} B, C, \Gamma}\,(\&)}{\vdash A \mathbin{\&} B, \forall X C, \Gamma}\,(\forall)$	$\overset{c^{\&}_{\forall}}{\longrightarrow}$ $\underset{c^{\forall}_{\&}}{\longleftarrow}$	$\dfrac{X \text{ not free in } A,\Gamma \quad \dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, \forall X C, \Gamma}\,(\forall) \quad X \text{ not free in } B,\Gamma \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma}}{\vdash B, \forall X C, \Gamma}\,(\forall)}{\vdash A \mathbin{\&} B, \forall X C, \Gamma}\,(\&)$

Table 1.11: Rule commutation (Part 7/16)

$\& - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C[D/X], \Gamma} \quad \overset{\phi}{\vdash B, C[D/X], \Gamma}}{\vdash A \& B, C[D/X], \Gamma}(\&)}{\vdash A \& B, \exists X C, \Gamma}(\exists)$	$\overset{C^{\&}_{\exists}}{\longrightarrow} \underset{C^{\exists}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C[D/X], \Gamma}}{\vdash A, \exists X C, \Gamma}(\exists) \quad \dfrac{\overset{\phi}{\vdash B, C[D/X], \Gamma}}{\vdash B, \exists X C, \Gamma}(\exists)}{\vdash A \& B, \exists X C, \Gamma}(\&)$
$\& - mix_2 - 1$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, \Delta}}{\vdash A \& B, \Delta}(\&)}{\vdash A \& B, \Gamma, \Delta}(mix_2)$	$\overset{C^{\&}_{mix_2}}{\longrightarrow} \underset{C^{mix_2}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}(mix_2) \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash B, \Delta}}{\vdash B, \Gamma, \Delta}(mix_2)}{\vdash A \& B, \Gamma, \Delta}(\&)$
$\& - mix_2 - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash B, \Gamma}}{\vdash A \& B, \Gamma}(\&) \quad \overset{\phi}{\vdash \Delta}}{\vdash A \& B, \Gamma, \Delta}(mix_2)$	$\overset{C^{\&}_{mix_2}}{\longrightarrow} \underset{C^{mix_2}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}(mix_2) \quad \dfrac{\overset{\tau}{\vdash B, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash B, \Gamma, \Delta}(mix_2)}{\vdash A \& B, \Gamma, \Delta}(\&)$
$\& - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Gamma}}{\vdash A \& B, \Gamma}(\&) \quad \dfrac{\overset{\tau}{\vdash A, \Gamma} \quad \overset{\mu}{\vdash B, \Gamma}}{\vdash A \& B, \Gamma}(\&)}{\vdash A \& B, \Gamma}(\cup)$	$\overset{C^{\&}_{\cup}}{\longrightarrow} \underset{C^{\cup}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\tau}{\vdash A, \Gamma}}{\vdash A, \Gamma}(\cup) \quad \dfrac{\overset{\phi}{\vdash B, \Gamma} \quad \overset{\mu}{\vdash B, \Gamma}}{\vdash B, \Gamma}(\cup)}{\vdash A \& B, \Gamma}(\&)$
$\& - \emptyset$	$\dfrac{}{\vdash A \& B, \Gamma}(\emptyset)$	$\overset{C^{\&}_{\emptyset}}{\longrightarrow} \underset{C^{\emptyset}_{\&}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash A, \Gamma}(\emptyset) \quad \dfrac{}{\vdash B, \Gamma}(\emptyset)}{\vdash A \& B, \Gamma}(\&)$
$\oplus_1 - \oplus_1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A \oplus B, C, \Gamma}(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}(\oplus_1)$	$\overset{C^{\oplus_1}_{\oplus_1}}{\longrightarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}(\oplus_1)$
$\oplus_1 - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma}}{\vdash A \oplus B, D, \Gamma}(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}(\oplus_2)$	$\overset{C^{\oplus_1}_{\oplus_2}}{\longrightarrow} \underset{C^{\oplus_2}_{\oplus_1}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, D, \Gamma}}{\vdash A, C \oplus D, \Gamma}(\oplus_2)}{\vdash A \oplus B, C \oplus D, \Gamma}(\oplus_1)$
$\oplus_1 - \top$	$\dfrac{}{\vdash A \oplus B, \top, \Gamma}(\top)$	$\overset{C^{\oplus_1}_{\top}}{\longrightarrow} \underset{C^{\top}_{\oplus_1}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash A, \top, \Gamma}(\top)}{\vdash A \oplus B, \top, \Gamma}(\oplus_1)$
$\oplus_1 - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A \oplus B, C, \Gamma}(\oplus_1)}{\vdash A \oplus B, ?C, \Gamma}(?_d)$	$\overset{C^{\oplus_1}_{?_d}}{\longrightarrow} \underset{C^{?_d}_{\oplus_1}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, ?C, \Gamma}(?_d)}{\vdash A \oplus B, ?C, \Gamma}(\oplus_1)$
$\oplus_1 - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma}}{\vdash A \oplus B, ?C, ?C, \Gamma}(\oplus_1)}{\vdash A \oplus B, ?C, \Gamma}(?_c)$	$\overset{C^{\oplus_1}_{?_c}}{\longrightarrow} \underset{C^{?_c}_{\oplus_1}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?C, ?C, \Gamma}}{\vdash A, ?C, \Gamma}(?_c)}{\vdash A \oplus B, ?C, \Gamma}(\oplus_1)$
$\oplus_1 - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}(\oplus_1)}{\vdash A \oplus B, ?C, \Gamma}(?_w)$	$\overset{C^{\oplus_1}_{?_w}}{\longrightarrow} \underset{C^{?_w}_{\oplus_1}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, ?C, \Gamma}(?_w)}{\vdash A \oplus B, ?C, \Gamma}(\oplus_1)$

Table 1.12: Rule commutation (Part 8/16)

$\oplus_1 - \forall$	X not free in $A\oplus B,\Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A\oplus B, C,\Gamma}\,(\oplus_1)}{\vdash A\oplus B,\forall XC,\Gamma}\,(\forall)$	$\xrightarrow{C^{\oplus_1}_{\forall}}$ $\xleftarrow[C^{\forall}_{\oplus_1}]{}$	X not free in A,Γ $\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A,\forall XC,\Gamma}\,(\forall)}{\vdash A\oplus B,\forall XC,\Gamma}\,(\oplus_1)$
$\oplus_1 - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C[D/X],\Gamma}}{\vdash A\oplus B, C[D/X],\Gamma}\,(\oplus_1)}{\vdash A\oplus B,\exists XC,\Gamma}\,(\exists)$	$\xrightarrow{C^{\oplus_1}_{\exists}}$ $\xleftarrow[C^{\exists}_{\oplus_1}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, C[D/X],\Gamma}}{\vdash A,\exists XC,\Gamma}\,(\exists)}{\vdash A\oplus B,\exists XC,\Gamma}\,(\oplus_1)$
$\oplus_1 - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A\oplus B,\Gamma}\,(\oplus_1) \quad \overset{\phi}{\vdash \Delta}}{\vdash A\oplus B,\Gamma,\Delta}\,(mix_2)$	$\xrightarrow{C^{\oplus_1}_{mix_2}}$ $\xleftarrow[C^{mix_2}_{\oplus_1}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A,\Gamma,\Delta}\,(mix_2)}{\vdash A\oplus B,\Gamma,\Delta}\,(\oplus_1)$
$\oplus_1 - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A,\Delta}}{\vdash A\oplus B,\Delta}\,(\oplus_1)}{\vdash A\oplus B,\Gamma,\Delta}\,(mix_2)$	$\xrightarrow{C^{\oplus_1}_{mix_2}}$ $\xleftarrow[C^{mix_2}_{\oplus_1}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A,\Delta}}{\vdash A,\Gamma,\Delta}\,(mix_2)}{\vdash A\oplus B,\Gamma,\Delta}\,(\oplus_1)$
$\oplus_1 - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A\oplus B,\Gamma}\,(\oplus_1) \quad \dfrac{\overset{\phi}{\vdash A,\Gamma}}{\vdash A\oplus B,\Gamma}\,(\oplus_1)}{\vdash A\oplus B,\Gamma}\,(\cup)$	$\xrightarrow{C^{\oplus_1}_{\cup}}$ $\xleftarrow[C^{\cup}_{\oplus_1}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma} \quad \overset{\phi}{\vdash A,\Gamma}}{\vdash A,\Gamma}\,(\cup)}{\vdash A\oplus B,\Gamma}\,(\oplus_1)$
$\oplus_1 - \emptyset$	$\dfrac{}{\vdash A\oplus B,\Gamma}\,(\emptyset)$	$\xrightarrow{C^{\oplus_1}_{\emptyset}}$ $\xleftarrow[C^{\emptyset}_{\oplus_1}]{}$	$\dfrac{\dfrac{}{\vdash A,\Gamma}\,(\emptyset)}{\vdash A\oplus B,\Gamma}\,(\oplus_1)$
$\oplus_2 - \oplus_2$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, D,\Gamma}}{\vdash A\oplus B, D,\Gamma}\,(\oplus_2)}{\vdash A\oplus B, C\oplus D,\Gamma}\,(\oplus_2)$	$\xrightarrow{C^{\oplus_2}_{\oplus_2}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, D,\Gamma}}{\vdash B, C\oplus D,\Gamma}\,(\oplus_2)}{\vdash A\oplus B, C\oplus D,\Gamma}\,(\oplus_2)$
$\oplus_2 - \top$	$\dfrac{}{\vdash A\oplus B,\top,\Gamma}\,(\top)$	$\xrightarrow{C^{\oplus_2}_{\top}}$ $\xleftarrow[C^{\top}_{\oplus_2}]{}$	$\dfrac{\dfrac{}{\vdash B,\top,\Gamma}\,(\top)}{\vdash A\oplus B,\top,\Gamma}\,(\oplus_2)$
$\oplus_2 - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, C,\Gamma}}{\vdash A\oplus B, C,\Gamma}\,(\oplus_2)}{\vdash A\oplus B, ?C,\Gamma}\,(?_d)$	$\xrightarrow{C^{\oplus_2}_{?_d}}$ $\xleftarrow[C^{?_d}_{\oplus_2}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, C,\Gamma}}{\vdash B, ?C,\Gamma}\,(?_d)}{\vdash A\oplus B, ?C,\Gamma}\,(\oplus_2)$

Table 1.13: Rule commutation (Part 9/16)

$\oplus_2 - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, ?C, ?C, \Gamma}}{\vdash A \oplus B, ?C, ?C, \Gamma}\,(\oplus_2)}{\vdash A \oplus B, ?C, \Gamma}\,(?_c)$	$\overset{C^{\oplus_2}_{?_c}}{\longrightarrow}$ $\underset{C^{?_c}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, ?C, ?C, \Gamma}}{\vdash B, ?C, \Gamma}\,(?_c)}{\vdash A \oplus B, ?C, \Gamma}\,(\oplus_2)$
$\oplus_2 - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_2)}{\vdash A \oplus B, ?C, \Gamma}\,(?_w)$	$\overset{C^{\oplus_2}_{?_w}}{\longrightarrow}$ $\underset{C^{?_w}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash B, ?C, \Gamma}\,(?_w)}{\vdash A \oplus B, ?C, \Gamma}\,(\oplus_2)$
$\oplus_2 - \forall$	X not free in $A \oplus B, \Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash B, C, \Gamma}}{\vdash A \oplus B, C, \Gamma}\,(\oplus_2)}{\vdash A \oplus B, \forall X C, \Gamma}\,(\forall)$	$\overset{C^{\oplus_2}_{\forall}}{\longrightarrow}$ $\underset{C^{\forall}_{\oplus_2}}{\longleftarrow}$	X not free in B, Γ $\dfrac{\dfrac{\overset{\pi}{\vdash B, C, \Gamma}}{\vdash B, \forall X C, \Gamma}\,(\forall)}{\vdash A \oplus B, \forall X C, \Gamma}\,(\oplus_2)$
$\oplus_2 - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, C[D/X], \Gamma}}{\vdash A \oplus B, C[D/X], \Gamma}\,(\oplus_2)}{\vdash A \oplus B, \exists X C, \Gamma}\,(\exists)$	$\overset{C^{\oplus_2}_{\exists}}{\longrightarrow}$ $\underset{C^{\exists}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, C[D/X], \Gamma}}{\vdash B, \exists X C, \Gamma}\,(\exists)}{\vdash A \oplus B, \exists X C, \Gamma}\,(\oplus_2)$
$\oplus_2 - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_2) \quad \overset{\phi}{\vdash \Delta}}{\vdash A \oplus B, \Gamma, \Delta}\,(mix_2)$	$\overset{C^{\oplus_2}_{mix_2}}{\longrightarrow}$ $\underset{C^{mix_2}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash B, \Gamma, \Delta}\,(mix_2)}{\vdash A \oplus B, \Gamma, \Delta}\,(\oplus_2)$
$\oplus_2 - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash B, \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash A \oplus B, \Delta}\,(\oplus_2)}{\vdash A \oplus B, \Gamma, \Delta}\,(mix_2)$	$\overset{C^{\oplus_2}_{mix_2}}{\longrightarrow}$ $\underset{C^{mix_2}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash B, \Gamma, \Delta}\,(mix_2)}{\vdash A \oplus B, \Gamma, \Delta}\,(\oplus_2)$
$\oplus_2 - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_2) \quad \dfrac{\overset{\phi}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_2)}{\vdash A \oplus B, \Gamma}\,(\cup)$	$\overset{C^{\oplus_2}_{\cup}}{\longrightarrow}$ $\underset{C^{\cup}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma} \quad \overset{\phi}{\vdash B, \Gamma}}{\vdash B, \Gamma}\,(\cup)}{\vdash A \oplus B, \Gamma}\,(\oplus_2)$
$\oplus_2 - \emptyset$	$\dfrac{}{\vdash A \oplus B, \Gamma}\,(\emptyset)$	$\overset{C^{\oplus_2}_{\emptyset}}{\longrightarrow}$ $\underset{C^{\emptyset}_{\oplus_2}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash B, \Gamma}\,(\emptyset)}{\vdash A \oplus B, \Gamma}\,(\oplus_2)$
$\top - \top$	$\dfrac{}{\vdash \top_1, \top_2, \Gamma}\,(\top_1)$	$\overset{C^{\top}_{\top}}{\longrightarrow}$	$\dfrac{}{\vdash \top_1, \top_2, \Gamma}\,(\top_2)$
$\top - ?_d$	$\dfrac{\dfrac{}{\vdash \top, A, \Gamma}\,(\top)}{\vdash \top, ?A, \Gamma}\,(?_d)$	$\overset{C^{\top}_{?_d}}{\longrightarrow}$ $\underset{C^{?_d}_{\top}}{\longleftarrow}$	$\dfrac{}{\vdash \top, ?A, \Gamma}\,(\top)$

Table 1.14: Rule commutation (Part 10/16)

$\top - ?_c$	$\dfrac{\dfrac{}{\vdash \top, ?A, ?A, \Gamma}(\top)}{\vdash \top, ?A, \Gamma}(?_c)$	$\underset{c^{?_c}_{\top}}{\overset{c^{\top}_{?_c}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, ?A, \Gamma}(\top)$
$\top - ?_w$	$\dfrac{\dfrac{}{\vdash \top, \Gamma}(\top)}{\vdash \top, \Gamma}(?_w)$	$\underset{c^{?_w}_{\top}}{\overset{c^{\top}_{?_w}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, ?A, \Gamma}(\top)$
$\top - \forall$	X not free in $\top,\Gamma$ $\dfrac{\dfrac{}{\vdash \top, A, \Gamma}(\top)}{\vdash \top, \forall X A, \Gamma}(\forall)$	$\underset{c^{\forall}_{\top}}{\overset{c^{\top}_{\forall}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \forall X A, \Gamma}(\top)$
$\top - \exists$	$\dfrac{\dfrac{}{\vdash \top, A[B/X], \Gamma}(\top)}{\vdash \top, \exists X A, \Gamma}(\exists)$	$\underset{c^{\exists}_{\top}}{\overset{c^{\top}_{\exists}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \exists X A, \Gamma}(\top)$
$\top - mix_2 - 1$	$\dfrac{\dfrac{}{\vdash \top, \Gamma}(\top) \qquad \overset{\pi}{\vdash \Delta}}{\vdash \top, \Gamma, \Delta}(mix_2)$	$\underset{c^{mix_2}_{\top}}{\overset{c^{\top}_{mix_2}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \Gamma, \Delta}(\top)$
$\top - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{}{\vdash \top, \Delta}(\top)}{\vdash \top, \Gamma, \Delta}(mix_2)$	$\underset{c^{mix_2}_{\top}}{\overset{c^{\top}_{mix_2}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \Gamma, \Delta}(\top)$
$\top - \cup$	$\dfrac{\dfrac{}{\vdash \top, \Gamma}(\top) \qquad \dfrac{}{\vdash \top, \Gamma}(\top)}{\vdash \top, \Gamma}(\cup)$	$\underset{c^{\cup}_{\top}}{\overset{c^{\top}_{\cup}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \Gamma}(\top)$
$\top - \emptyset$	$\dfrac{}{\vdash \top, \Gamma}(\emptyset)$	$\underset{c^{\emptyset}_{\top}}{\overset{c^{\top}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{}{\vdash \top, \Gamma}(\top)$
$?_d - ?_d$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash ?A, B, \Gamma}(?_d)}{\vdash ?A, ?B, \Gamma}(?_d)$	$\xrightarrow{c^{?_d}_{?_d}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, ?B, \Gamma}(?_d)}{\vdash ?A, ?B, \Gamma}(?_d)$
$?_d - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, \Gamma}}{\vdash ?A, ?B, ?B, \Gamma}(?_d)}{\vdash ?A, ?B, \Gamma}(?_c)$	$\underset{c^{?_c}_{?_d}}{\overset{c^{?_d}_{?_c}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, \Gamma}}{\vdash A, ?B, \Gamma}(?_c)}{\vdash ?A, ?B, \Gamma}(?_d)$
$?_d - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash ?A, \Gamma}(?_d)}{\vdash ?A, ?B, \Gamma}(?_w)$	$\underset{c^{?_w}_{?_d}}{\overset{c^{?_d}_{?_w}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, ?B, \Gamma}(?_w)}{\vdash ?A, ?B, \Gamma}(?_d)$

Table 1.15: Rule commutation (Part 11/16)

$?_d - \forall$	X not free in $?A,\Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash ?A, B, \Gamma}\,(?_d)}{\vdash ?A, \forall X B, \Gamma}\,(\forall)$	$\xrightarrow{C^{?_d}_{\forall}}$ $\xleftarrow[C^{\forall}_{?_d}]{}$	X not free in A,Γ $\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, \forall X B, \Gamma}\,(\forall)}{\vdash ?A, \forall X B, \Gamma}\,(?_d)$
$?_d - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B[C/X], \Gamma}}{\vdash A, B[C/X], \Gamma}\,(?_d)}{\vdash ?A, \exists X B, \Gamma}\,(\exists)$	$\xrightarrow{C^{?_d}_{\exists}}$ $\xleftarrow[C^{\exists}_{?_d}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, B[C/X], \Gamma}}{\vdash A, \exists X B, \Gamma}\,(\exists)}{\vdash ?A, \exists X B, \Gamma}\,(?_d)$
$?_d - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash ?A, \Gamma}\,(?_d) \qquad \overset{\phi}{\vdash \Delta}}{\vdash ?A, \Gamma, \Delta}\,(mix_2)$	$\xrightarrow{C^{?_d}_{mix_2}}$ $\xleftarrow[C^{mix_2}_{?_d}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash ?A, \Gamma, \Delta}\,(?_d)$
$?_d - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, \Delta}}{\vdash ?A, \Delta}\,(?_d)}{\vdash ?A, \Gamma, \Delta}\,(mix_2)$	$\xrightarrow{C^{?_d}_{mix_2}}$ $\xleftarrow[C^{mix_2}_{?_d}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash ?A, \Gamma, \Delta}\,(?_d)$
$?_d - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash ?A, \Gamma}\,(?_d) \qquad \dfrac{\overset{\phi}{\vdash A, \Gamma}}{\vdash ?A, \Gamma}\,(?_d)}{\vdash ?A, \Gamma}\,(\cup)$	$\xrightarrow{C^{?_d}_{\cup}}$ $\xleftarrow[C^{\cup}_{?_d}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash A, \Gamma}}{\vdash A, \Gamma}\,(\cup)}{\vdash ?A, \Gamma}\,(?_d)$
$?_d - \emptyset$	$\dfrac{}{\vdash ?A, \Gamma}\,(\emptyset)$	$\xrightarrow{C^{?_d}_{\emptyset}}$ $\xleftarrow[C^{\emptyset}_{?_d}]{}$	$\dfrac{\dfrac{}{\vdash A, \Gamma}\,(\emptyset)}{\vdash ?A, \Gamma}\,(?_d)$
$?_c - ?_c$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, ?B, ?B, \Gamma}}{\vdash ?A, ?B, ?B, \Gamma}\,(?_c)}{\vdash ?A, ?B, \Gamma}\,(?_c)$	$\xrightarrow{C^{?_c}_{?_c}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, ?B, ?B, \Gamma}}{\vdash ?A, ?A, ?B, \Gamma}\,(?_c)}{\vdash ?A, ?B, \Gamma}\,(?_c)$
$?_c - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, \Gamma}\,(?_c)}{\vdash ?A, ?B, \Gamma}\,(?_w)$	$\xrightarrow{C^{?_c}_{?_w}}$ $\xleftarrow[C^{?_w}_{?_c}]{}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, ?A, ?B, \Gamma}\,(?_w)}{\vdash ?A, ?B, \Gamma}\,(?_c)$
$?_c - \forall$	X not free in $?A,\Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, B, \Gamma}}{\vdash ?A, B, \Gamma}\,(?_c)}{\vdash ?A, \forall X B, \Gamma}\,(\forall)$	$\xrightarrow{C^{?_c}_{\forall}}$ $\xleftarrow[C^{\forall}_{?_c}]{}$	X not free in $?A,?A,\Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, B, \Gamma}}{\vdash ?A, ?A, \forall X B, \Gamma}\,(\forall)}{\vdash ?A, \forall X B, \Gamma}\,(?_c)$

Table 1.16: Rule commutation (Part 12/16)

$?_c - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, B[C/X], \Gamma}}{\vdash ?A, ?A, B[C/X], \Gamma}(?_d)}{\vdash ?A, \exists XB, \Gamma}(\exists)$	$\underset{C^{\exists}_{?_c}}{\overset{C^{?_c}_{\exists}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, B[C/X], \Gamma}}{\vdash ?A, ?A, \exists XB, \Gamma}(\exists)}{\vdash ?A, \exists XB, \Gamma}(?_c)$
$?_c - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, \Gamma}(?_c) \quad \overset{\phi}{\vdash \Delta}}{\vdash ?A, \Gamma, \Delta}(mix_2)$	$\underset{C^{mix_2}_{?_c}}{\overset{C^{?_c}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash ?A, ?A, \Gamma, \Delta}(mix_2)}{\vdash ?A, \Gamma, \Delta}(?_c)$
$?_c - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash ?A, ?A, \Delta}}{\vdash ?A, \Delta}(?_c)}{\vdash ?A, \Gamma, \Delta}(mix_2)$	$\underset{C^{mix_2}_{?_c}}{\overset{C^{?_c}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash ?A, ?A, \Delta}}{\vdash ?A, ?A, \Gamma, \Delta}(mix_2)}{\vdash ?A, \Gamma, \Delta}(?_c)$
$?_c - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, \Gamma}(?_c) \quad \dfrac{\overset{\phi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, \Gamma}(?_c)}{\vdash ?A, \Gamma}(\cup)$	$\underset{C^{\cup}_{?_c}}{\overset{C^{?_c}_{\cup}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A, ?A, \Gamma} \quad \overset{\phi}{\vdash ?A, ?A, \Gamma}}{\vdash ?A, ?A, \Gamma}(\cup)}{\vdash ?A, \Gamma}(?_c)$
$?_c - \emptyset$	$\dfrac{}{\vdash ?A, \Gamma}(\emptyset)$	$\underset{C^{\emptyset}_{?_c}}{\overset{C^{?_c}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash ?A, ?A, \Gamma}(\emptyset)}{\vdash ?A, \Gamma}(?_c)$
$?_w - ?_w$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash ?A, \Gamma}(?_w)}{\vdash ?A, ?B, \Gamma}(?_w)$	$\xrightarrow{C^{?_c}_{?_c}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash ?B, \Gamma}(?_w)}{\vdash ?A, ?B, \Gamma}(?_w)$
$?_w - \forall$	X not free in $?A, \Gamma$ $\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash ?A, B, \Gamma}(?_w)}{\vdash ?A, \forall XB, \Gamma}(\forall)$	$\underset{C^{\forall}_{?_w}}{\overset{C^{?_w}_{\forall}}{\rightleftarrows}}$	X not free in Γ $\dfrac{\dfrac{\overset{\pi}{\vdash B, \Gamma}}{\vdash \forall XB, \Gamma}(\forall)}{\vdash ?A, \forall XB, \Gamma}(?_w)$
$?_w - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash B[C/X], \Gamma}}{\vdash B[C/X], \Gamma}(?_w)}{\vdash ?A, \exists XB, \Gamma}(\exists)$	$\underset{C^{\exists}_{?_w}}{\overset{C^{?_w}_{\exists}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash B[C/X], \Gamma}}{\vdash \exists XB, \Gamma}(\exists)}{\vdash ?A, \exists XB, \Gamma}(?_w)$
$?_w - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash ?A, \Gamma}(?_w) \quad \overset{\phi}{\vdash \Delta}}{\vdash ?A, \Gamma, \Delta}(mix_2)$	$\underset{C^{mix_2}_{?_w}}{\overset{C^{?_w}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2)}{\vdash ?A, \Gamma, \Delta}(?_w)$

Table 1.17: Rule commutation (Part 13/16)

$?_w - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash ?A, \Delta}\,(?_w)}{\vdash ?A, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{?_w}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{?_w}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,(mix_2)}{\vdash ?A, \Gamma, \Delta}\,(?_w)$
$?_w - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash ?A, \Gamma}\,(?_w) \quad \dfrac{\overset{\phi}{\vdash \Gamma}}{\vdash ?A, \Gamma}\,(?_w)}{\vdash ?A, \Gamma}\,(\cup)$	$\overset{c^{?_w}_{\cup}}{\longrightarrow}$ $\underset{c^{\cup}_{?_w}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Gamma}}{\vdash \Gamma}\,(\cup)}{\vdash ?A, \Gamma}\,(?_w)$
$?_w - \emptyset$	$\dfrac{}{\vdash ?A, \Gamma}\,(\emptyset)$	$\overset{c^{?_w}_{\emptyset}}{\longrightarrow}$ $\underset{c^{\emptyset}_{?_w}}{\longleftarrow}$	$\dfrac{\dfrac{}{\vdash \Gamma}\,(\emptyset)}{\vdash ?A, \Gamma}\,(?_w)$
$\forall - \forall$	$Y \text{ not free in } \forall XA,\Gamma \quad \dfrac{X \text{ not free in } B,\Gamma \quad \dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash \forall XA, B, \Gamma}\,(\forall)}{\vdash \forall XA, \forall YB, \Gamma}\,(\forall)$	$\overset{c^{\forall}_{\forall}}{\longrightarrow}$	$X \text{ not free in } \forall YB,\Gamma \quad \dfrac{Y \text{ not free in } A,\Gamma \quad \dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A, \forall YB, \Gamma}\,(\forall)}{\vdash \forall XA, \forall YB, \Gamma}\,(\forall)$
$\forall - \exists$	$\dfrac{X \text{ not free in } B[C/Y],\Gamma \quad \dfrac{\overset{\pi}{\vdash A, B[C/Y], \Gamma}}{\vdash \forall XA, B[C/Y], \Gamma}\,(\forall)}{\vdash \forall XA, \exists YB, \Gamma}\,(\exists)$	$\overset{c^{\forall}_{\exists}}{\longrightarrow}$ $\underset{c^{\exists}_{\forall}}{\longleftarrow}$	$X \text{ not free in } \exists YB,\Gamma \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, B[C/Y], \Gamma}}{\vdash A, \exists YB, \Gamma}\,(\exists)}{\vdash \forall XA, \exists YB, \Gamma}\,(\forall)$
$\forall - mix_2 - 1$	$\dfrac{X \text{ not free in } \Gamma \quad \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash \forall XA, \Gamma}\,(\forall) \quad \overset{\phi}{\vdash \Delta}}{\vdash \forall XA, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{\forall}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{\forall}}{\longleftarrow}$	$X \text{ not free in } \Gamma,\Delta \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash \forall XA, \Gamma, \Delta}\,(\forall)$
$\forall - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad X \text{ not free in } \Delta \quad \dfrac{\overset{\phi}{\vdash A, \Delta}}{\vdash \forall XA, \Delta}\,(\forall)}{\vdash \forall XA, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{\forall}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{\forall}}{\longleftarrow}$	$X \text{ not free in } \Gamma,\Delta \quad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash \forall XA, \Gamma, \Delta}\,(\forall)$
$\forall - \cup$	$\dfrac{X \text{ not free in } \Gamma \quad \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash \forall XA, \Gamma}\,(\forall) \quad X \text{ not free in } \Gamma \quad \dfrac{\overset{\phi}{\vdash A, \Gamma}}{\vdash \forall XA, \Gamma}\,(\forall)}{\vdash \forall XA, \Gamma}\,(\cup)$	$\overset{c^{\forall}_{\cup}}{\longrightarrow}$ $\underset{c^{\cup}_{\forall}}{\longleftarrow}$	$X \text{ not free in } \Gamma \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash A, \Gamma}}{\vdash A, \Gamma}\,(\cup)}{\vdash \forall XA, \Gamma}\,(\forall)$
$\forall - \emptyset$	$\dfrac{}{\vdash \forall XA, \Gamma}\,(\emptyset)$	$\overset{c^{\forall}_{\emptyset}}{\longrightarrow}$ $\underset{c^{\emptyset}_{\forall}}{\longleftarrow}$	$X \text{ not free in } \Gamma \quad \dfrac{\dfrac{}{\vdash A, \Gamma}\,(\emptyset)}{\vdash \forall XA, \Gamma}\,(\forall)$
$\exists - \exists$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[C/X], B[D/Y], \Gamma}}{\vdash \exists XA, B[D/Y], \Gamma}\,(\exists)}{\vdash \exists XA, \exists YB, \Gamma}\,(\exists)$	$\overset{c^{\exists}_{\exists}}{\longrightarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[C/X], B[D/Y], \Gamma}}{\vdash A[C/X], \exists YB, \Gamma}\,(\exists)}{\vdash \exists XA, \exists YB, \Gamma}\,(\exists)$
$\exists - mix_2 - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma}}{\vdash \exists XA, \Gamma}\,(\exists) \quad \overset{\phi}{\vdash \Delta}}{\vdash \exists XA, \Gamma, \Delta}\,(mix_2)$	$\overset{c^{\exists}_{mix_2}}{\longrightarrow}$ $\underset{c^{mix_2}_{\exists}}{\longleftarrow}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A[B/X], \Gamma, \Delta}\,(mix_2)}{\vdash \exists XA, \Gamma, \Delta}\,(\exists)$

Table 1.18: Rule commutation (Part 14/16)

$\exists - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A[B/X], \Delta}}{\vdash \exists X A, \Delta}(\exists)}{\vdash \exists X A, \Gamma, \Delta}(mix_2)$	$\underset{c^{mix_2}_{\exists}}{\overset{c^{\exists}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A[B/X], \Delta}}{\vdash A[B/X], \Gamma, \Delta}(mix_2)}{\vdash \exists X A, \Gamma, \Delta}(\exists)$
$\exists - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma}}{\vdash \exists X A, \Gamma}(\exists) \quad \dfrac{\overset{\phi}{\vdash A[B/X], \Gamma}}{\vdash \exists X A, \Gamma}(\exists)}{\vdash \exists X A, \Gamma}(\cup)$	$\underset{c^{\cup}_{\exists}}{\overset{c^{\exists}_{\cup}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash A[B/X], \Gamma} \quad \overset{\phi}{\vdash A[B/X], \Gamma}}{\vdash A[B/X], \Gamma}(\cup)}{\vdash \exists X A, \Gamma}(\exists)$
$\exists - \emptyset$	$\dfrac{}{\vdash \exists X A, \Gamma}(\emptyset)$	$\underset{c^{\emptyset}_{\exists}}{\overset{c^{\exists}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash A[B/X], \Gamma}(\emptyset)}{\vdash \exists X A, \Gamma}(\exists)$
$mix_2 - mix_2 - 1$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Delta, \Sigma}(mix_2)}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$	$\underset{c^{mix_2}_{mix_2}}{\overset{c^{mix_2}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2) \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$
$mix_2 - mix_2 - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2) \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$	$\underset{c^{mix_2}_{mix_2}}{\overset{c^{mix_2}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Sigma}(mix_2) \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$
$mix_2 - mix_2 - 3$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Delta, \Sigma}(mix_2)}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$	$\underset{c^{mix_2}_{mix_2}}{\overset{c^{mix_2}_{mix_2}}{\rightleftarrows}}$	$\dfrac{\overset{\phi}{\vdash \Delta} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Sigma}(mix_2)}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$
$mix_2 - \cup - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2) \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2)}{\vdash \Gamma, \Delta}(\cup)$	$\underset{c^{\cup}_{mix_2}}{\overset{c^{mix_2}_{\cup}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta} \quad \overset{\tau}{\vdash \Delta}}{\vdash \Delta}(\cup)}{\vdash \Gamma, \Delta}(mix_2)$
$mix_2 - \cup - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2) \quad \dfrac{\overset{\tau}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2)}{\vdash \Gamma, \Delta}(\cup)$	$\underset{c^{\cup}_{mix_2}}{\overset{c^{mix_2}_{\cup}}{\rightleftarrows}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Gamma}}{\vdash \Gamma}(\cup) \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2)$
$mix_2 - \emptyset - 1$	$\dfrac{}{\vdash \Gamma, \Delta}(\emptyset)$	$\underset{c^{\emptyset}_{mix_2}}{\overset{c^{mix_2}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash \Gamma}(\emptyset) \quad \overset{\pi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2)$

Table 1.19: Rule commutation (Part 15/16)

$mix_2 - \emptyset - 2$	$\dfrac{}{\vdash \Gamma, \Delta}(\emptyset)$	$\underset{C^{\emptyset}_{mix_2}}{\overset{C^{mix_2}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{}{\vdash \Delta}(\emptyset)}{\vdash \Gamma, \Delta}(mix_2)$
$\cup - \cup$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Gamma}}{\vdash \Gamma}(\cup) \quad \dfrac{\overset{\tau}{\vdash \Gamma} \quad \overset{\mu}{\vdash \Gamma}}{\vdash \Gamma}(\cup)}{\vdash \Gamma}(\cup)$	$\xrightarrow{C^{\cup}_{\cup}}$	$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Gamma}}{\vdash \Gamma}(\cup) \quad \dfrac{\overset{\phi}{\vdash \Gamma} \quad \overset{\mu}{\vdash \Gamma}}{\vdash \Gamma}(\cup)}{\vdash \Gamma}(\cup)$
$\cup - \emptyset$	$\dfrac{}{\vdash \Gamma}(\emptyset)$	$\underset{C^{\emptyset}_{\cup}}{\overset{C^{\cup}_{\emptyset}}{\rightleftarrows}}$	$\dfrac{\dfrac{}{\vdash \Gamma}(\emptyset) \quad \dfrac{}{\vdash \Gamma}(\emptyset)}{\vdash \Gamma}(\cup)$

Table 1.20: Rule commutation (Part 16/16)

$?w - ?c - 1$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A_1, \Gamma}}{\vdash ?A_1, ?A_2, \Gamma}(?w)}{\vdash ?A_1, \Gamma}(?c)$	$\overset{oe}{\rightsquigarrow}$	$\overset{\pi}{\vdash ?A, \Gamma}$
$?w - ?c - 2$	$\dfrac{\dfrac{\overset{\pi}{\vdash ?A_2, \Gamma}}{\vdash ?A_1, ?A_2, \Gamma}(?w)}{\vdash ?A_2, \Gamma}(?c)$	$\overset{oe}{\rightsquigarrow}$	$\overset{\pi}{\vdash ?A, \Gamma}$
$mix_0 - mix_2 - 1$	$\dfrac{\dfrac{}{\vdash}(mix_0) \quad \overset{\pi}{\vdash \Gamma}}{\vdash \Gamma}(mix_2)$	$\overset{om}{\rightsquigarrow}$	$\overset{\pi}{\vdash \Gamma}$
$mix_0 - mix_2 - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{}{\vdash}(mix_0)}{\vdash \Gamma}(mix_2)$	$\overset{om}{\rightsquigarrow}$	$\overset{\pi}{\vdash \Gamma}$
$\emptyset - \cup - 1$	$\dfrac{\dfrac{}{\vdash \Gamma}(\emptyset) \quad \overset{\pi}{\vdash \Gamma}}{\vdash \Gamma}(\cup)$	$\overset{oa}{\rightsquigarrow}$	$\overset{\pi}{\vdash \Gamma}$
$\emptyset - \cup - 2$	$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{}{\vdash \Gamma}(\emptyset)}{\vdash \Gamma}(\cup)$	$\overset{oa}{\rightsquigarrow}$	$\overset{\pi}{\vdash \Gamma}$

Table 1.21: Directed Rétoré transformation

Chapter 2

Properties of Sequent Calculus

In this chapter, we study properties of our transformations of proofs, which are particular kinds of *abstract rewriting systems*. In general, one loves to have two properties for this kind of systems: *strong normalization* and *confluence*. We call *convergent* a rewriting system with these two properties. Here are intuitions for these properties in the framework of functional programming languages, where a transformation is the evaluation of a function f in some argument x, *i.e.* replacing $f(x)$ by its value.

The first property, strong normalization, states that there is no infinite sequence of reductions. In the context of functional programming, strong normalization of (well-typed) programs means that accepted programs will stop at some point (maybe in one billion years, but will still halt). The second property, confluence, states that if there are several reductions that can be applied on an object o, yielding objects o_1 and o_2, then there are reductions of o_1 and of o_2 going to the same object. In the framework of programs, this means that if one can evaluate different functions in a program, say $f(g(x))$, then whichever function f or g is chosen first to be reduced, we can do other evaluations to reach the same result in both cases.

These two properties together yield to a strong result: when reducing any object in whatever way, one always reaches a normal form – on which no more reductions can be applied – which does not depend on the choices of reduction done, *i.e.* which is unique. For programs, this means that whatever evaluation of functions one chooses, in whatever order, the evaluated program reaches at some point a value, which is independent of the choices of reduction. As an example, consider a computation of $(1 + 2) \times (2 + 3)$. One can first compute the left or the right $+$, leading to either $3 \times (2+3)$ or $(1+2) \times 5$, then the other, yielding 3×5 in both cases, which finally reduces to 15. It would be disastrous to have a situation where computing first the left $+$ and then the second would lead to different results, or to have an unending computation where the computer never gives an answer. This is why, intuitively, these two properties matter.

In linear logic, we have two reduction systems: axiom-expansion and cut-elimination. The fundamental intuition here is that these two relations should be seen as *equivalence* of proofs. In other words, two sequent calculus proofs π and ϕ represent the "same" object if and only if they are related by axiom-expansion and cut-elimination, *i.e.* $\pi =_{\beta\eta} \phi$. Said differently, what often matters here, as for most applications, is less a proof and more its equivalence class by $\beta\eta$-equality. Convergence, leading to a unique normal form, would yield a canonical representative of a class; this is exactly what happens with integers and addition: $2+3$ and $2+2+1$ are the "same" because they both belong to the class of 5, which is the canonical representative.

Convergence of directed Rétoré transformation is obvious. Regarding axiom-expansion, everything works well, and one easily proves strong normalization and confluence. Even better, in a setting with no quantifier rules one can prove that given two proofs π and ϕ related by cut-elimination and axiom-expansion, posing π' (resp. ϕ') the unique (thanks to convergence of axiom-expansion) result of expanding all axioms in π (resp. ϕ), the proofs π' and ϕ' are related by cut-elimination. In other words, if $\pi =_{\beta\eta} \phi$ then $\pi' =_{\beta} \phi'$. This key observation allows us to reduce the study of proofs equal up to cut-elimination and axiom-expansion to the one of proofs equal up to cut-elimination only.

Unfortunately, cut-elimination is less well-behaved: it neither enjoys strong normalization nor confluence. This is why we need weaker properties. While strong normalization fails in general, this should only be due to the $cut - cut$ commutative case. Thus, when doing only other cases, namely $\xrightarrow{\overline{\beta}}$ steps, one recovers strong normalization. Meanwhile, even if cut-elimination is not confluent (even without the $cut - cut$ commutative case), it is confluent up to rule commutation, a property called *Church-Rosser modulo*. The idea is fairly simple, and can even be explained using junior high school notions and the usual exercise of expanding expressions. Consider arithmetical expressions made of variables $a, b, c, \ldots$ representing numbers, with the usual addition $+$ and multiplication $\times$. A standard exercise is the following: given an expression, expand it as much as possible using the distributivity laws $a \times (b + c) \to a \times b + a \times c$ and $(a + b) \times c \to a \times c + b \times c$. Looking at these distributivity laws as rewriting rules, one can wonder if this system is confluent. The answer is no, *i.e.* expanding a given term may yield different results according to the order in which the expansions are done. For instance, here are all the possible ways to expand $(a + b) \times (c + d)$.

$$\begin{tikzcd}[column sep=tiny]
 & & & (a+b)\times(c+d) \arrow[dll] \arrow[drr] & & & \\
 & a\times(c+d)+b\times(c+d) \arrow[dl] \arrow[ddr] & & & & (a+b)\times c+(a+b)\times d \arrow[dl] \arrow[ddr] & \\
(a\times c+a\times d)+b\times(c+d) \arrow[ddr] & & & & (a\times c+b\times c)+(a+b)\times d \arrow[ddr] & & \\
 & & a\times(c+d)+(b\times c+b\times d) \arrow[dl] & & & & (a+b)\times c+(a\times d+b\times d) \arrow[dl] \\
 & (a\times c+a\times d)+(b\times c+b\times d) & & & & (a\times c+b\times c)+(a\times d+b\times d) &
\end{tikzcd}$$

One sees there are two possible results: $(a\times c+a\times d)+(b\times c+b\times d)$ and $(a\times c+b\times c)+(a\times d+b\times d)$. While these two expressions are indeed different, it poses no problem because they are equal up to rearranging the terms, *i.e.* up to associativity and commutativity of the addition – using $(x+y)+z = x + (y + z)$ and $x + y = y + x$. One can prove this is a general phenomenon: while expanding an expression may not lead to a unique result, all possible results are related up to associativity and commutativity. Therefore, expanding expressions is not confluent but it is Church-Rosser modulo associativity and commutativity. A consequence is that a teacher correcting such an exercise cannot simply check that a student gave strictly the same answer as the one found by the teacher, but must check for a solution equal to it up to associativity and commutativity.

Going back to cut-elimination, we will work in the framework of $\mathsf{MALL}^{0,2}$ and prove cut-elimination is Church-Rosser modulo rule commutation, as well as strongly normalizing when not using an infinity of $cut - cut$ commutative cases. We will even prove the stronger result that two proofs are equal up to cut-elimination if and only if any of their normal forms are related by rule commutation. The main use of these properties in this thesis is to reduce the study of the equality of proofs up to cut-elimination to the study of *cut*-free proofs equal up to rule commutation.

Outline The aim of this chapter is to characterize the equality of proofs up to axiom-expansion and cut-elimination by means of rule commutation. This result is not surprising, but has not been proved before as far as we know, and is rather tedious to settle. To reach this goal, we first give standard definitions and results of the theory of Abstract Rewriting Systems about normalization, confluence, Church-Rosser modulo, etc (Section 2.1). We then consider axiom-expansion and easily prove strong normalization and confluence, as well as a reduction of equality up to cut-elimination and axiom-expansion to equality up to cut-elimination only (Section 2.2). The study of cut-elimination itself is more complex, as this system does not enjoy strong normalization and confluence, but only weaker properties such as Church-Rosser modulo rule commutations. We prove those weaker properties in the setting of $\mathsf{MALL}^{0,2}$, the demonstrations being more technical compared to axiom-expansion for a more comprehensive study is needed, with in addition problems when relaxing even slightly the setting (Section 2.3).

2.1 Abstract Rewriting Systems

We define here concepts from the theory of abstract rewriting systems (ARS), and state some results from the literature. Most of those can be found for instance in [Ter03]. They will be used for studying our transformations on proofs (axiom-expansion, cut-elimination and Rétoré transformation), as well as for the corresponding ones in the syntax of proof-nets. We give some standard notions such as strong normalization, confluence, ... (Section 2.1.1) as well as others which are less well-known, occurring when properties hold only up to an equivalence relation: Church-Rosser modulo a relation, local coherence modulo a relation, ... (Section 2.1.2).

2.1.1 Standard notions of Abstract Rewriting Systems

An **abstract rewriting system** is the data of a set A and a binary relation $\to$ on A. We use standard notations from this theory: given a relation $\to$, $\to^*$ (resp. $\to^+$, $\to^=$) is the transitive reflexive (resp. transitive, reflexive) closure of $\to$, while $\leftarrow$ is the converse relation – symmetric relations will correspond to symmetric symbols. By $\to^n$ we mean a sequence of n successive $\to$ steps. We denote by $\cdot$ the composition of relations, and by $\cup$ their union. The inclusion of relations is as usual denoted $\subseteq$. In all illustrations of definitions of this section about abstract rewriting systems, we adopt the following convention: hypotheses are in solid black whereas conclusions are in dashed red.

Definition 2.1 (Normalization)**.** Let $\to$ be a binary relation on the set A.

- When $a \to b$ we say a **reduces** to b.

- An element $a \in A$ is **a normal form** (for $\to$) if there exists no $b \in A$ such that $a \to b$. If $a \to^* b$ with b a normal form, we say b is **a normal form of** a.

- The relation $\to$ is **weakly normalizing** if for all $a \in A$, there exists a normal form $b \in A$ such that $a \to^* b$ – *i.e.* if there exists a normal form of any element.

- The relation $\to$ is **strongly normalizing** if for all $a \in A$, any rewriting sequence starting from a is finite, *i.e.* of the shape $a \to^* b$ for some $b \in A$.

Remark that strong normalization implies weak normalization.

Definition 2.2 (Confluence)**.** Let $\to$ be a binary relation on the set A.

- The relation $\to$ is **confluent**, or **Church-Rosser**, if ${}^*\!\leftarrow \cdot \to^* \subseteq \to^* \cdot {}^*\!\leftarrow$, illustrated below.

$$\begin{tikzcd}
& \cdot \arrow[dl, "*"'] \arrow[dr, "*"] & \\
\cdot \arrow[dr, dashed, "*"] & & \cdot \arrow[dl, dashed, "*"'] \\
& \cdot &
\end{tikzcd}$$

- The relation $\to$ has **the diamond property** if $\leftarrow \cdot \to \subseteq (\to \cdot \leftarrow) \cup =$, illustrated below.[1]

$$\begin{tikzcd}
& \cdot \arrow[dl] \arrow[dr] & \\
\cdot \arrow[dr, dashed] & & \cdot \arrow[dl, dashed] \\
& \cdot &
\end{tikzcd}
\quad \text{or} \quad
\begin{tikzcd}
& \cdot \arrow[dl] \arrow[dr] & \\
\cdot & = & \cdot
\end{tikzcd}$$

One can prove that the diamond property implies confluence, by a simple induction on the sum of the lengths of the two $\to^*$ sequences.

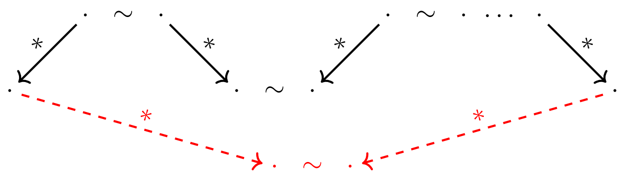
If $\to$ is strongly normalizing and confluent, we call it **convergent**. In this case, each $a \in A$ has a unique normal form.

2.1.2 Abstract Rewriting Systems modulo an equivalence relation

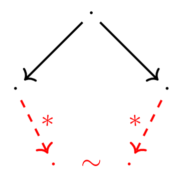
We now extend previous definitions to the case “up to an equivalence relation”.

Definition 2.3. Let $\sim$ and $\to$ be (binary) relations on a set A such that $\sim$ is an equivalence relation. The relation $\to$ is **Church-Rosser modulo** $\sim$ if $(\to \cup \leftarrow \cup \sim)^* \subseteq \to^* \cdot \sim \cdot {}^*\!\leftarrow$, illustrated below.

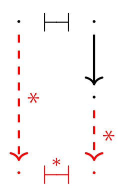
[1] Contrary to the usual definition $\leftarrow \cdot \to \subseteq \to \cdot \leftarrow$ found in the literature, *e.g.* [Ter03, Definition 1.1.8], we add $\cup =$, *i.e.* replace confluence in one step by confluence in one or zero step. This is due to the following case: if $b \leftarrow a \to b$ with b normal form, then there is no hope to find a c such that $b \to c \leftarrow b$; hence the diamond property stated as $\leftarrow \cdot \to \subseteq \to \cdot \leftarrow$ cannot hold as soon as $a \to b$ for b a normal form. While our notion is the one usually wanted (but perhaps not the one usually stated), there is no good terminology we are aware of to distinguish it from $\leftarrow \cdot \to \subseteq \to \cdot \leftarrow$, both notions being called the diamond property.



Definition 2.4. Let $\sim, \to$ be relations on a set A such that $\sim$ is an equivalence relation. The relation $\to$ is **locally confluent modulo** $\sim$ if $\leftarrow \cdot \to \subseteq \to^* \cdot \sim \cdot {}^*\!\leftarrow$, illustrated below.



Definition 2.5. Let $\vdash\!\dashv, \to$ be relations on a set A such that $\vdash\!\dashv$ is symmetric. The relation $\to$ is **locally coherent with** $\vdash\!\dashv$ if $\vdash\!\dashv \cdot \to \subseteq \to^* \cdot \overset{*}{\vdash\!\dashv} \cdot {}^*\!\leftarrow$, illustrated below.



Other definitions can be adapted to this setting, but we will not need them here. In order to obtain the main result of this chapter – stating that the normal forms for cut-elimination of two $\mathsf{MALL}^{0,2}$ proofs related by $=_\beta$, are related by $\overset{r^*}{\vdash\!\dashv}$ (Theorem 2.49) – we see it as a Church-Rosser property modulo an equivalence relation. This property can be deduced by applying the following result from Huet.

Theorem 2.6 ([Hue80])**.** *Let $\vdash\!\dashv$ and $\to$ be relations on a set A such that $\vdash\!\dashv$ is symmetric. If $\to \cdot \overset{*}{\vdash\!\dashv}$ is strongly normalizing, $\to$ is locally confluent modulo $\overset{*}{\vdash\!\dashv}$ and locally coherent with $\vdash\!\dashv$, then $\to$ is Church-Rosser modulo $\overset{*}{\vdash\!\dashv}$.*

While this theorem is enough for MALL^2, it cannot be applied to $\mathsf{MALL}^{0,2}$ nor for the full logic, for having $\to \cdot \overset{*}{\vdash\!\dashv}$ strongly normalizing does not hold in the general case. The following generalization can be applied for $\mathsf{MALL}^{0,2}$, and is conjectured to be applicable in full linear logic, with all optional rules and Rétoré transformations.

Theorem 2.7 ([AT12, Theorem 2.2])**.** *Let $\vdash\!\dashv$, $\to$ and $\rightsquigarrow$ be relations on a set A such that $\vdash\!\dashv$ is symmetric and $\rightsquigarrow \subseteq \vdash\!\dashv$. Set $\Rightarrow = \to \cup \rightsquigarrow$. Suppose:*

(i) $\to \cdot \rightsquigarrow^*$ *is strongly normalizing;*

(ii) $\leftarrow \cdot \to \subseteq \Rightarrow^* \cdot \overset{=}{\vdash\!\dashv} \cdot {}^*\!\Leftarrow$*;*

(iii) $\vdash\!\dashv \cdot \to \subseteq (\overset{=}{\vdash\!\dashv} \cdot {}^*\!\Leftarrow) \cup (\to \cdot \Rightarrow^* \cdot \overset{=}{\vdash\!\dashv} \cdot {}^*\!\Leftarrow)$.

Then $\to$ *is Church-Rosser modulo* $\overset{*}{\vdash\!\dashv}$.

This is indeed a generalization, as when instantiating $\rightsquigarrow\, = \,\vdash\!\dashv$ local confluence and local coherence implies respectively Items (ii) and (iii). This last theorem is surprisingly easy to prove: it is done in around one page in [AT12, Section 2], with only standard definitions and no need of previously known results.

There are other theorems from the literature for proving a rewriting system is Church-Rosser modulo an equivalence relation, see for instance [Ohl98] with many of those. Unfortunately, most of them are about non-local properties, *i.e.* considering $\overset{*}{\vdash\!\dashv}$ instead of $\vdash\!\dashv$. For instance, instead of asking for local coherence with $\vdash\!\dashv$ one needs it with $\overset{*}{\vdash\!\dashv}$ (meaning $\overset{*}{\vdash\!\dashv} \cdot \to\, \subseteq\, \to^* \cdot \overset{*}{\vdash\!\dashv} \cdot {}^*\!\leftarrow$). In our applications, we will consider $\vdash\!\dashv\, = \,\overset{r}{\vdash\!\dashv}$. While doing cases on a step of $\overset{r}{\vdash\!\dashv}$ is as tedious as it is boring, it still has the advantage of being a simple case study. Meanwhile, considering an arbitrary $\overset{r^*}{\vdash\!\dashv}$ step seems complicated, as successive rule commutations can completely reshape a proof. This prevents us from using these results from the literature asking for non-local properties.

2.2 Axiom-expansion

We prove in this section that $=_{\beta\eta}$, equality up to cut-elimination and axiom-expansion, can be reduced to $=_\beta$, equality up to cut-elimination only, between atomic-axiom proofs, *i.e.* normal forms for axiom-expansion. The major property is that axiom-expansion $\xrightarrow{\eta}$ is convergent.

Proposition 2.8. *In linear logic with any of the optional rules, axiom-expansion* $\xrightarrow{\eta}$ *is strongly normalizing and confluent.*

Proof. Strong normalization follows from the fact that a $\xrightarrow{\eta}$ step strictly decreases the sum of the sizes of the formulas on which an *ax*-rule is applied.

Confluence can be deduced from the diamond property. Observe that there is no critical pair: two distinct steps $\phi \xleftarrow{\eta} \pi \xrightarrow{\eta} \tau$ always commute, *i.e.* there exists μ such that $\phi \xrightarrow{\eta} \mu \xleftarrow{\eta} \tau$, by doing in ϕ the rewriting corresponding to the one done in $\pi \xrightarrow{\eta} \tau$, and similarly in τ the one done in $\pi \xrightarrow{\eta} \phi$. Hence the diamond property of $\xrightarrow{\eta}$, and thus its confluence. $\square$

With a similar proof, one can prove that (any subset of the) Rétoré transformation (Definition 1.17) is convergent.

Remark 2.9. As long as there exists an *ax*-rule not on an atom, an $\xrightarrow{\eta}$ axiom-expansion step can be applied. Thus, atomic-axiom proofs correspond exactly to proofs in normal form for $\xrightarrow{\eta}$.

Thanks to Proposition 2.8, we denote by $\eta(\pi)$ the unique η-normal form of a proof π, *i.e.* the proof obtained by expanding iteratively all *ax*-rules in π (in any order thanks to confluence). Thus, we can define $\mathrm{id}_A = \eta(ax_A)$, the axiom-expansion of the proof consisting of only one *ax*-rule. The goal of this section is to prove the following (proof on page 62).

Proposition 2.10 (Reduction to atomic-axiom proofs)**.** *Consider the sub-system of linear logic with all rules (including optional ones) except the quantifiers rules* $\forall$ *and* $\exists$*. Set* $\overset{o}{\vdash\!\dashv}$ *some (possible none, or all) of the Rétoré transformations. Let* π *and* τ *be proofs such that* $\pi =_{\beta\eta o} \tau$*. Then* $\eta(\pi) =_{\beta o} \eta(\tau)$ *with, in this sequence, only proofs in* η*-normal form.*

This allows us to then consider atomic-axioms proofs only, manipulated through composition by cut, cut-elimination and possibly Rétoré transformations, never to speak again of η rewriting.

We will need some intermediate results to reach our goal, first studying the interactions of $\xrightarrow{\eta}$ with $\xrightarrow{\beta}$, then with $\overset{o}{\vdash\!\dashv}$. We set $a(\pi \xrightarrow{\beta^*} \tau)$ the multiset of the sizes of the formulas in the *ax* key cases of these $\xrightarrow{\beta}$ reductions.

Lemma 2.11. *Let π, τ and ϕ be proofs such that $\tau \xleftarrow{\beta} \pi \xrightarrow{\eta} \phi$. Then there exists μ such that $\tau \xrightarrow{\eta^*} \mu \xleftarrow{\beta} \phi$ or there exist μ_1 and μ_2 such that $\phi \xrightarrow{\eta^*} \mu_1 \xrightarrow{\beta^*} \mu_2 \xleftarrow{\eta^*} \tau$. Furthermore, $a(\phi \xrightarrow{\beta} \mu) = a(\pi \xrightarrow{\beta} \tau)$ in the first case and $a(\mu_1 \xrightarrow{\beta^*} \mu_2) < a(\pi \xrightarrow{\beta} \tau)$ in the second one. Diagrammatically:*

$$
\begin{tikzcd}
\pi \arrow[r, "\eta"] \arrow[d, "\beta"'] \arrow[dr, phantom, "="] & \phi \arrow[d, dashed, "\beta"] \\
\tau \arrow[r, dashed, "\eta^*"'] & \mu
\end{tikzcd}
\qquad
\begin{tikzcd}
\pi \arrow[r, "\eta"] \arrow[d, "\beta"'] & \phi \arrow[r, dashed, "\eta^*"] \arrow[d, phantom, ">"] & \mu_1 \arrow[d, dashed, "\beta^*"] \\
\tau \arrow[rr, dashed, "\eta^*"'] & & \mu_2
\end{tikzcd}
$$

Proof. Call r the *ax*-rule that $\pi \xrightarrow{\eta} \phi$ expands, and A its formula. If the cut-elimination step is not an *ax* key case using r, then the two steps commute and there exists μ such that $\phi \xrightarrow{\beta} \mu$ and $\tau \xrightarrow{\eta} \mu$ (or $\tau \xrightarrow{\eta} \cdot \xrightarrow{\eta} \mu$ if r belongs to a sub-proof duplicated by the $\xrightarrow{\beta}$ step, or $\tau = \mu$ if it belongs to a sub-proof erased by the $\xrightarrow{\beta}$ step). In particular, $a(\phi \xrightarrow{\beta} \mu) = a(\pi \xrightarrow{\beta} \tau)$ for they use the same rules.

Otherwise, the cut-elimination step is an *ax* key case on r, with a *cut*-rule we call c and a sub-proof ρ in the other branch of c than the one leading to r; c introduces the formula A. The reasoning we apply is depicted on Figure 2.1. Starting from ϕ, consider the rules introducing $A^\perp$ in (all slices of) ρ. If any of them are *ax*-rules, then these are necessarily on the formula A; expand those *ax*-rules, in both ϕ and τ (keeping the same names for proofs π, τ and ρ by abuse). Then, in ϕ, commute the *cut*-rule c with rules of ρ until reaching the rules introducing $A^\perp$ in every slice (which are rules of the main connective of $A^\perp$ or $\top$-rules). Applying the corresponding key cases or $\top - cut$ commutative case (first commuting with a rule of the expanded axiom r if A is a positive formula, and doing it after if A is a negative formula), then the *ax* key cases on strict sub-formulas of A yield τ. During these *ax* key cases, we cut on sub-formulas of A, so on formulas of a strictly smaller size. Therefore $\phi \xrightarrow{\eta^*} \cdot \xrightarrow{\beta^*} \cdot \xleftarrow{\eta^*} \tau$, with $a(\cdot \xrightarrow{\beta^*} \cdot) < a(\pi \xrightarrow{\beta} \tau)$. ☐

Fact 2.12. *Any cut-elimination step $\xrightarrow{\beta}$ which is not an $\forall - \exists$ key case preserves being in η-normal form.*

Proof. A $\xrightarrow{\beta}$ rewriting step can never create an *ax*-rule on a formula A, except by duplicating an already present *ax*-rule on the same formula. The only exception to this is the $\forall - \exists$ key case, because it turns a proof π_1 into $\pi_1[B/X]$, so any *ax*-rule on X becomes an *ax*-rule on B, which is not atomic if B is not an atom. ☐

Lemma 2.13. *Consider the sub-system made of linear logic with any of the optional rules, but without the quantifiers $\forall$ and $\exists$. Let π, τ and ϕ be proofs such that $\tau \xleftarrow{\eta^*} \pi \xrightarrow{\beta^*} \phi$, with τ an η-normal proof. There exists an η-normal proof μ such that $\tau \xrightarrow{\beta^*} \mu \xleftarrow{\eta^*} \phi$. Diagrammatically:*

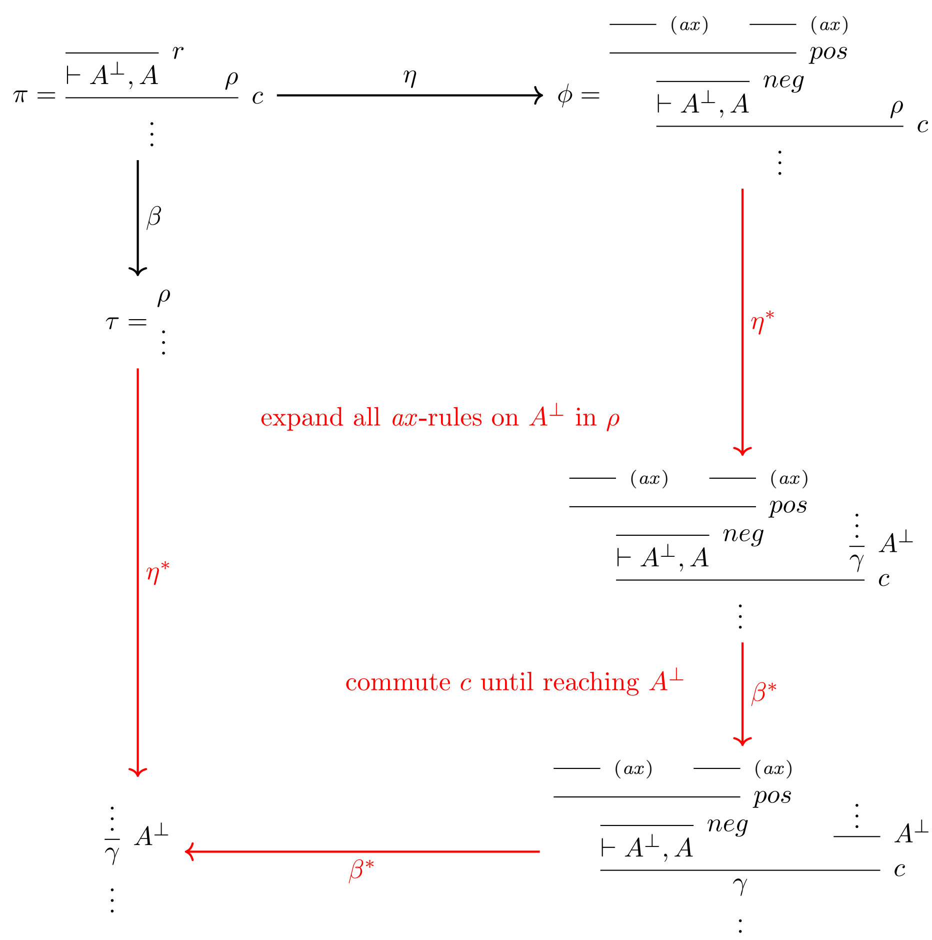


Figure 2.1: Schematic representation of the second case of the proof of Lemma 2.11

$$\begin{tikzcd}
\pi \arrow[r, "\eta^*"] \arrow[d, "\beta^*"] & \tau \arrow[d, dashed, "\beta^*"] & \textit{η-normal} \\
\phi \arrow[r, dashed, "\eta^*"] & \mu & \textit{η-normal}
\end{tikzcd}$$

Proof. Assume $\tau \xleftarrow{\eta^m} \pi \xrightarrow{\beta^n} \phi$. If we find a μ respecting the diagram, then it is an η-normal form by Fact 2.12. We reason by induction on the lexicographic order of the triple $\left(a(\pi \xrightarrow{\beta^*} \phi), n, m\right)$. If $n = 0$ or $m = 0$, then the result trivially holds ($\mu = \tau$ or $\mu = \phi$).

Consider the case $n+1$ and $m+1$. Therefore, $\tau \xleftarrow{\eta^m} \iota \xleftarrow{\eta} \pi \xrightarrow{\beta} \kappa \xrightarrow{\beta^n} \phi$. We apply Lemma 2.11 on $\iota \xleftarrow{\eta} \pi \xrightarrow{\beta} \kappa$, yielding ρ such that $\iota \xrightarrow{\beta} \rho \xleftarrow{\eta^*} \phi$ with $a(\iota \xrightarrow{\beta} \rho) = a(\pi \xrightarrow{\beta} \kappa)$ or ρ_1 and ρ_2 such that $\iota \xrightarrow{\eta^*} \rho_1 \xrightarrow{\beta^*} \rho_2 \xleftarrow{\eta^*} \kappa$ with $a(\rho_1 \xrightarrow{\beta^*} \rho_2) < a(\pi \xrightarrow{\beta} \kappa)$. Both of these cases, and the reasonings we will apply, are illustrated by diagrams preceding them, with the following color convention: in blue are uses of Lemma 2.11, in green of confluence of $\xrightarrow{\eta}$ and in red of the induction hypothesis.

$$\begin{tikzcd}
\pi \arrow[r, "\eta"] \arrow[d, "\beta"'] \arrow[dr, phantom, "="] & \iota \arrow[r, "\eta^m"] \arrow[d, "\beta"] \arrow[dr, phantom, "IH"] & \tau \arrow[d, "\beta^*"] & \text{η-normal} \\
\kappa \arrow[r, "\eta^*"] \arrow[d, "\beta^n"'] \arrow[drr, phantom, "IH"] & \rho \arrow[r, "\eta^*"] & \nu \arrow[d, "\beta^*"] & \text{η-normal} \\
\phi \arrow[rr, "\eta^*"] & & \mu & \text{η-normal}
\end{tikzcd}$$

Assume to be in the first case. Applying the induction hypothesis on $\tau \xleftarrow{\eta^m} \iota \xrightarrow{\beta} \rho$, with τ in η-normal form, $a(\iota \xrightarrow{\beta} \rho) = a(\pi \xrightarrow{\beta} \kappa) \leq a(\pi \xrightarrow{\beta^*} \phi)$, $1 \leq n+1$ and $m < m+1$, there exists an η-normal proof ν such that $\tau \xrightarrow{\beta^*} \nu \xleftarrow{\eta^*} \rho$. We now apply the induction hypothesis on $\nu \xleftarrow{\eta^*} \rho \xleftarrow{\eta^*} \kappa \xrightarrow{\beta^n} \phi$, with ν in η-normal form, $a(\kappa \xrightarrow{\beta^n} \phi) \leq a(\pi \xrightarrow{\beta^*} \phi)$ and $n < n+1$. We obtain an η-normal proof μ such that $\nu \xrightarrow{\beta^*} \mu \xleftarrow{\eta^*} \phi$. This concludes the first case.

$$
\begin{tikzcd}
\pi \arrow[r, "\eta"] \arrow[dd, "\beta"'] & \iota \arrow[r, "\eta^m"] \arrow[d, "\eta^*"] & \tau \arrow[dd, "\beta^*"] & \eta\text{-normal} \\
& \rho_1 \arrow[ur, "\eta^*"'] \arrow[d, "\beta^*"] \arrow[dl, phantom, ">"] \arrow[dr, phantom, "IH"] & & \\
\kappa \arrow[r, "\eta^*"] \arrow[d, "\beta^n"'] & \rho_2 \arrow[r, "\eta^*"] \arrow[d, phantom, "IH"] & \nu \arrow[d, "\beta^*"] & \eta\text{-normal} \\
\phi \arrow[rr, "\eta^*"] & & \mu & \eta\text{-normal}
\end{tikzcd}
$$

Consider the second case. Using the confluence of $\xrightarrow{\eta}$ on $\tau \xleftarrow{\eta^m} \iota \xrightarrow{\eta^*} \rho_1$ (Proposition 2.8), with τ in η-normal form, yields $\tau \xleftarrow{\eta^*} \rho_1$. We then apply the induction hypothesis on $\tau \xleftarrow{\eta^*} \rho_1 \xrightarrow{\beta^*} \rho_2$, with τ in η-normal form and $a(\rho_1 \xrightarrow{\beta^*} \rho_2) < a(\pi \xrightarrow{\beta^*} \phi)$. This yields an η-normal proof ν such that $\tau \xrightarrow{\beta^*} \nu \xleftarrow{\eta^*} \rho_2$. We use the induction hypothesis again, this time on $\nu \xleftarrow{\eta^*} \rho_2 \xleftarrow{\eta^*} \kappa \xrightarrow{\beta^n} \phi$, with ν in η-normal form, $a(\kappa \xrightarrow{\beta^n} \phi) \leq a(\pi \xrightarrow{\beta^*} \phi)$ and $n < n+1$. We obtain an η-normal proof μ with $\nu \xrightarrow{\beta^*} \mu \xleftarrow{\eta^*} \phi$, solving the second case. □

Lemma 2.14. *If $\pi \overset{o}{\vdash\!\dashv} \phi$, the $\eta(\pi) \overset{o}{\vdash\!\dashv} \eta(\phi)$ using a $\overset{o}{\vdash\!\dashv}$ step of the same kind ($\overset{om}{\rightsquigarrow}$, $\overset{om}{\leftrightsquigarrow}$, $\overset{oa}{\rightsquigarrow}$, ...).*

Proof. We start by proving the following. Let π, ϕ and π' be proofs such that $\pi' \xleftarrow{\eta} \pi \overset{o}{\vdash\!\dashv} \phi$. Then $\pi' \overset{o}{\vdash\!\dashv} \cdot \xleftarrow{\eta} \phi$, with this $\overset{o}{\vdash\!\dashv}$ step being the same kind as in $\pi \overset{o}{\vdash\!\dashv} \phi$. Diagrammatically:

$$
\begin{tikzcd}
\pi \arrow[r, phantom, "\overset{o}{\vdash\!\dashv}"] \arrow[d, "\eta"] & \phi \arrow[d, dashed, "\eta"] \\
\pi' \arrow[r, phantom, "\overset{o}{\vdash\!\dashv}"] & \cdot
\end{tikzcd}
$$

Indeed, in ϕ, one can apply an $\xrightarrow{\eta}$ step on the ax-rule corresponding to the $\pi \xrightarrow{\eta} \pi'$ step. One gets a proof ϕ', and remark that $\pi' \overset{o}{\vdash\!\dashv} \phi'$ using the same rules as in $\pi \overset{o}{\vdash\!\dashv} \phi$.

Now let us prove our result. We have $\eta(\pi) \xleftarrow{\eta^*} \pi \overset{o}{\vdash\!\dashv} \phi \xrightarrow{\eta^*} \eta(\phi)$. Apply the above result on the left sequence of $\xrightarrow{\eta}$ steps, yielding by induction $\eta(\pi) \overset{o}{\vdash\!\dashv} \tau \xleftarrow{\eta^*} \phi \xrightarrow{\eta^*} \eta(\phi)$ for some proof τ. As a $\overset{o}{\vdash\!\dashv}$ step preserves being an η-normal proof, using Proposition 2.8 one gets $\tau = \eta(\phi)$, concluding the proof. □

We can now prove the main result of this section, Proposition 2.10.

Proof of Proposition 2.10. We reason by induction on the length of the sequence $\pi =_{\beta\eta o} \tau$. If it is of null length, then $\pi = \tau$ hence $\eta(\pi) = \eta(\tau)$. Otherwise, there is a proof ϕ such that $\pi - \phi =_{\beta\eta o} \tau$ with $- \in \{\xrightarrow{\beta}; \xleftarrow{\beta}; \xrightarrow{\eta}; \xleftarrow{\eta}; \overset{o}{\vdash\!\dashv}\}$. By induction hypothesis, $\eta(\phi) =_{\beta o} \eta(\tau)$, with only atomic-axiom proofs in this sequence. We distinguish cases according to $\pi - \phi$.

If $\pi \xrightarrow{\beta} \phi$, then as $\eta(\pi) \xleftarrow{\eta^*} \pi$ we can apply Lemma 2.13 to obtain an η-normal proof μ such that $\eta(\pi) \xrightarrow{\beta^*} \mu \xleftarrow{\eta^*} \phi$. Thus, $\mu = \eta(\phi)$, so $\eta(\pi) \xrightarrow{\beta^*} \eta(\phi) =_\beta \eta(\tau)$ and the result holds.

Similarly, if $\pi \xleftarrow{\beta} \phi$ then, as $\phi \xrightarrow{\eta^*} \eta(\phi)$, there exists an η-normal proof μ such that $\pi \xrightarrow{\eta^*} \mu \xleftarrow{\beta^*} \eta(\phi)$ (Lemma 2.13). Thus $\mu = \eta(\pi)$, and $\eta(\pi) \xleftarrow{\beta^*} \eta(\phi) =_\beta \eta(\tau)$.

If $\pi \overset{o}{\vdash\!\dashv} \phi$, then by Lemma 2.14 $\eta(\pi) \overset{o}{\vdash\!\dashv} \eta(\phi)$, allowing us to conclude $\eta(\pi) =_{\beta o} \eta(\tau)$.

Finally, if $\pi \xrightarrow{\eta} \phi$ or $\pi \xleftarrow{\eta} \phi$, then $\eta(\pi) = \eta(\phi) =_\beta \eta(\tau)$ and the conclusion follows. □

2.3 Cut-elimination

We now study cut-elimination. Weak normalization of cut-elimination is important in logic: it permits to have a cut-free proof of any provable sequent, and thus to work with a restricted set of rules, simplifying reasonings. Furthermore, the cut-rule in $\mathsf{MALL}^{0,2}$ is the only rule not having the sub-sequent property (Fact 1.7). Not having this property makes automatic proof search complex, due to the size of the formula space from which one can choose a cut formula. The existence of a cut-free proof allows to remove this choice.

In linear logic, normalization and confluence of cut-elimination in proof-nets (a graphical syntax for proofs, that we will consider in Chapter 4) have been well studied in various settings: for MELL [Acc13; DG99], MALL [HG05; LM08], nets with additive boxes [Tor03; Tor01] and even with all connectives and second order quantifiers [PT10], in differential nets [PT17; Tra09], etc. Meanwhile, there are way less results about normalization and confluence in sequent calculus. Weak normalization of cut-elimination is a folklore result, see for instance [LL22, Section 1.5].

Theorem 2.15. *Cut-elimination for the sequent calculus of linear logic is weakly normalizing.*

However, at the best of my knowledge there is not a lot of other results about cut-elimination in sequent calculus. One of the reasons for this lack of results is that only weaker properties hold, contrary to axiom-expansion. Indeed, cut-elimination neither enjoys strong normalization nor confluence (Section 2.3.1). Nonetheless, cut-elimination still has some good properties, albeit weaker ones; we prove those in the simpler framework of $\mathsf{MALL}^{0,2}$. If one remove the $cut-cut$ commutation, *i.e.* consider $\xrightarrow{\overline{\beta}}$ instead of $\xrightarrow{\beta}$, then the resulting rewriting system is strongly normalizing, and we can even add some commutations between steps while keeping strong normalization (Sections 2.3.3 and 2.3.4). We conjecture a corresponding result for the full logic (Section 2.3.5). To illustrate the gap between weak and strong normalization, we first give a very simple proof that cut-elimination is weakly normalizing in MALL (Section 2.3.2). Finally, even if cut-elimination is not confluent, it is Church-Rosser modulo rule commutation (Section 2.3.6). A result which is not surprising, and rather tedious to settle. It has the important consequence of reducing equality up to cut-elimination to equality up to rule commutation between normal forms. When studying this problem, we thought that this Church-Rosser property has not been proved before, leading to the study described in this section as we needed this result for the study of isomorphisms (see Chapter 6). It has since been brought to our attention that Cockett and Pastro had introduced in [CP05] a term language for MALL with rewriting rules corresponding to cut-elimination and rule commutations, and for which this Church-Rosser property was proved in the appendix of their paper. We thus end this section with a discussion on the similarities and differences between their work and ours (Section 2.3.7).

2.3.1 No strong normalization nor confluence

Cut-elimination is not strongly normalizing, because of the $cut - cut$ commutative case: in a proof with two successive *cut*-rules, one can apply this rewriting step ad nauseam, alternating between two proofs. Unfortunately, the root of the problem is not simply the $cut - cut$ commutation, but is more on the confluence side. Cut-elimination is not confluent, as two commutative cases may be applied on the same *cut*-rule, and doing one or the other may not lead to the same normal form. As an example, the proof

$$\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash B^\perp, B}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \qquad \dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash C^\perp, C}\,(ax)}{\vdash A^\perp \otimes C^\perp, A, C}\,(\otimes)}{\vdash A \otimes B, A^\perp \otimes C^\perp, B^\perp, C}\,(cut)$$

reduces to both

$$\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash C^\perp, C}\,(ax)}{\vdash A, A^\perp \otimes C^\perp, C}\,(\otimes) \quad \dfrac{}{\vdash B^\perp, B}\,(ax)}{\vdash A \otimes B, A^\perp \otimes C^\perp, B^\perp, C}\,(\otimes) \quad \text{and} \quad \dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \quad \dfrac{}{\vdash B^\perp, B}\,(ax)}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes) \quad \dfrac{}{\vdash C^\perp, C}\,(ax)}{\vdash A \otimes B, A^\perp \otimes C^\perp, B^\perp, C}\,(\otimes)$$

according respectively to whether the *cut*-rule is first commuted with the $\otimes$-rule on its left or on its right in the first cut-elimination step. Still, even if these two proofs are not equal, they are related by rule commutation!

2.3.2 Easy proof of weak normalization in MALL

Here is a standalone proof that cut-elimination in MALL is weakly normalizing. The argument is very simple: we define a quantity called the *mass* of a *cut*-rule, which depends only on the shape of the *cut*-rule, and prove a cut-elimination step applied on a top-most *cut*-rule produces only *cut*-rules of lesser masses.

Definition 2.16. The **mass** $m(A)$ of a formula A is a natural number defined by induction:

- $m(X^+) = m(X^-) = m(1) = m(\perp) = m(\top) = m(0) = 2$
- $m(A \otimes B) = m(A \parr B) = m(A \,\&\, B) = m(A \oplus B) = (m(A) + 1) \times (m(B) + 1)$

We extend this notion to sequents by defining $m(\vdash A_1, \dots, A_n) = \prod_{i=1}^n m(A_i)$ (with the usual convention that the empty product equals 1).

The **mass** of a *cut*-rule c of the shape $\dfrac{\vdash A^\perp, \Gamma \quad \vdash A, \Delta}{\vdash \Gamma, \Delta}\,(cut)$ is $m(c) = m(A) \times m(\Gamma, \Delta)$.

The **mass** $m(\pi)$ of a proof π is the sum of the masses of its *cut*-rules.

Fact 2.17. *For any formula A, $m(A) = m(A^\perp) > 1$.*

Lemma 2.18. *Consider a cut-elimination step $\tau \xrightarrow{\beta} \phi$ and call c the (unique) cut-rule in τ involved in this step and $(c_i)_{i \in [\![1;n]\!]}$ the $n \in \{0; 1; 2\}$ resulting cut-rules in ϕ. Then $m(c) > \sum_{i=1}^n m(c_i)$.*

Proof. It suffices to compute the masses before and after each cut-elimination step. In the following case study, we use implicitly Fact 2.17.

ax key case Here

$$\tau = \cfrac{\cfrac{}{\vdash A^\perp, A}(ax) \qquad \overset{\pi}{\vdash A, \Gamma}}{\underset{\rho}{\vdash A, \Gamma}}(c) \qquad\qquad \phi = \underset{\rho}{\overset{\pi}{\vdash A, \Gamma}}$$

We have $n = 0$ and $m(c) = m(A) \times m(A) \times m(\Gamma) > 0 = \sum_{i=1}^n m(c_i)$.

$\parr - \otimes$ *key case* Here

$$\tau = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}(\otimes) \qquad \cfrac{\overset{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash B^\perp \parr A^\perp, \Sigma}(\parr)}{\underset{\rho}{\vdash \Gamma, \Delta, \Sigma}}(c) \qquad\qquad \phi = \cfrac{\overset{\pi_1}{\vdash A, \Gamma} \qquad \cfrac{\overset{\pi_2}{\vdash B, \Delta} \quad \overset{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash A^\perp, \Delta, \Sigma}(c_2)}{\underset{\rho}{\vdash \Gamma, \Delta, \Sigma}}(c_1)$$

We have $n = 2$ and:

$$\begin{aligned}
& m(c) - \sum_{i=1}^{n} m(c_i) \\
& = m(A \otimes B) \times m(\Gamma, \Delta, \Sigma) - m(A) \times m(\Gamma, \Delta, \Sigma) - m(B) \times m(A^\perp, \Delta, \Sigma) \\
& \geq (m(A \otimes B) - m(A) - m(B) \times m(A)) \times m(\Gamma, \Delta, \Sigma) \\
& = (m(B) + 1) \times m(\Gamma, \Delta, \Sigma) > 0
\end{aligned}$$

$\& - \oplus_1$ *key case* Here

$$\tau = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Gamma}}{\vdash A \,\&\, B, \Gamma}(\&) \qquad \cfrac{\overset{\pi_3}{\vdash B^\perp, \Delta}}{\vdash B^\perp \oplus A^\perp, \Delta}(\oplus_1)}{\underset{\rho}{\vdash \Gamma, \Delta}}(c) \qquad\qquad \phi = \cfrac{\overset{\pi_2}{\vdash B, \Gamma} \quad \overset{\pi_3}{\vdash B^\perp, \Delta}}{\underset{\rho}{\vdash \Gamma, \Delta}}(c_1)$$

We have $n = 1$ and $m(c) = m(A \,\&\, B) \times m(\Gamma, \Delta) > m(B) \times m(\Gamma, \Delta) = \sum_{i=1}^n m(c_i)$.

$\& - \oplus_2$ *key case* This case is very similar to the $\& - \oplus_1$ key case.

$\perp - 1$ *key case* Here

$$\tau = \cfrac{\cfrac{}{\vdash 1}(1) \qquad \cfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \Gamma, \perp}(\perp)}{\underset{\rho}{\vdash \Gamma}}(c) \qquad\qquad \phi = \underset{\rho}{\overset{\pi}{\vdash \Gamma}}$$

We have $n = 0$ and $m(c) = 2m(\Gamma) > 0 = \sum_{i=1}^n m(c_i)$.

$\parr - cut$ *commutative case* Here

$$\tau = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, B, C, \Gamma}}{\vdash A, B \parr C, \Gamma}(\parr) \qquad \overset{\pi_2}{\vdash A^\perp, \Delta}}{\underset{\rho}{\vdash B \parr C, \Gamma, \Delta}}(c) \qquad\qquad \phi = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, B, C, \Gamma} \quad \overset{\pi_2}{\vdash A^\perp, \Delta}}{\vdash B, C, \Gamma, \Delta}(c_1)}{\underset{\rho}{\vdash B \parr C, \Gamma, \Delta}}(\parr)$$

We have $n = 1$ and:

$$\begin{aligned}
&m(c) - \sum_{i=1}^{n} m(c_i) \\
&= m(A) \times m(B \parr C) \times m(\Gamma, \Delta) - m(A) \times m(B) \times m(C) \times m(\Gamma, \Delta) \\
&= m(A) \times (m(B \parr C) - m(B)m(C)) \times m(\Gamma, \Delta) \\
&= m(A) \times (m(B) + m(C) + 1) \times m(\Gamma, \Delta) > 0
\end{aligned}$$

$\otimes - cut - 1$, $\otimes - cut - 2$, $\oplus_i - cut$ and $\bot - cut$ commutative cases These cases are quite similar to the $\parr - cut$ commutative case.

$\& - cut$ commutative case Here

$$\tau = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_2}{\vdash A, C, \Gamma}}{\vdash A, B \,\&\, C, \Gamma}\,(\&) \quad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\underset{\rho}{\vdash B \,\&\, C, \Gamma, \Delta}}\,(c) \qquad \phi = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash B, \Gamma, \Delta}\,(c_1) \quad \cfrac{\overset{\pi_2}{\vdash A, C, \Gamma} \quad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash C, \Gamma, \Delta}\,(c_2)}{\underset{\rho}{\vdash B \,\&\, C, \Gamma, \Delta}}\,(\&)$$

We have $n = 2$ and:

$$\begin{aligned}
&m(c) - \sum_{i=1}^{n} m(c_i) \\
&= m(A) \times m(\Gamma, \Delta) \times m(B \,\&\, C) - m(A) \times m(\Gamma, \Delta) \times m(B) - m(A) \times m(\Gamma, \Delta) \times m(C) \\
&= m(A) \times m(\Gamma, \Delta) \times (m(B \,\&\, C) - m(B) - m(C)) \\
&= m(A) \times m(\Gamma, \Delta) \times (m(B)m(C) + 1) > 0
\end{aligned}$$

$\top - cut$ commutative case Here

$$\tau = \cfrac{\cfrac{}{\vdash A, \top, \Gamma}\,(\top) \quad \overset{\pi}{\vdash A^\perp, \Delta}}{\underset{\rho}{\vdash \top, \Gamma, \Delta}}\,(c) \qquad \phi = \cfrac{}{\underset{\rho}{\vdash \top, \Gamma, \Delta}}\,(\top)$$

We have $n = 0$ and $m(c) = 2 \times m(A) \times m(\Gamma, \Delta) > 0 = \sum_{i=1}^{n} m(c_i)$. □

Proposition 2.19. *Cut-elimination is weakly normalizing in* MALL. *Moreover, a proof π can be normalized in at most $m(\pi)$ steps.*

Proof. The mass is a norm of normalization for a strategy consisting in eliminating a top-most *cut*-rule. Consider such a rule c in a given proof π, and apply any $\xrightarrow{\beta}$ step on it, yielding a proof ϕ. By Lemma 2.18, the sum of the masses of the resulting *cut*-rules c_i is smaller than $m(c)$, while the masses of other *cut*-rules are unaffected – because the mass is defined locally and there is no *cut*-rule above c, so no *cut*-rule to be duplicated or erased. Thus $m(\phi) = m(\pi) - m(c) + \sum_{i=1}^{n} m(c_i) < m(\pi)$. We conclude by induction hypothesis. □

Remark that the core of the proof is quite original as we consider only masses of formulas and sequents, and nothing about the surrounding proof. These quantities are usually not good for cut-elimination, and seems hard to come by without a linear point of view. It is possible to extend this proof in presence of exponentials by considering exponential trees and adding in the mass of the cut formula the size of its exponential tree.

Transforming this proof into one of strong normalization of $\xrightarrow{\beta} \cdot \overset{r^*}{\vdash\!\dashv}$, still in MALL, is not difficult. It suffices to keep the mass but to modify the quantity associated to a proof, so as to get a reduction during any $\xrightarrow{\overline{\beta}}$ step and not only a top-most one. As we keep the mass, our measure is still preserved by $\overset{r}{\vdash\!\dashv}$.

Definition 2.20. In a proof π, we define a partial order $\prec$ between *cut*-rules by $c' \prec c$ when c is in the sub-proof of π of root c'. This relation is reflexive: $c \prec c$.

The **density** $d(c)$ of a *cut*-rule c in a proof π is defined as $d(c) = \sum_{c' \prec c} m(c')$.

The **density** $d(\pi)$ of a proof π is the multiset of the density of its *cut*-rules.

Lemma 2.21. *If $\tau \xrightarrow{\overline{\beta}} \phi$ then $d(\tau) > d(\phi)$.*

Proof. Using Lemma 2.18, each step replaces a *cut*-rule with *cut*-rules whose sum of masses is smaller. Other masses stay the same as the only modified sequents are not the conclusion sequent of other *cut*-rules than the one eliminated. The only reduction where there can be more *cut*-rules below a given one, notwithstanding the replacement of the eliminated one, is when a sub-proof containing a *cut*-rule is duplicated in a $\& - cut$ commutative case. In such a case, the only *cut*-rules not keeping their densities are c the eliminated one – yielding two *cut*-rules c_1 and c_2 of smaller mass – and the *cut*-rules above it, by definition of $\prec$. Any duplicated *cut*-rule goes from a density $\alpha + m(c)$ to two *cut*-rules of smaller densities $\alpha + m(c_1)$ and $\alpha + m(c_2)$. Thus, we only replace *cut*-rules by couples of smaller ones, and $d(\tau) > d(\phi)$. ☐

Lemma 2.22. *If $\pi \overset{r}{\vdash\!\dashv} \pi'$, then $d(\pi) = d(\pi')$.*

Proof. As rule commutations act below a cut-free proof, they cannot erase nor duplicate *cut*-rules. As a rule commutation only changes the sequents between the rules it commutes, it does not change the sequent below any *cut*-rule, and in particular does not modify the mass of any *cut*-rule. Thence, it preserves the density of all *cut*-rules, which depends only on the masses of the *cut*-rules. ☐

Corollary 2.23. *The relation $\xrightarrow{\overline{\beta}} \cdot \overset{r^*}{\vdash\!\dashv}$ is strongly normalizing in MALL.*

Proof. By Lemmas 2.21 and 2.22, a step of $\xrightarrow{\overline{\beta}}$ decreases the density of a proof while one of $\overset{r}{\vdash\!\dashv}$ preserves it. Hence, a step of $\xrightarrow{\overline{\beta}} \cdot \overset{r^*}{\vdash\!\dashv}$ strictly decreases the density, ensuring termination. ☐

As we will see in the next sections, proving strong normalization of $\xrightarrow{\overline{\beta}}$ not only up to rule commutation but also up to $cut - cut$ commutation necessitates more work.

2.3.3 Strong normalization in MALL^2 and MALL^0

The goal of this section is proving the strong normalization of the relation $\xrightarrow{\overline{\beta}} \cdot (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ in the framework of MALL^0 and MALL^2, so as to apply Theorem 2.6 – we do not prove it for $\mathsf{MALL}^{0,2}$, simply because it is false: see Section 2.3.5 for a counter-example in any sub-system containing the $\top$-, mix_0- and mix_2-rules. Nevertheless, we will place ourselves into the framework of $\mathsf{MALL}^{0,2}$ as both cases follow the same proof scheme. Remark it is quite important here to consider the cut-free commutations $\overset{r}{\vdash\!\dashv}$, and not the $\overset{rc}{\vdash\!\dashv}$ commutations including cut: $\xleftarrow{\overline{\beta}} \subseteq \overset{rc}{\vdash\!\dashv}$, so obviously $\xrightarrow{\overline{\beta}} \cdot \overset{rc}{\vdash\!\dashv}$ cannot be strongly normalizing, even in the simplest sub-systems of linear logic.

We prove the wished result by giving a measure which is preserved by $\overset{r}{\vdash\!\dashv}$ and $\overset{c}{\vdash\!\dashv}$ but decreases during a $\xrightarrow{\overline{\beta}}$ step. This measure is complex, and will need some intermediate definitions.

Definition 2.24. We define the **weight** $w(A)$ of a formula A by induction:

- $w(X^+) = w(X^-) = w(1) = w(\bot) = w(\top) = w(0) = 1$
- $w(A \otimes B) = w(A \parr B) = w(A \oplus B) = w(A) + w(B) + 1$
- $w(A \mathbin{\&} B) = \max(w(A), w(B)) + 1$

We extend this notion to sequents, by defining for a sequent $\vdash \Gamma = \vdash A_1, \ldots, A_n$ that $w(\vdash \Gamma) = \sum_{i=1}^{n} w(A_i)$.

Fact 2.25. *For any formula A, $w(A)$ is a positive integer.*

Proof. By induction on the formula A. □

Definition 2.26. We define the **weight** $w(\pi)$ of a proof π as a natural number by induction:

- $w\left(\dfrac{}{\vdash A^\perp, A}\,{\scriptstyle (ax)}\right) = w(A^\perp) + w(A)$.
- $w\left(\dfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash A^\perp, \Delta}}{\vdash \Gamma, \Delta}\,{\scriptstyle (cut)}\right) = w(\pi_1) + w(\pi_2)$
- $w\left(\dfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,{\scriptstyle (\otimes)}\right) = w(\pi_1) + w(\pi_2) + 1$
- $w\left(\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}\,{\scriptstyle (\parr)}\right) = w(\pi_1) + 1$
- $w\left(\dfrac{}{\vdash 1}\,{\scriptstyle (1)}\right) = 1$
- $w\left(\dfrac{\overset{\pi_1}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,{\scriptstyle (\bot)}\right) = w(\pi_1) + 1$
- $w\left(\dfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Gamma}}{\vdash A \mathbin{\&} B, \Gamma}\,{\scriptstyle (\&)}\right) = \max(w(\pi_1), w(\pi_2)) + 1$
- $w\left(\dfrac{\overset{\pi_1}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}\,{\scriptstyle (\oplus_1)}\right) = w(\pi_1) + w(B) + 1$
- $w\left(\dfrac{\overset{\pi_1}{\vdash B, \Gamma}}{\vdash A \oplus B, \Gamma}\,{\scriptstyle (\oplus_2)}\right) = w(\pi_1) + w(A) + 1$
- $w\left(\dfrac{}{\vdash \top, \Gamma}\,{\scriptstyle (\top)}\right) = 1 + w(\vdash \Gamma)$
- $w\left(\dfrac{\overset{\pi_1}{\vdash \Gamma} \quad \overset{\pi_2}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,{\scriptstyle (mix_2)}\right) = w(\pi_1) + w(\pi_2)$

- $w\left(\dfrac{}{\vdash}\ (mix_0)\right) = 0$

In the above definition, be careful on the following points: the &-rule introduces a max, in a $\oplus_i$-rule the unused formula gives its weight, and the mix_2-rule behave more like the cut-rule than like the $\otimes$-rule. About this last point, it is useful to consider the following commutation:

$$\dfrac{}{\vdash \top, X^-, X^+}\ (\top) \quad \overset{r}{\mapsto} \quad \dfrac{\dfrac{}{\vdash \top}\ (\top) \quad \dfrac{}{\vdash X^-, X^+}\ (ax)}{\vdash \top, X^-, X^+}\ (mix_2)$$

We want the same weight for these two proofs, so that Lemma 2.27 holds. The only possibility is to have the mix_2-rule only summing weights.

Lemma 2.27. *For a proof π of a sequent $\vdash \Gamma$, $w(\pi) \geq w(\vdash \Gamma)$. Furthermore, if π is cut-free, then $w(\pi) = w(\vdash \Gamma)$.*

Proof. This can be easily proven by induction on π. We give here the two most interesting cases, others are easier or similar – the full case study can be found in Appendix A.

If $\pi = \dfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Gamma}}{\vdash A \mathbin{\&} B, \Gamma}\ (\&)$. Remark that π is cut-free if and only if π_1 and π_2 are. Then

$$\begin{aligned}
w(\pi) &= \max(w(\pi_1), w(\pi_2)) + 1 \\
&\geq \max(w(\vdash A, \Gamma), w(\vdash B, \Gamma)) + 1 \text{ by induction, with equality if } \pi \text{ is cut-free} \\
&= \max(w(A) + w(\vdash \Gamma), w(B) + w(\vdash \Gamma)) + 1 \\
&= \max(w(A), w(B)) + w(\vdash \Gamma) + 1 \\
&= w(A \mathbin{\&} B) + w(\vdash \Gamma) \\
&= w(\vdash A \mathbin{\&} B, \Gamma)
\end{aligned}$$

If $\pi = \dfrac{\overset{\pi_1}{\vdash A_1, \Gamma}}{\vdash A_1 \oplus A_2, \Gamma}\ (\oplus_1)$. Remark that π is cut-free if and only if π_1 is. Then

$$\begin{aligned}
w(\pi) &= w(\pi_1) + w(A_2) + 1 \\
&\geq w(\vdash A_1, \Gamma) + w(A_2) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\
&= w(\vdash \Gamma) + w(A_1) + w(A_2) + 1 \\
&= w(\vdash \Gamma) + w(A_1 \oplus A_2) \\
&= w(\vdash A_1 \oplus A_2, \Gamma)
\end{aligned}$$

□

Remark 2.28. There exists a proof π using a cut-rule but whose weight is the one of its sequent. In the following proof, the left sub-tree of the &-rule has weight 5 while its right sub-tree has weight 3, whence the weight of the full proof is $6 = w(\vdash (1 \oplus (1 \oplus 1)) \mathbin{\&} 1)$.

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash 1 \oplus (1 \oplus 1)}\ (\oplus_1) \quad \dfrac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash 1, \bot}\ (\bot) \quad \dfrac{}{\vdash 1}\ (1)}{\vdash 1}\ (cut)}{\vdash (1 \oplus (1 \oplus 1)) \mathbin{\&} 1}\ (\&)$$

This is due to the max when weighting a &-rule, which may "hide" some weight.

Remark 2.29. As a corollary of Lemma 2.27, $\xrightarrow{\eta}$ preserves the weight of a proof, for it transforms a cut-free proof into another one, and weight is defined inductively.

Lemma 2.30. *For any proof π and any sequent $\vdash \Gamma$ appearing in this proof, $w(\pi) \geq w(\vdash \Gamma)$.*

Proof. The sub-proof ϕ of π of root the sequent $\vdash \Gamma$ respects $w(\phi) \geq w(\vdash \Gamma)$ (Lemma 2.27). One can easily prove by induction on π that for any of its sub-proofs τ, $w(\pi) \geq w(\tau)$ (*i.e.* that w is monotonic). Thence, $w(\pi) \geq w(\phi) \geq w(\vdash \Gamma)$. $\square$

Corollary 2.31. *Proofs of null weight are exactly those made exclusively of mix_2- and mix_0-rules.*

Proof. This follows from Facts 1.8 and 2.25 and Lemma 2.30: a proof of null weight has only empty sequents, and therefore is made entirely of mix_2- and mix_0-rules. $\square$

Definition 2.32. A **block** $\mathfrak{B}$ of *cut*-rules in a proof π is a maximal set of consecutive *cut*-rules in π.

We call **measure** $|\mathfrak{B}|$ of a block $\mathfrak{B}$ of *cut*-rules in a proof π the weight of its root *cut*-rule, *i.e.* $|\mathfrak{B}| = \sum_i w(\pi_i)$ where the π_i are the sub-proofs whose conclusions are the premises of the *cut*-rules of $|\mathfrak{B}|$, premises which are by definition not the conclusion of a *cut*-rule.

The **measure** $|c|$ of a *cut*-rule c in a proof π is the measure of the (unique) block it belongs to.

The **measure** $|\pi|$ of a proof π is the multiset of the measures of its *cut*-rules.

For example, consider the following proof π:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax) \quad \dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax) \quad \dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, X^+}\,(cut)}{\vdash X^-, X^+ \oplus 0}\,(\oplus_1)}{\vdash X^-, X^+ \oplus 0}\,(cut) \quad \dfrac{\dfrac{}{\vdash \top, X^+}\,(\top) \quad \dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash \top \,\&\, X^-, X^+}\,(\&)}{\vdash X^-, X^+}\,(cut) \quad \dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax) \quad \dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, X^+}\,(cut)}{\vdash X^-, X^+}\,(cut)$$

It contains two blocks: the blue one $\mathcal{B}_b$ with one *cut*-rule, and the red one $\mathcal{B}_r$ with the other four. The measure of the blue block is the weight of the sub-proof starting from the blue *cut*-rule, namely $|\mathcal{B}_b| = 4$. Meanwhile, the measure of the red block is the weight of the whole proof, $|\mathcal{B}_r| = 15$. Therefore, $|\pi| = \{4; 15; 15; 15; 15\}$.

Remark 2.33. A block $\mathfrak{B}$ of n *cut*-rules in a proof π has its measure $|\mathfrak{B}|$ appearing n times in $|\pi|$, once for each of its *cut*-rules.

Lemma 2.34. *If τ and ϕ are proofs of $\mathsf{MALL}^{0,2}$ such that $\tau \xrightarrow{\overline{\beta}} \phi$, then $|\tau| > |\phi|$ unless this step is a $cut - mix_2 - 1$ (resp. $cut - mix_2 - 2$) commutative step where the sub-proof above the right (resp. left) premise of the mix_2-rule is a proof made entirely of mix_2- and mix_0-rules, in which case $|\tau| = |\phi|$.*

In particular, for τ and ϕ proofs in either MALL^2 or MALL^0, if $\tau \xrightarrow{\overline{\beta}} \phi$ then $|\tau| > |\phi|$.

Proof. It suffices to compute the measure before and after each cut-elimination step. To begin with, let us consider some general arguments. Blocks "enough below" the rules involved in the $\xrightarrow{\overline{\beta}}$ step can be easily handled. These blocks are those in the external context of the reduction step, and not containing the *cut*-rule involved in the $\xrightarrow{\overline{\beta}}$ step (as the bottom rule of a cut-elimination step is always a *cut*-rule). It suffices to prove the weight of all sub-proofs does not increase when applying a $\xrightarrow{\overline{\beta}}$ step, to obtain that the measures of these blocks do not increase. Remark that it is particularly important to have a max in the inductive definition of the weight in case of a &-rule; a $\& - cut$ commutative reduction may duplicate a sub-proof, but leaves the weight of the proof unchanged.

Symmetrically, for blocks "enough above" the rules of the $\xrightarrow{\overline{\beta}}$ step, there is nothing to do, for their sub-proofs remain the sames. These blocks are those above the rules involved in the reduction step, not having a *cut*-rule directly above one of the rules of the $\xrightarrow{\overline{\beta}}$ step. They may be duplicated or erased, but as their measure is less than the one of the block containing the reduced *cut*-rule, it suffices to prove the measure of this last block decreases to obtain the result.

Hence, we only have to consider the block containing the *cut*-rule, as well as blocks just above the other rules involved in the $\xrightarrow{\overline{\beta}}$ step. That the weight of a proof does not increase during a $\xrightarrow{\overline{\beta}}$ step can be easily checked, as it has an inductive definition.

In all cases, we will call $\mathfrak{B}_p^\tau$ the block in τ containing the *cut*-rule c on which the $\overline{\beta}$ step acts, which is composed of $n+1$ *cut*-rules.

ax key case. We have $\tau = \dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \overset{\pi}{\vdash A, \Gamma}}{\vdash A, \Gamma}\,(cut)}{\rho}$ and $\phi = \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\rho}$.

Call $\mathfrak{B}_p^\phi$ the block in ϕ corresponding to the $\mathfrak{B}_p^\tau$, which has corresponding *cut*-rules excepted c the eliminated one (it might be empty, in which case the result holds). The weight of sub-proofs decreases during the reduction step, for we removed an *ax*-rule. By the preliminary remarks of our proof, we only have to compare the measures of $\mathfrak{B}_p^\tau$ and $\mathfrak{B}_p^\phi$.

We have $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_p^\phi| + w(A^\perp) + w(A)$, as we remove an *ax*-rule. Therefore, in τ we had $n+1$ *cut*-rules of weight $|\mathfrak{B}_p^\phi| + w(A^\perp) + w(A)$ corresponding in ϕ to n *cut*-rules of weight $|\mathfrak{B}_p^\phi|$. Whence, $|\tau| > |\phi|$.

$⅋ - \otimes$ *key case.* Here $\tau = \dfrac{\dfrac{\dfrac{\overset{\pi_1}{\vdash A, \Gamma} \quad \overset{\pi_2}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \qquad \dfrac{\overset{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash B^\perp ⅋ A^\perp, \Sigma}\,(⅋)}{\vdash \Gamma, \Delta, \Sigma}\,(cut)}{\rho}$ while

$$\phi = \dfrac{\dfrac{\overset{\pi_1}{\vdash A, \Gamma} \qquad \dfrac{\overset{\pi_2}{\vdash B, \Delta} \quad \overset{\pi_3}{\vdash B^\perp, A^\perp, \Sigma}}{\vdash A^\perp, \Delta, \Sigma}\,(cut)}{\vdash \Gamma, \Delta, \Sigma}\,(cut)}{\rho}.$$

Call $\mathfrak{B}_i$ the (possibly empty) block at the root of π_i, containing n_i *cut*-rules, for $i \in \{1; 2; 3\}$, and $\mathfrak{B}_p^\phi$ the block in ϕ containing the two produced *cut*-rules. Blocks $\mathfrak{B}_p^\tau$, $\mathfrak{B}_1$, $\mathfrak{B}_2$ and $\mathfrak{B}_3$ are merged into the same block $\mathfrak{B}_p^\phi$.

We compute $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_1| + |\mathfrak{B}_2| + |\mathfrak{B}_3| + 2 + \alpha$, with α the sum of the weight of the other sub-proofs leading to the block $\mathfrak{B}_p^\tau$ than the two of roots the eliminated $⅋$ and $\otimes$-rules. Also $|\mathfrak{B}_p^\phi| = |\mathfrak{B}_1| + |\mathfrak{B}_2| + |\mathfrak{B}_3| + \alpha$ (for we remove the $⅋$ and $\otimes$-rules). Therefore, in τ we had $n+1$ *cut*-rules of weight $|\mathfrak{B}_1| + |\mathfrak{B}_2| + |\mathfrak{B}_3| + 2 + \alpha$, and n_i *cut*-rules of weight $|\mathfrak{B}_i|$, while in ϕ we have $n + n_1 + n_2 + n_3 + 2$ *cut*-rules of weight $|\mathfrak{B}_1| + |\mathfrak{B}_2| + |\mathfrak{B}_3| + \alpha$. Whence, $|\tau| > |\phi|$.

$\& - \oplus_1$ *key case.* Here $\tau = \dfrac{\dfrac{\dfrac{\pi_1}{\vdash A_1, \Gamma} \quad \dfrac{\pi_2}{\vdash A_2, \Gamma}}{\vdash A_1 \& A_2, \Gamma}(\&) \quad \dfrac{\dfrac{\pi_3}{\vdash A_2^\perp, \Delta}}{\vdash A_2^\perp \oplus A_1^\perp, \Delta}(\oplus_1)}{\dfrac{\vdash \Gamma, \Delta}{\rho}}(cut)$ while

$$\phi = \dfrac{\dfrac{\pi_2}{\vdash A_2, \Gamma} \quad \dfrac{\pi_3}{\vdash A_2^\perp, \Delta}}{\dfrac{\vdash \Gamma, \Delta}{\rho}}(cut).$$

Call $\mathfrak{B}_i$ the (possibly empty) block at the root of π_i, containing n_i *cut*-rules, for $i \in \{1; 2; 3\}$, and $\mathfrak{B}_p^\phi$ the block in ϕ containing the produced *cut*-rule. Blocks $\mathfrak{B}_p^\tau$, $\mathfrak{B}_2$ and $\mathfrak{B}_3$ are merged into the same block $\mathfrak{B}_p^\phi$, while $\mathfrak{B}_1$ is deleted.

We compute $|\mathfrak{B}_p^\tau| = \max(|\mathfrak{B}_1|, |\mathfrak{B}_2|) + |\mathfrak{B}_3| + 2 + w(A_1^\perp) + \alpha$, with α the sum of the weight of the other sub-proofs leading to the block $\mathfrak{B}_p^\tau$ than the two of roots the eliminated $\&$ and $\oplus_1$-rules. Also $|\mathfrak{B}_p^\phi| = |\mathfrak{B}_2| + |\mathfrak{B}_3| + \alpha$. Therefore, in τ we had $n + 1$ *cut*-rules of weight $\max(|\mathfrak{B}_1|, |\mathfrak{B}_2|) + |\mathfrak{B}_3| + 2 + w(A_1^\perp) + \alpha$, and n_i *cut*-rules of weight $|\mathfrak{B}_i|$, while in ϕ we have $n + n_2 + n_3 + 1$ *cut*-rules of weight $|\mathfrak{B}_2| + |\mathfrak{B}_3| + \alpha$. Whence, $|\tau| > |\phi|$.

$\& - \oplus_2$ *key case.* This case is very similar to the $\& - \oplus_1$ key case.

$\perp - 1$ *key case.* Here $\tau = \dfrac{\dfrac{}{\vdash 1}(1) \quad \dfrac{\dfrac{\pi}{\vdash \Gamma}}{\vdash \Gamma, \perp}(\perp)}{\dfrac{\vdash \Gamma}{\rho}}(cut)$ and $\phi = \dfrac{\dfrac{\pi}{\vdash \Gamma}}{\rho}$.

Call $\mathfrak{B}_p^\phi$ the block in ϕ corresponding to the $\mathfrak{B}_p^\tau$, which has corresponding *cut*-rules excepted for the eliminated one, and maybe additional rules from a block $\mathfrak{B}$ just above the $\perp$-rule in τ.

We have $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_p^\phi| + 2$, as we remove the $\perp$ and 1-rules, and $|\mathfrak{B}| \leq |\mathfrak{B}_p^\phi|$. Therefore, in τ we had $n+1$ *cut*-rules of weight $|\mathfrak{B}_p^\phi| + 2$ and k rules of weight $|\mathfrak{B}| \leq |\mathfrak{B}_p^\phi|$, corresponding in ϕ to $n+k$ *cut*-rules of weight $|\mathfrak{B}_p^\phi|$. Whence, $|\tau| > |\phi|$.

$⅋ - cut$ *commutative case.* Here $\tau = \dfrac{\dfrac{\dfrac{\pi_1}{\vdash A, B, C, \Gamma}}{\vdash A, B ⅋ C, \Gamma}(⅋) \quad \dfrac{\pi_2}{\vdash A^\perp, \Delta}}{\dfrac{\vdash B ⅋ C, \Gamma, \Delta}{\rho}}(cut)$ while

$$\phi = \dfrac{\dfrac{\dfrac{\pi_1}{\vdash A, B, C, \Gamma} \quad \dfrac{\pi_2}{\vdash A^\perp, \Delta}}{\vdash B, C, \Gamma, \Delta}(cut)}{\dfrac{\vdash B ⅋ C, \Gamma, \Delta}{\rho}}(⅋).$$

Call $\mathfrak{B}_1$ the (possibly empty) block at the root of π_1, containing n_1 *cut*-rules, $\mathfrak{B}_a$ the sub-block in ϕ (and τ) containing c and the *cut*-rules directly above its right premise, and $\mathfrak{B}_b^\phi$ the (possibly empty) block in ϕ containing the rules corresponding to the ones of $\mathfrak{B}_p^\tau \backslash \mathfrak{B}_a$. The blocks $\mathfrak{B}_p^\tau$ and $\mathfrak{B}_1$ are mapped to the blocks $\mathfrak{B}_b^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$.

The difference between $\mathfrak{B}_p^\tau$ and $\mathfrak{B}_1$ in τ and $\mathfrak{B}_b^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$ in ϕ is that $\mathfrak{B}_a$ (including c) moved from $\mathfrak{B}_p^\tau$ to $\mathfrak{B}_1$. We have $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_b^\phi|$ (provided $\mathfrak{B}_b^\phi$ is not empty) and $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_a \cup \mathfrak{B}_1| + 1 >$

$|\mathfrak{B}_a \cup \mathfrak{B}_1|$, as the weight of the commuted $⅋$-rule is included in $|\mathfrak{B}_p^\tau|$ but not $|\mathfrak{B}_a \cup \mathfrak{B}_1|$. Thence, $|\tau| > |\phi|$.

$\otimes - cut - 1$ *commutative case.* Here $\tau = \dfrac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_2}{\vdash C, \Delta}}{\vdash A, B \otimes C, \Gamma, \Delta}\,(\otimes) \qquad \overset{\pi_3}{\vdash A^\perp, \Sigma}}{\dfrac{\vdash B \otimes C, \Gamma, \Delta, \Sigma}{\rho}}\,(cut)$ and

$$\phi = \dfrac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_3}{\vdash A^\perp, \Sigma}}{\vdash B, \Gamma, \Sigma}\,(cut) \qquad \overset{\pi_2}{\vdash C, \Delta}}{\dfrac{\vdash B \otimes C, \Gamma, \Delta, \Sigma}{\rho}}\,(\otimes).$$

Call $\mathfrak{B}_1$ the (possibly empty) block at the root of π_1, $\mathfrak{B}_a$ the sub-block in ϕ (and τ) containing c and the *cut*-rules directly above its right premise, and $\mathfrak{B}_b^\phi$ the (possibly empty) block in ϕ containing the rules corresponding to the ones of $\mathfrak{B}_p^\tau \backslash \mathfrak{B}_a$. The blocks $\mathfrak{B}_p^\tau$ and $\mathfrak{B}_1$ are mapped to the blocks $\mathfrak{B}_b^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$.

The difference between $\mathfrak{B}_p^\tau$ and $\mathfrak{B}_1$ in τ and $\mathfrak{B}_b^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$ in ϕ is that $\mathfrak{B}_a$ (including c) moved from $\mathfrak{B}_p^\tau$ to $\mathfrak{B}_1$. We have $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_b^\phi|$ (provided $\mathfrak{B}_b^\phi$ is not empty) and $|\mathfrak{B}_p^\tau| = |\mathfrak{B}_a \cup \mathfrak{B}_1| + w(\pi_2) + 1 > |\mathfrak{B}_a \cup \mathfrak{B}_1|$, as the weight of the commuted $\otimes$-rule is included in $|\mathfrak{B}_p^\tau|$ but not $|\mathfrak{B}_a \cup \mathfrak{B}_1|$. Thence, $|\tau| > |\phi|$.

$\otimes - cut - 2$ *commutative case.* This case is very similar to the $\otimes - cut - 1$ commutative case.

$\& - cut$ *commutative case.* Here $\tau = \dfrac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_2}{\vdash A, C, \Gamma}}{\vdash A, B \,\&\, C, \Gamma}\,(\&) \qquad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\dfrac{\vdash B \,\&\, C, \Gamma, \Delta}{\rho}}\,(cut)$ and

$$\phi = \dfrac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \quad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash B, \Gamma, \Delta}\,(cut) \qquad \dfrac{\overset{\pi_2}{\vdash A, C, \Gamma} \quad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash C, \Gamma, \Delta}\,(cut)}{\dfrac{\vdash B \,\&\, C, \Gamma, \Delta}{\rho}}\,(\&).$$

Call $\mathfrak{B}_i$ the (possibly empty) block at the root of π_i, $i \in \{1; 2\}$, $\mathfrak{B}_a$ the sub-block in ϕ (and τ) containing c and the *cut*-rules directly above its right premise, and $\mathfrak{B}_b^\phi$ the (possibly empty) block in ϕ containing the rules corresponding to the ones of $\mathfrak{B}_p^\tau \backslash \mathfrak{B}_a$. The blocks $\mathfrak{B}_p^\tau$, $\mathfrak{B}_1$, $\mathfrak{B}_2$ are mapped to the blocks $\mathfrak{B}_b^\phi$, $\mathfrak{B}_a \cup \mathfrak{B}_1$, $\mathfrak{B}_a \cup \mathfrak{B}_2$.

As in the previous cases, some *cut*-rules (including at least c) strictly lose measure by going from $\mathfrak{B}_p^\tau$ to $\mathfrak{B}_a \cup \mathfrak{B}_1$ and $\mathfrak{B}_a \cup \mathfrak{B}_2$ (both copies separately weight less than the original), and the duplicated blocks (in π_3) are of weight less than the weight of $\mathfrak{B}_p^\tau$. So, again, we replace one or several measures by strictly smaller measures, hence $|\tau| > |\phi|$.

$\otimes - cut - 1$, $\otimes - cut - 2$, $\oplus_i - cut$ *and* $\bot - cut$ *commutative cases.* These cases are quite similar to the $⅋ - cut$ commutative case.

$\top - cut$ *commutative case.* Here $\tau = \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \overset{\pi}{\vdash A^\perp, \Delta}}{\dfrac{\vdash \top, \Gamma, \Delta}{\rho}}\,(cut)$ whereas $\phi = \dfrac{\dfrac{}{\vdash \top, \Gamma, \Delta}\,(\top)}{\rho}$.

The weights of sub-proofs do not increase during the reduction step, using Lemma 2.27. As we remove at least the *cut*-rule c, and the measures of all blocks do not increase, the result follows as in the *ax* key case.

$mix_2 - cut - 1$ commutative case. Here
$$\tau = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, \Gamma} \qquad \overset{\pi_2}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \overset{\pi_3}{\vdash A^{\perp}, \Sigma}}{\cfrac{\vdash \Gamma, \Delta, \Sigma}{\rho}}\,(cut)$$
and
$$\phi = \cfrac{\cfrac{\overset{\pi_1}{\vdash A, \Gamma} \qquad \overset{\pi_3}{\vdash A^{\perp}, \Sigma}}{\vdash \Gamma, \Sigma}\,(cut) \qquad \overset{\pi_2}{\vdash \Delta}}{\cfrac{\vdash \Gamma, \Delta, \Sigma}{\rho}}\,(mix_2).$$

As in the $\otimes - cut - 1$ commutative case, call $\mathfrak{B}_1$ the (possibly empty) block at the root of π_1, $\mathfrak{B}_a$ the sub-block in ϕ (and τ) containing c and the *cut*-rules directly above its right premise, and $\mathfrak{B}_b^{\phi}$ the (possibly empty) block in ϕ containing the rules corresponding to the ones of $\mathfrak{B}_p^{\tau} \backslash \mathfrak{B}_a$. The blocks $\mathfrak{B}_p^{\tau}$ and $\mathfrak{B}_1$ are mapped to the blocks $\mathfrak{B}_b^{\phi}$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$.

The difference between $\mathfrak{B}_p^{\tau}$ and $\mathfrak{B}_1$ in τ and $\mathfrak{B}_b^{\phi}$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$ in ϕ is that $\mathfrak{B}_a$ (including c) moved from $\mathfrak{B}_p^{\tau}$ to $\mathfrak{B}_1$. We have $|\mathfrak{B}_p^{\tau}| = |\mathfrak{B}_b^{\phi}|$ (provided $\mathfrak{B}_b^{\phi}$ is not empty) and $|\mathfrak{B}_p^{\tau}| = |\mathfrak{B}_a \cup \mathfrak{B}_1| + w(\pi_2)$, as the weight of the proof π_2 is included in $|\mathfrak{B}_p^{\tau}|$ but not $|\mathfrak{B}_a \cup \mathfrak{B}_1|$. If $w(\pi_2) > 0$, which is equivalent to π_2 not being made exclusively of mix_2- and mix_0-rules by Corollary 2.31, then $|\mathfrak{B}_p^{\tau}| > |\mathfrak{B}_a \cup \mathfrak{B}_1|$, thence, $|\tau| > |\phi|$. Otherwise, $|\tau| = |\phi|$.

$mix_2 - cut - 2$ commutative case. This case is very similar to the $mix_2 - cut - 1$ commutative case. ☐

Lemma 2.35. *If $\pi \overset{c}{\vdash\!\dashv} \pi'$ then $|\pi| = |\pi'|$.*

Proof. Because a $cut - cut$ commutation does not modify the measure of the block of *cut*-rules it is done in, nor the weight of the sub-proof starting from this block. ☐

Lemma 2.36. *If $\pi \overset{r}{\vdash\!\dashv} \pi'$, then $|\pi| = |\pi'|$.*

Proof. As rule commutations act below a cut-free proof, it is enough to prove they preserve the weight of the minimal sub-proof containing them. This is the case by Lemma 2.27, for two cut-free proofs on a given sequent share the same weight. ☐

Remark 2.37. For Lemma 2.36 to hold, we need that rule commutations are applied only when there is no *cut*-rule above. Otherwise, a sub-proof created (resp. destructed) by a $\top - \otimes$ commutation may increase (resp. decrease) the weight and the measure of the proof. Similarly, a sub-proof duplicated (resp. superimposed) by a $\& - \otimes$ commutation may increase (resp. decrease) the measure of the proof, even if its weight remains unchanged. Other rule commutations pose no such problem.

Proposition 2.38. *In MALL^2 and MALL^0, the relation $\xrightarrow{\overline{\beta}} \cdot (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ is strongly normalizing.*

Proof. By Lemmas 2.34, 2.35 and 2.36, a step of $\xrightarrow{\overline{\beta}}$ decreases the measure of a proof while one of $\overset{r}{\vdash\!\dashv}$ or $\overset{c}{\vdash\!\dashv}$ preserves it. Hence, a step of $\xrightarrow{\overline{\beta}} \cdot (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ strictly decreases the measure, ensuring termination. ☐

It is not possible to find a measure for proofs in $\mathsf{MALL}^{0,2}$ which is preserved by $\overset{r}{\vdash\!\dashv}$ and $\overset{c}{\vdash\!\dashv}$ and strictly decreasing by $\xrightarrow{\overline{\beta}}$, because the generalization of Proposition 2.38 to $\mathsf{MALL}^{0,2}$ is false – see Section 2.3.5 for a counter-example. What can still be proved (with the very same proof as we have here) is that, in case of an infinite reduction, one has a suffix with an infinity of $cut - mix_2$ cut-elimination commutative cases where one of the premise of the mix_2-rule is of null weight. Similarly, adding the $\emptyset$-rule cannot be done here, for it allows proving the empty sequent as does the mix_0-rule (so Lemma 2.34 fails in this generalized framework), but it can be added if the mix_2-rule has been removed. In such a case, one poses $w\left(\frac{}{\vdash \Gamma}\ (\emptyset)\right) = w(\vdash \Gamma)$. The problem is harder for the $\cup$-rule: a $\cup - cut$ commutative case always preserves the measure of a proof.

2.3.4 Strong normalization in $\mathsf{MALL}^{0,2}$

As written in the previous section, $\xrightarrow{\overline{\beta}} \cdot(\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ is not strongly normalizing in $\mathsf{MALL}^{0,2}$. We set here $\overset{r\backslash\top}{\rightsquigarrow}$ the rule commutations excepted $\top$-commutations in the direction where they "create" &- $\otimes$- and mix_2- rules, *i.e.* without the $C^{\&}_{\top}$, $C^{\otimes}_{\top}$ and $C^{mix_2}_{\top}$ commutations (but with those in the other direction, namely with $C^{\top}_{\&}$, $C^{\top}_{\otimes}$ and $C^{\top}_{mix_2}$). Here, we prove that $\xrightarrow{\overline{\beta}} \cdot(\overset{r\backslash\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$ is strongly normalizing, thanks to the results of the previous section. To do so, we define a second measure on proofs of $\mathsf{MALL}^{0,2}$.

Definition 2.39. We call mix**-measure** $|\mathfrak{B}|_{mix}$ of a block $\mathfrak{B}$ of cut-rules in a proof π the sum for all slices s containing $\mathfrak{B}$ of the number of mix_0-rules above $\mathfrak{B}$ in this slice s.

The mix**-measure** $|c|_{mix}$ of a cut-rule c in a proof π is the measure of the (unique) block it belongs to.

The mix**-measure** $|\pi|_{mix}$ of a proof π is the multiset of the mix-measures of its cut-rules.

We prove here that the lexicographic order $(|\cdot|, |\cdot|_{mix})$ is a decreasing measure for a $\xrightarrow{\overline{\beta}} \cdot(\overset{r\backslash\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$ step, allowing us to conclude.

Lemma 2.40. *Let π and ϕ be proofs in $\mathsf{MALL}^{0,2}$.*

(i) *If $\pi \overset{om}{\rightsquigarrow} \phi$ then $|\pi| = |\phi|$.*

(ii) *If $\pi \xrightarrow{\overline{\beta}} \phi$ where this step is a $cut - mix_2 - 1$ (resp. $cut - mix_2 - 2$) commutative step where the sub-proof above the right (resp. left) premise of the mix_2-rule is a proof made entirely of mix_2- and mix_0-rules, then $|\pi|_{mix} > |\phi|_{mix}$.*

(iii) *If $\pi \overset{c}{\vdash\!\dashv} \phi$ then $|\pi|_{mix} = |\phi|_{mix}$.*

(iv) *If $\pi \overset{r\backslash\top}{\rightsquigarrow} \phi$ then $|\pi|_{mix} \geq |\phi|_{mix}$.*

(v) *If $\pi \overset{om}{\rightsquigarrow} \phi$ then $|\pi|_{mix} \geq |\phi|_{mix}$.*

Proof.

(i) Simply because this operation replaces a sub-proof by another of the same weight.

(ii) We proceed as in the proof of Lemma 2.34. Such an operation does not modify the number of slice nor the number of mix_0-rules by slice, hence the mix-measure of blocks "below or above enough". Set:

$$\pi = \dfrac{\dfrac{\dfrac{\tau_1}{\vdash A,\Gamma} \quad \dfrac{\tau_2}{\vdash \Delta}}{\vdash A,\Gamma,\Delta}\,(mix_2) \qquad \dfrac{\tau_3}{\vdash A^\perp,\Sigma}}{\dfrac{\vdash \Gamma,\Delta,\Sigma}{\mu}}\,(cut) \quad \text{and} \quad \phi = \dfrac{\dfrac{\dfrac{\tau_1}{\vdash A,\Gamma} \quad \dfrac{\tau_3}{\vdash A^\perp,\Sigma}}{\vdash \Gamma,\Sigma}\,(cut) \qquad \dfrac{\tau_2}{\vdash \Delta}}{\dfrac{\vdash \Gamma,\Delta,\Sigma}{\mu}}\,(mix_2)$$

Call $\mathfrak{B}_1$ the (possibly empty) block at the root of τ_1, $\mathfrak{B}_a$ the sub-block in π (and ϕ) containing c and the *cut*-rules directly above its right premise (in τ_3), $\mathfrak{B}^\pi$ the block containing c in π and $\mathfrak{B}^\phi$ the (possibly empty) block in ϕ containing the rules corresponding to the ones of $\mathfrak{B}^\pi \backslash \mathfrak{B}_a$. The blocks $\mathfrak{B}^\pi$ and $\mathfrak{B}_1$ are mapped to the blocks $\mathfrak{B}^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$.

The difference between $\mathfrak{B}^\pi$ and $\mathfrak{B}_1$ in π and $\mathfrak{B}^\phi$ and $\mathfrak{B}_a \cup \mathfrak{B}_1$ in ϕ is that $\mathfrak{B}_a$ (including c) moved from $\mathfrak{B}^\pi$ to $\mathfrak{B}_1$. Name n the number of mix_0-rules of τ_2 multiplied by the number of slices of the sub-proof of root $\mathfrak{B}^\pi$; $n > 0$ as τ_2 cannot be made of only mix_2-rules and in above c in at least one slice. We have $|\mathfrak{B}^\pi|_{mix} = |\mathfrak{B}^\phi|_{mix}$ (provided $\mathfrak{B}^\phi$ is not empty) and $|\mathfrak{B}^\pi|_{mix} = |\mathfrak{B}_a \cup \mathfrak{B}_1|_{mix} + n > |\mathfrak{B}_a \cup \mathfrak{B}_1|_{mix}$, as the mix_0-rules of the proof τ_2 is included in $|\mathfrak{B}^\pi|_{mix}$ but not $|\mathfrak{B}_a \cup \mathfrak{B}_1|_{mix}$. Thence, the *mix*-measure of no *cut*-rule is modified by the rewriting step, except for those in $\mathfrak{B}_a$ (including c) which lose value n; meaning $|\pi|_{mix} > |\phi|_{mix}$.

(iii) Because we consider *cut*-blocks, and a $cut - cut$ commutation does not modify the *mix*-measure of the block of *cut*-rules it is done in, nor the number of slices or the number and place of mix_0-rule in any slice.

(iv) The only rule commutation that increase the number of slices is a $C^{\&}_{\top}$ commutation, which is not considered here. The only rule commutations that may increase the number of mix_0-rules in a slice are $C^{\otimes}_{\top}$ and $C^{mix_2}_{\top}$ commutations (in case the sub-proof introduced contains a mix_0-rule), which are also not considered here. The result follows.

(v) A $\overset{om}{\rightsquigarrow}$ step does not modify the number of slices, and can only reduce the number of mix_0-rules above a block in a given slice.

∎

Proposition 2.41. *In* $\mathsf{MALL}^{0,2}$*, the relation* $\xrightarrow{\overline{\beta}} \cdot (\overset{r\backslash\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$ *is strongly normalizing.*

Proof. By Lemmas 2.34, 2.35, 2.36 and 2.40, a step of $\xrightarrow{\overline{\beta}}$ decreases the lexicographic measure $(|\cdot|, |\cdot|_{mix})$ of a proof while one of $\overset{r\backslash\top}{\rightsquigarrow}$, $\overset{c}{\vdash\!\dashv}$ or $\overset{om}{\rightsquigarrow}$ decreases or preserves it. Hence, this measure is strictly decreasing for a step of $\xrightarrow{\overline{\beta}} \cdot (\overset{r\backslash\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$, ensuring termination. ∎

Remark 2.42. In the definition of $\overset{r\backslash\top}{\rightsquigarrow}$, one could allow more commutations, and remove only the $C^{\otimes}_{\top}$ and $C^{mix_2}_{\top}$ commutations where there is a mix_0-rule on the premise of a mix_2-rule up to rule commutation. One can also surely keep all $C^{\&}_{\top}$ commutations, by considering some kind of pseudo-slices where a &-sub-formula in the context of a $\top$ creates two such slices, and similarly for a &-sub-formula in the non kept formula of a $\oplus_i$-rule. We choose not to do so as this complicates the proof of (iv) of Lemma 2.40, and our definition of $\overset{r\backslash\top}{\rightsquigarrow}$ is enough for our goals.

As a noteworthy corollary, we get strong normalization of $\xrightarrow{\overline{\beta}}$ in $\mathsf{MALL}^{0,2}$.

Corollary 2.43. *In* $\mathsf{MALL}^{0,2}$*, cut-elimination* $\xrightarrow{\beta}$ *is weakly normalizing, and* $\xrightarrow{\overline{\beta}}$ *is strongly normalizing.*

Proof. Using Proposition 2.38, one can reach a normal form for $\xrightarrow{\overline{\beta}}$. This is also a normal form for $\overset{c}{\vdash\!\dashv}$ thanks to Fact 1.13. $\square$

2.3.5 Strong normalization in full linear logic

In the full logic, $\xrightarrow{\overline{\beta}} \cdot (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ is not strongly normalizing, so Theorem 2.6 cannot be applied. Here are some counter-examples, showing in which cases strong normalization fails.

Counter-example with $\top - mix_2$ commutation in the presence of mix_0 While this relation is strongly normalizing in MALL^2 and MALL^0, it is not in $\mathsf{MALL}^{0,2}$. Consider the following proofs:

$$\pi_1 = \dfrac{\dfrac{}{\vdash \top, X^+}(\top) \qquad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut)$$

$$\pi_2 = \dfrac{\dfrac{\dfrac{}{\vdash \top, X^+}(\top) \quad \dfrac{}{\vdash}(mix_0)}{\vdash \top, X^+}(mix_2) \qquad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut)$$

$$\pi_3 = \dfrac{\dfrac{\dfrac{}{\vdash \top, X^+}(\top) \qquad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut) \quad \dfrac{}{\vdash}(mix_0)}{\vdash \top, X^+}(mix_2)$$

One can remark that $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. As π_1 is a sub-proof of π_3, one can go on ad nauseam. The problem here is that the $\top - mix_2$ commutation can introduce some "noise" with the mix_0-rule: while we can apply an infinity of cut-elimination steps, the *cut*-rule is always on the same sequent. A similar example exists by replacing the mix_0-rules with $\emptyset$-rules.

Counter-example with $\top - \cup$ commutation In a sub-system with both $\top$- and $\cup$-rules, the relation $\xrightarrow{\overline{\beta}} \cdot \overset{r^*}{\vdash\!\dashv}$ is also not strongly normalizing. A counter-example is given with the following proofs:

$$\pi_1 = \dfrac{\dfrac{}{\vdash \top, X^+}(\top) \qquad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut)$$

$$\pi_2 = \dfrac{\dfrac{\dfrac{}{\vdash \top, X^+}(\top) \quad \dfrac{}{\vdash \top, X^+}(\top)}{\vdash \top, X^+}(\cup) \qquad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut)$$

$$\pi_3 = \dfrac{\dfrac{\dfrac{}{\vdash \top, X^+}(\top) \quad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut) \qquad \dfrac{\dfrac{}{\vdash \top, X^+}(\top) \quad \dfrac{}{\vdash X^-, X^+}(ax)}{\vdash \top, X^+}(cut)}{\vdash \top, X^+}(\cup)$$

One can remark that $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. As π_1 is a sub-proof of π_3 – it has even been duplicated! – these two steps can be repeated at will.

Counter-example with $\top - ?_c$ and $\top - ?_w$ commutations Even with no optional rules, strong normalization of $\xrightarrow{\overline{\beta}} \cdot \overset{r^*}{\vdash\!\dashv}$ in the framework of exponentials is false. Set π the following proof:

$$\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^- \parr X^+}\,(\parr)}{\vdash !(X^- \parr X^+)}\,(!)$$

We define the following proofs:[2]

$$\phi_1 = \quad \dfrac{\overset{\pi}{\vdash !(X^- \parr X^+)} \qquad \dfrac{}{\vdash ?(X^- \otimes X^+), \top}\,(\top)}{\vdash \top}\,(cut)$$

$$\phi_2 = \quad \dfrac{\overset{\pi}{\vdash !(X^- \parr X^+)} \qquad \dfrac{\dfrac{}{\vdash ?(X^- \otimes X^+), ?(X^- \otimes X^+), \top}\,(\top)}{\vdash ?(X^- \otimes X^+), \top}\,(?_c)}{\vdash \top}\,(cut)$$

$$\phi_3 = \quad \dfrac{\overset{\pi}{\vdash !(X^- \parr X^+)} \qquad \dfrac{\dfrac{\dfrac{}{\vdash ?(X^- \otimes X^+), \top}\,(\top)}{\vdash ?(X^- \otimes X^+), ?(X^- \otimes X^+), \top}\,(?_w)}{\vdash ?(X^- \otimes X^+), \top}\,(?_c)}{\vdash \top}\,(cut)$$

$$\phi_4 = \quad \dfrac{\overset{\pi}{\vdash !(X^- \parr X^+)} \qquad \dfrac{\overset{\pi}{\vdash !(X^- \parr X^+)} \qquad \dfrac{\dfrac{}{\vdash ?(X^- \otimes X^+), \top}\,(\top)}{\vdash ?(X^- \otimes X^+), ?(X^- \otimes X^+), \top}\,(?_w)}{\vdash ?(X^- \otimes X^+), \top}\,(cut)}{\vdash \top}\,(cut)$$

Using one $\top - ?_c$ followed by one $\top - ?_w$ commutations, $\phi_1 \overset{r}{\vdash\!\dashv} \phi_2 \overset{r}{\vdash\!\dashv} \phi_3$. But using a $?_c - !$ then a $?_w - !$ key cut-elimination cases, one find $\phi_3 \xrightarrow{\overline{\beta}} \phi_4 \xrightarrow{\overline{\beta}} \phi_1$. Thus, $\phi_1 \overset{r^*}{\vdash\!\dashv} \cdot \xrightarrow{\overline{\beta}^+} \phi_1$, a counter-example to the strong normalization of $\overset{r^*}{\vdash\!\dashv} \cdot \xrightarrow{\overline{\beta}}$. This example is even more problematic than the previous ones, because it allows to apply an infinity of only key cases of cut-elimination!

Remark a similar example could be given by replacing these two $\top - ?_c$ and $\top - ?_w$ commutations with a $\overset{oe}{\rightsquigarrow}$, giving directly $\phi_1 \overset{oe}{\leftrightsquigarrow} \phi_3$.

[2] In ϕ_3, it does not matter which occurrence of $?(X^- \otimes X^+)$ is weakened in the $?_w$-rule: both possibilities would work.

Counter-example with $\top - \otimes$ commutation in the presence of exponentials Simply forbidding a $\top - ?_c$ and $\top - ?_w$ commutations in the direction where they introduce rules is not enough. One can do a similar reasoning as in the previous example, using the proofs below.

$$\pi_1 = \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\top)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\mathit{cut})$$

$$\pi_2 = \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{\dfrac{\dfrac{}{\vdash !X^-, ?X^+}\,(\mathit{ax})}{\vdash !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash !X^-, ?X^+}\,(?_c)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\otimes)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\mathit{cut})$$

$$\pi_3 = \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{\dfrac{}{\vdash !X^-, ?X^+}\,(\mathit{ax})}{\vdash !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(\otimes)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(?_c)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\mathit{cut})$$

$$\pi_4 = \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{}{\vdash !X^-, ?X^+}\,(\mathit{ax})}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\otimes)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(?_c)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\mathit{cut})$$

$$\pi_5 = \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(\mathit{ax})}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \quad \dfrac{\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{}{\vdash !X^-, ?X^+}\,(\mathit{ax})}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\otimes)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(\mathit{cut})}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(\mathit{cut})}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(?_c)$$

$$\pi_6 = \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{}{\vdash !X^-, ?X^+}\,(ax)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\otimes)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(cut)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(?_c)$$

$$\pi_7 = \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, ?X^+}\,(?_d)}{\vdash !X^-, ?X^+}\,(!) \qquad \dfrac{}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(\top)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(cut)}{\vdash \top, 0 \otimes !X^-, ?X^+, ?X^+}\,(?_w)}{\vdash \top, 0 \otimes !X^-, ?X^+}\,(?_c)$$

One can remark that $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \overset{r}{\vdash\!\dashv} \pi_3 \overset{r}{\vdash\!\dashv} \pi_4 \xrightarrow{\overline{\beta}} \pi_5 \xrightarrow{\overline{\beta}} \pi_6 \overset{r}{\vdash\!\dashv} \pi_7$. As π_1 is a sub-proof of π_7, one can go on eternally, not using any $\top - ?_c$ nor $\top - ?_w$ commutation. The problem here is that the $\top - \otimes$ commutation can introduce a new sub-proof containing $?_c$-rules.

Counter-example with $\top - \exists$ commutation Second-order quantifiers are also problematic in this setting. Consider the following proofs:

$$\pi_1 = \dfrac{\dfrac{}{\vdash \top, \exists X X^+}\,(\top) \qquad \dfrac{}{\vdash \forall X X^-, \top}\,(\top)}{\vdash \top, \top}\,(cut)$$

$$\pi_2 = \dfrac{\dfrac{\dfrac{}{\vdash \top, \exists X X^+}\,(\top)}{\vdash \top, \exists X X^+}\,(\exists) \qquad \dfrac{}{\vdash \forall X X^-, \top}\,(\top)}{\vdash \top, \top}\,(cut)$$

$$\pi_3 = \dfrac{\dfrac{\dfrac{}{\vdash \top, \exists X X^+}\,(\top)}{\vdash \top, \exists X X^+}\,(\exists) \qquad \dfrac{\dfrac{}{\vdash X^-, \top}\,(\top)}{\vdash \forall X X^-, \top}\,(\forall)}{\vdash \top, \top}\,(cut)$$

We have $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \overset{r}{\vdash\!\dashv} \pi_3 \xrightarrow{\overline{\beta}} \pi_1$, with this step being a $\forall - \exists$ key cut-elimination step.

Optional rules Finally, in the above examples, one can replace the $\otimes$-rules with mix_2-rules and $\top$-rules with $\emptyset$ ones to get other counter-examples.

Conjectures Notwithstanding these examples, I think strong normalization holds.

Conjecture 2.44 (Almost strong normalization of cut-elimination)**.** *Consider the framework of full linear logic, with all optional rules. Set* $\overset{\top}{\rightsquigarrow}$ *the rule commutations involving* $\top$*- and* $\emptyset$*-rules directed in* $C_{\_}^{\top}$ *and* $C_{\_}^{\emptyset}$ *in Tables 1.5 to 1.20, and* $\overset{r\setminus\top}{\rightsquigarrow}$ *rule commutations such that in a* $C_{\top}^{\otimes}$ *or* $C_{\emptyset}^{\otimes}$ *commutation, the created proof has no* $?_c$*-,* mix_0*- and* $\cup$*-rule nor a* $\emptyset$*-rule applied on an empty sequent, and without the* $C_{\top}^{?_c}$*,* $C_{\top}^{\exists}$*,* $C_{\top}^{\cup}$*,* $C_{\emptyset}^{?_c}$*,* $C_{\emptyset}^{\exists}$ *and* $C_{\emptyset}^{\cup}$ *commutations. The relation* $(\xrightarrow{\beta} \cup \overset{\top}{\rightsquigarrow} \cup \overset{o}{\rightsquigarrow}) \cdot (\overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv})^*$ *is strongly normalizing, which in particular implies that* $\xrightarrow{\beta} \cdot(\overset{o}{\rightsquigarrow} \cup \overset{\top}{\rightsquigarrow} \cup \overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv})^*$ *is strongly normalizing.*

This result would be enough to apply Theorem 2.7, and a stronger result should not hold looking at our counter-examples. Unfortunately, proving this conjecture seems really hard, and our tools from $\mathsf{MALL}^{0,2}$ badly adapted. Remark the difference between the counter-examples and this conjecture is about what is to be considered a reduction and what a permutation, *i.e.* which relations should be directed and which should be taken in both directions. In case the above conjecture is false, it is possible to regard some more rule commutations as directed to get other weaker conjectures. In particular, if one only wants to apply Theorem 2.7, then the following weaker result would be enough, by removing all $C_{\top}^{-}$ and $C_{\emptyset}^{-}$ commutations in $\overset{r}{\vdash\!\dashv}$ and by considering them as directed instead. What is more, this new conjecture should be provable from [PT10].

Conjecture 2.45. *Consider the framework of full linear logic, with all optional rules. Set* $\overset{\top}{\rightsquigarrow}$ *the rule commutations involving* $\top$*- and* $\emptyset$*-rules directed in* $C_{\_}^{\top}$ *and* $C_{\_}^{\emptyset}$ *in Tables 1.5 to 1.20, and* $\overset{r\setminus\top}{\vdash\!\dashv}$ *rule commutations not involving a* $\top$*- nor* $\emptyset$*-rule. The relation* $(\xrightarrow{\beta} \cup \overset{\top}{\rightsquigarrow} \cup \overset{o}{\rightsquigarrow}) \cdot (\overset{r\setminus\top}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ *is strongly normalizing, which in particular implies that* $\xrightarrow{\beta} \cdot(\overset{o}{\rightsquigarrow} \cup \overset{\top}{\rightsquigarrow} \cup \overset{r\setminus\top}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ *is strongly normalizing.*

Sketch of a proof. We assume here that the reader has their copy of [PT10] at disposition, and that they are familiar with some concepts of linear logic such as proof-nets. If it is not the case, this sketch of proof will surely be unintelligible, and better skipped than read. The idea is to adapt [PT10, Theorem 5.12], which is about strong normalization for (a kind of) proof-nets of the full linear logic, to proofs of the sequent calculus. As the whole point of this long paper was proving this theorem, checking this modification necessitates many steps. Nonetheless, it seems doable by proceeding as follows in this extended sketch.

The authors of [PT10] consider *sliced pure structures*, which are basically proof-nets for full linear logic that are only correct slice by slice, called *AC correctness*. A small modification in their definition is needed in our case: a $\top$-vertex should not have any distinguished (or main) conclusion but only auxiliary conclusions – this requires adapting the definition of a flat [PT10, Definition 2.1], corresponding to the one of a slice. This allows to have sliced pure structures invariant by $\top - \top$ commutations, as well as interpreting a $\emptyset$-rule by a $\top$-vertex, and keeping invariance by $\top - \emptyset$ commutations.

This minor modification of the syntax can be easily checked as not interfering with the proofs of the paper, mainly because its authors consider only in details the non-erasing cut-elimination step, and a $\top - cut$ commutation is erasing. Indeed, this slight modification implies some changes for the definition of cut-elimination in sliced pure structures [PT10, Definition 2.12]. When eliminating a *cut* with a $\top$-vertex above one of its premise, we apply the very same reduction as defined in [PT10]. Such a reduction may not be the image of one from the sequent calculus, in case the premise is the only one labeled $\top$. Still, it does not matter here as strong normalization for more steps implies strong normalization for the wished steps. There is also no need to consider a main conclusion of a $\top$-vertex when reducing a *cut* with an *ax*-vertex above one of its premises. Now, there might be two possible cut-elimination cases applicable on a given *cut*, if it has above its premises an *ax*-vertex and a $\top$-vertex, but this is quite harmless (there is still some basic verification to do, see [PT10, Remark 2.13]).

One can then follow all results of the paper, and see its proofs still hold after our modification. As a $\top - cut$ case is a logical erasing cut [PT10, Definitions 3.1 and 4.1], it mainly suffices to check the proof of [PT10, Theorem 4.2], and not many proofs consider erasing steps: they are [PT10, Proposition 3.4, Lemma 4.3, 4.4, Proposition 4.5] – in fact a $\top - cut$ cut-elimination case is sometimes not even explicitly handled as this case is trivial. We introduce at this point a key difference: one can define the equivalents of $\overset{\top}{\rightsquigarrow}$, $\overset{oa}{\rightsquigarrow}$ and $\overset{oe}{\rightsquigarrow}$ on sliced pure structures. The first is a generalization of a $\top - cut$ elimination case, with a $\top$-vertex that can erase any sub-structure. The second is that given two slices s and r that differ only by a sub-slice, sub-slice which is composed of a sole $\top$-vertex in s, then s can be removed from the multiset of slices; or more generally (and simply) that a slice can be erased. The last one is the usual transformation on $\mathsf{MELL}^{0,2}_{uf}$ proof-nets: in a slice, a $?_w$-vertex above a $?_c$ one can be simplified by removal of these two vertices. There is nothing corresponding to $\overset{om}{\rightsquigarrow}$ for flats as they already quotient by this relation. We consider these as new logical erasing steps, and can check that the previously mentioned results for the $\top - cut$ case also holds for these new cases – the main difference in the proofs is for a $\overset{oe}{\rightsquigarrow}$ transformation in [PT10, Lemma 4.4]: to postpone such a step with respect to a non-erasing step, in the case of a cut-elimination step between a *cut* below the erased $?_c$-vertex and the vertex above the latter, one has to apply a non-erasing $?_c - !$ key case, then the considered cut-elimination case, then some $?_w - !$ key cases. Therefore, the adaptation of [PT10, Theorem 4.2] is true for our slightly different sliced pure structures: a sliced pure structure satisfying AC (acyclicity by slice) which is weakly normalizing for non-erasing steps is strongly normalizing for all steps, including the new erasing ones we just added.

We now adapt [PT10, Section 5] by considering proofs of sequent calculus instead of proof-nets. This is quite easy, as the authors of [PT10] consider proof-nets as the images of proofs: given a proof-net β, the associated sliced pure structure $sl(\beta)$ is defined from a proof π desequentializing to β. We thus simply define the sliced pure structure $sl(\pi)$ associated to a proof π as done in [PT10, Section 5.3], with the obvious extension in case of the optional rules:

- a mix_2-rule is translated as the union of all couples of slices (like a $\otimes$-rule);
- a mix_0-rule is translated as the empty slice;
- a $\cup$-rule is translated as taking the union (as multisets) of the slices (like a &-rule);
- a $\emptyset$-rule is translated by a single slice consisting of a $\top$-vertex, which has no main conclusion (like a $\top$-rule).

We have as usual that the image of a proof is correct, *i.e.* AC. Also, that typed AC sliced pure structures are weakly normalizing for non-erasing steps is still true in this setting, and an adaptation by replacing proof-nets with proofs in the proof with reducibility candidates of [PT10, Theorem 5.11] should be enough to get that if π is a proof then $sl(\pi)$ is weakly normalizing for non-erasing steps. Another way to prove it would be to extend their notion of proof-nets with $\cup$- and $\emptyset$-rules, and check the proof of [PT10, Theorem 5.11] stays valid in this framework.

Afterwards starts the core of this proof, where considering proofs instead of proof-nets makes a difference. One has to prove lemmas equivalent to [PT10, Lemma 5.3, 5.4, 5.5] in the framework of sequent calculus, namely:

- If $\pi \xrightarrow{\beta} \pi'$ or $\pi \overset{\top}{\rightsquigarrow} \pi'$ or $\pi \overset{o}{\rightsquigarrow} \pi'$, then $sl(\pi) \xrightarrow{\beta^*} sl(\pi')$ (remember we put the steps corresponding to $\overset{\top}{\rightsquigarrow}$ and $\overset{o}{\rightsquigarrow}$ in sliced pure structures), with $sl(\pi) = sl(\pi')$ if and only if this cut-elimination step is a $\forall - \exists$ key step or a commutative step other than $\top - cut$, $! - cut$ and $\emptyset - cut$, or if the step is a $\overset{om}{\rightsquigarrow}$ step.
- If $\pi \overset{r\setminus\top}{\longmapsto\!\dashv} \pi'$ then $sl(\pi) = sl(\pi')$.
- If $\pi \overset{c}{\longmapsto\!\dashv} \pi'$ then $sl(\pi) = sl(\pi')$.

We then conclude by contradiction. Taking an infinite sequence of $(\xrightarrow{\overline{\beta}} \cup \overset{\top}{\rightsquigarrow} \cup \overset{o}{\rightsquigarrow}) \cdot (\overset{r\setminus\top}{\longmapsto\!\dashv} \cup \overset{c}{\longmapsto\!\dashv})^*$ steps in sequent calculus, its image in AC sliced pure structures cannot be an infinite sequence of $\xrightarrow{\beta}$ steps. Therefore, there exists an infinite suffix of this sequence where all $(\xrightarrow{\overline{\beta}} \cup \overset{\top}{\rightsquigarrow} \cup \overset{o}{\rightsquigarrow})$ steps are $\forall - \exists$ key steps or commutative steps other than $\top - cut$, $! - cut$ and $\emptyset - cut$, or $\overset{om}{\rightsquigarrow}$ steps. But only a finite number of such steps can be applied, including when interleaving them with $(\overset{r\setminus\top}{\longmapsto\!\dashv} \cup \overset{c}{\longmapsto\!\dashv})^*$ steps. Indeed, a $\forall - \exists$ key step decreases the number of $\forall$- and $\exists$-rules by slice, as does a $\overset{om}{\rightsquigarrow}$ step about mix_2- and mix_0-rules. And a commutative step which is not a $\top - cut$, $! - cut$, $\emptyset - cut$ nor a $cut - cut$ step decreases the number of rules above blocks of cut-rules in a slice, counting in a slice for each rule the number of cut-rules in blocks below it. Finally, a $\overset{r\setminus\top}{\longmapsto\!\dashv} \cup \overset{c}{\longmapsto\!\dashv}$ step preserves the number of slices, the numbers and kinds of rules by slices and the blocks of cut-rules per slice. We conclude that such a suffix cannot exist. □

2.3.6 Confluence of cut-elimination

We show here two results in $\mathsf{MALL}^{0,2}$, and assume implicitly in all this section that proofs and rules under consideration belong to this sub-system. The first is that rule commutations are included in equality up to cut-elimination, proved in Section 2.3.6.1. The second is that cut-elimination is Church-Rosser modulo rule commutation, whose demonstration is in Section 2.3.6.2; it also holds in the presence of the *mix*-Rétoré transformation (Section 2.3.6.3).

Proposition 2.46 ($\overset{rc}{\longmapsto\!\dashv} \subseteq =_\beta$)**.** *Given proofs π and τ in $\mathsf{MALL}^{0,2}$, if $\pi \overset{rc}{\longmapsto\!\dashv} \tau$ then $\pi \xleftarrow{\beta^*} \cdot \xrightarrow{\beta^*} \tau$. In particular, if $\pi \overset{r}{\longmapsto\!\dashv} \tau$ then $\pi =_\beta \tau$.*

Theorem 2.47. *In $\mathsf{MALL}^{0,2}$:*

- $\xrightarrow{\overline{\beta}}$ *is Church-Rosser modulo* $(\overset{r}{\longmapsto\!\dashv} \cup \overset{c}{\longmapsto\!\dashv})^*$.
- $\xrightarrow{\beta}$ *is Church-Rosser modulo* $\overset{r^*}{\longmapsto\!\dashv}$.

Theorem 2.48. *In $\mathsf{MALL}^{0,2}$:*

- $\xrightarrow{\overline{\beta}} \cup \overset{om}{\rightsquigarrow}$ *is Church-Rosser modulo* $(\overset{r}{\longmapsto\!\dashv} \cup \overset{c}{\longmapsto\!\dashv} \cup \overset{om}{\longmapsto\!\dashv})^*$.
- $\xrightarrow{\beta} \cup \overset{om}{\rightsquigarrow}$ *is Church-Rosser modulo* $(\overset{r}{\longmapsto\!\dashv} \cup \overset{om}{\longmapsto\!\dashv})^*$.

The generalization of Proposition 2.46 to the full system, with all optional rules, is a matter of checking (a lot of) cases – *c.f.* the 16 tables at the end of Chapter 1 with the 152 rule commutations. The first two results together lead to a characterization of normal forms with equality up to cut-elimination, with rule commutation being the “kernel” of cut-elimination in the following meaning: two proofs are β-equal if and only if their normal forms are $\overset{r^*}{\vdash\!\dashv}$-equal. Moreover, using the third result instead of the second one, one can add $\overset{\rho m}{\vdash\!\dashv}$.

Theorem 2.49 (Rule commutation is the kernel of cut-elimination)**.** *Set π and τ two proofs, and π' and τ' any of their respective normal forms by $\xrightarrow{\beta}$ (resp. by $\xrightarrow{\beta} \cup \overset{om}{\rightsquigarrow}$). The proofs π and τ are β-equal (resp. are equal up to $\xrightarrow{\beta}$ and $\overset{om}{\rightsquigarrow}$) if and only if π' and τ' are related by $\overset{r^*}{\vdash\!\dashv}$ (resp. are related by $(\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*$).*

Proof. For the direct way, by Theorem 2.47 (resp. Theorem 2.48) the normal forms, related by $=_{\beta o}$ with o being made of no Rétoré transformations (resp. with o being made of exactly om), are related by $\overset{r^*}{\vdash\!\dashv}$ (resp. by $(\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*$).

Conversely, if $\pi' \overset{r^*}{\vdash\!\dashv} \tau'$ (resp. $\pi'(\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*\tau'$) then by Proposition 2.46 one gets $\pi_1 \xrightarrow{\beta^*} \pi_1' =_{\beta o} \pi_2' \xleftarrow{\beta^*} \pi_2$. □

It is an important general result about sequent calculus, which we are convinced should hold for full linear logic. Once proved for β-equality, it can be extended to $\beta\eta$-equality; it also implies links between normal forms for cut-elimination.

Theorem 2.50. *Set o being either no Rétoré transformation, or being exactly mix-Rétoré (*i.e.* $\overset{o}{\rightsquigarrow} \in \{\emptyset; \overset{om}{\rightsquigarrow}\}$). Take π_1 and π_2 proofs, and set π_1' (resp. π_2') a result of expanding all axioms, eliminating all cuts and applying all $\overset{o}{\rightsquigarrow}$ in π_1 (resp. π_2). Then $\pi_1 =_{\beta\eta o} \pi_2$ if and only if $\pi_1'(\overset{r}{\vdash\!\dashv} \cup \overset{o}{\vdash\!\dashv})^*\pi_2'$.*

Proof. By Theorem 2.49, $\eta(\pi_1) =_{\beta o} \eta(\pi_2)$ if and only if $\pi_1'(\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*\pi_2'$. If $\pi_1 =_{\beta\eta o} \pi_2$, then $\eta(\pi_1) =_{\beta o} \eta(\pi_2)$ by Proposition 2.10. Conversely, if $\eta(\pi_1) =_{\beta o} \eta(\pi_2)$ then $\pi_1 \xrightarrow{\eta^*} \eta(\pi_1) =_{\beta o} \eta(\pi_2) \xleftarrow{\eta^*} \pi_2$. □

Corollary 2.51 (Confluence up to rule commutation)**.** *If π_1 and π_2 are cut-free proofs obtained by cut-elimination from the same proof π, then π_1 and π_2 are equal up to rule commutation.*

Proof. Consequence of Theorem 2.49, for $\pi_1 =_\beta \pi_2$. □

These lacking results in the literature of the sequent calculus of linear logic are quite meaningful. On one side, proofs are semantically seen up to cut-elimination and axiom-expansion, namely are considered up to $=_{\beta\eta}$ (and sometimes also up to $\overset{\rho m}{\vdash\!\dashv}$). On the other hand, plenty of results deal with proofs up to rule commutation, as the proof-net syntax that we will see later [HG16]. According to Theorem 2.50, looking at proofs up to $\overset{r}{\vdash\!\dashv}$ is indeed the right way for identifying proofs up to $=_{\beta\eta}$. This explanation is more convincing that the somewhat usual one that rule commutations are “bureaucracy” and thus should not matter, not only because this is a “non-choice” up to β-equality that is mandatory in the syntax, but also because it sums up exactly β-equality between normal forms.

Remark 2.52. From Theorem 2.47, one finds that $\overset{\rho m}{\vdash\!\dashv}$ is not included in $\beta\eta$-equality, so that adding it or not is truly a choice. Indeed, one can check it is not generated by rule commutation. If this theorem generalizes to the full linear logic, as we conjecture, then it would be the case for all Rétoré transformations. Similarly, commutations of the following kind would not in $=_{\beta\eta}$, meaning that having a $?_c$- or $?_c$-rule commuting with a !-rule is only true up to provability – *i.e.* it preserves being a proof, but not pure calculability given by the cut-elimination procedure.

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, ?\Gamma}}{\vdash !A, ?B, ?B, ?\Gamma}\,(!)}{\vdash !A, ?B, ?\Gamma}\,(?_c) \qquad \equiv \qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, ?B, ?B, ?\Gamma}}{\vdash !A, ?B, ?\Gamma}\,(?_c)}{\vdash !A, ?B, ?\Gamma}\,(!)$$

2.3.6.1 Cut-elimination contains rule commutation

Proving that rule commutation is included in equality up to cut-elimination is easy, albeit tedious.

Proof of Proposition 2.46. It suffices, for each commutation, to give a proof that can be reduced by cut-elimination to both sides of the commutation – using that both rule commutation and cut-elimination are contextual, so that one can ignore what happens below a rule commutation. Remark that this trivially holds for $\overset{c}{\vdash\!\dashv}$, as well as for other commutations of $\overset{rc}{\vdash\!\dashv}$ involving a *cut*-rule: two such proofs are related by $\xrightarrow{\overline{\beta}}$ or $\xleftarrow{\overline{\beta}}$. All cases are similar and follows the same idea, hence we give here only a few representative cases. A proof with every one of the 55 cases is given in Appendix A for completeness sake. In all cases here, we give a proof with a single *cut*-rule above two others proofs, such that the above proof reduces to the left one or to the right one according respectively to whether the *cut*-rule is first commuted with the rule on its left or on its right – once this first commutation is done, there will always be a single result of cut-elimination, that can be reached using only $\xrightarrow{\overline{\beta}}$ steps.

- *$C^{\&}_{\top}$ and $C^{\top}_{\&}$ commutations.*

$$\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \dfrac{}{\vdash B, \top, \Gamma}\,(\top)}{\vdash A \,\&\, B, \top, \Gamma}\,(\&)}{\vdash A \,\&\, B, \top, \Gamma}\,(cut)$$

$$\overset{\overline{\beta}}{\swarrow} \qquad\qquad\qquad \overset{\overline{\beta}}{\searrow}$$

$$\dfrac{}{\vdash A \,\&\, B, \top, \Gamma}\,(\top) \qquad\qquad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \dfrac{}{\vdash B, \top, \Gamma}\,(\top)}{\vdash A \,\&\, B, \top, \Gamma}\,(\&)$$

- *$C^{\oplus_1}_{\perp}$ and $C^{\perp}_{\oplus_1}$ commutations.*

$$\dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash A^{\perp}, A \oplus B}\,(\oplus_1) \qquad \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(cut)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(\oplus_1)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, \bot, \Gamma}\,(\bot)$$

- $C^{\otimes}_{\otimes}$ *commutations.*

$$\dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \overset{\phi}{\vdash A, D, \Delta}}{\vdash A, C \otimes D, \Gamma, \Delta}\,(\otimes) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax) \qquad \overset{\tau}{\vdash B, \Sigma}}{\vdash A^{\perp}, A \otimes B, \Sigma}\,(\otimes)}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(cut)$$

$$\xrightarrow{\beta} \quad \dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, D, \Delta} \qquad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, D, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \overset{\phi}{\vdash A, D, \Delta}}{\vdash A, C \otimes D, \Gamma, \Delta}\,(\otimes) \qquad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma}\,(\otimes)$$

Other $C^{\otimes}_{\otimes}$ commutations are similar.

- $C^{\parr}_{\&}$ *and* $C^{\&}_{\parr}$ *commutations.*

$$\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr) \qquad \dfrac{\overset{\phi}{\vdash A, B, D, \Gamma}}{\vdash A \parr B, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(\&) \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^{\perp}, B}\,(ax) \qquad \dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash A, B, B^{\perp} \otimes A^{\perp}}\,(\otimes)}{\vdash A \parr B, B^{\perp} \otimes A^{\perp}}\,(\parr)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(cut)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr) \qquad \dfrac{\overset{\phi}{\vdash A, B, D, \Gamma}}{\vdash A \parr B, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(\&)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma} \qquad \overset{\phi}{\vdash A, B, D, \Gamma}}{\vdash A, B, C \,\&\, D, \Gamma}\,(\&)}{\vdash A \parr B, C \,\&\, D, \Gamma}\,(\parr)$$

- $C^{mix_2}_{mix_2}$ *commutations*

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash \bot, \Delta}\,{\scriptstyle(\bot)}}{\vdash \bot, \Gamma, \Delta}\,{\scriptstyle(mix_2)} \qquad \dfrac{\dfrac{}{\vdash 1}\,{\scriptstyle(1)} \quad \overset{\tau}{\vdash \Sigma}}{\vdash 1, \Sigma}\,{\scriptstyle(mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle(cut)}$$

$$\overset{\bar\beta}{\swarrow} \qquad\qquad\qquad \overset{\bar\beta}{\searrow}$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Delta, \Sigma}\,{\scriptstyle(mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle(mix_2)} \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,{\scriptstyle(mix_2)} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle(mix_2)}$$

Other $C^{mix_2}_{mix_2}$ commutations are similar. □

Remark 2.53. In the proof of Proposition 2.46 for $\mathsf{MALL}^{0,2}$, each $\xrightarrow{\beta}$ step used for proving that the commutation of r and s is in β-equality are cases with ax-rules, rules of kinds r and s, or with the dual case, excepted for the $mix_2 - mix_2$ commutation that the author cannot manage to prove without using 1- and $\bot$-rules. This is not as surprising as it could be, for the mix_2- and mix_0-rules are strongly related to the multiplicative units: in presence of these two rules, $\bot$ and 1 are interprovable, and there is even a stronger property in presence of *mix*-Rétoré (*c.f.* Lemma 7.9).

2.3.6.2 Cut-elimination is Church-Rosser modulo rule commutation

Here is given the proof of Theorem 2.47. Our demonstration uses Theorem 2.7, on Page 57 in Section 2.1, and as such plenty of case studies and intermediate results.

We recall a $\xrightarrow{\bar\beta}$ step is a $\xrightarrow{\beta}$ step other than a $cut - cut$ commutation, and $\overset{c}{\vdash\!\dashv}$ is the $cut - cut$ commutation. We also recall a step of $\overset{r}{\vdash\!\dashv}$ is a *cut-free* rule commutation, *i.e.* is not a commutation involving a *cut*-rule nor having above the commuted rules a sub-proof with a *cut*-rule (but it may have a *cut*-rule in its external context); for instance in the $\top - \otimes$ commutation creating or deleting a sub-proof π, π is cut-free.[3] As $\overset{r}{\vdash\!\dashv}$ and $\overset{c}{\vdash\!\dashv}$ are symmetric, $\overset{r^*}{\vdash\!\dashv}$ and $\overset{c^*}{\vdash\!\dashv}$ are equivalence relations.

We will instantiate Theorem 2.7 with $\vdash\!\dashv\, = (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})$, $\rightarrow\, = \xrightarrow{\bar\beta}$ and $\rightsquigarrow\, = (\overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv})$ ($\overset{r\setminus\top}{\rightsquigarrow}$ being defined in Section 2.3.3). We already have that $\xrightarrow{\bar\beta} \cdot \rightsquigarrow^*$ is strongly normalizing (Proposition 2.41), so we need the other hypotheses of the lemma, namely those corresponding to local confluence and local coherence.

In this section we denote graphically some proofs with the following convention. When writing proofs as in

$$\begin{array}{c} \dfrac{\dfrac{\rho}{\quad}\,{\scriptstyle(r_2)}}{\quad}\,{\scriptstyle(r_1)} \\ \vdots \end{array} \quad \overset{r}{\vdash\!\dashv} \quad \begin{array}{c} \dfrac{\dfrac{\rho}{\quad}\,{\scriptstyle(r_1)}}{\quad}\,{\scriptstyle(r_2)} \\ \vdots \end{array}$$

we abuse notations in the cases where r_1 or r_2 is a & or $\top$-rule. The meaning is that, if say r_2 is a &-rule, then r_1 is duplicated in the proof on the left, and even possibly a whole sub-proof if r_1 is a $\otimes$-rule for instance. Similarly, if r_1 is a $\top$-rule, then this schema means that on the left hand-side r_2 and ρ are not here, and r_2 is created by the $\top$-commutation (and ρ is nothing). About ρ, we

[3] Actually, we only need it for the $\top - \otimes$, $\& - \otimes$, $\top - mix_2$ and $\& - mix_2$ commutations, but ask it for all commutations to homogenize and simplify some proofs.

mean there is a same sub-proof(s) above these two rules in both cases, *i.e.* above all premises of r_2 on the left there is a sub-proof τ_i which is also the sub-proof on the corresponding premise of r_1 on the right. Furthermore, in case one of the rule, say r_1, is a $\otimes$- or mix_2-rule, we also mean the sub-proof above the premise of r_1 which does not contain r_2 in the proof on the left, is also the sub-proof above the same premise in the proof on the right.

We start by some preliminary results, identifying sufficient conditions to apply a rule commutation.

Definition 2.54. A **contextual rule** r is the data of a kind of rule which is not an ax- nor a cut- nor a 1 nor a mix_0-rule, a main formula associated to this rule – if any, *i.e.* a mix_2-rule does not come with a main formula – and, if this kind is a $\otimes$- or mix_2-rule, then r comes along a choice of premise (left or right) as well as a proof. Given a rule s in a proof – s being made of a kind, premise sequent(s) and a conclusion sequent – and a contextual rule r, s is **associated** to r if it has the same kind, its main formula is the main formula of r and, in case it is a $\otimes$- or mix_2-rule and r has chosen its left (resp. right) premise the sub-proof above the right (resp. left) premise of s is the proof from r.

A **contextual block** $\mathfrak{B}$ is an (ordered) set of successive contextual rules, as they could appear in a proof. More explicitly, it is the data of a tree-like structure, with contextual rules as vertices, and such that each of these contextual rules has at most one child, except for a & contextual rule that has at most two distinguishable children (a left one and a right one), and a $\top$ one that has no child. Given a set of successive rules in a proof π, it is **associated** to $\mathfrak{B}$ if when looking at this set of rules seen as a sub-tree of π it has the same tree-like structure as $\mathfrak{B}$, and each of its rules is associated to the corresponding contextual rule of $\mathfrak{B}$. Moreover, for a $\otimes$- or mix_2-rule r of π, whose corresponding contextual rule in $\mathfrak{B}$ keeps its left (resp. right) premise and comes along a proof ϕ, we ask that the sub-proof above the right (resp. left) premise of r is ϕ.

As an example of contextual rule r, take as kind $\otimes$, as main formula $A \otimes B$, choose the left premise with the proof $\dfrac{}{\vdash A^\perp, A}\,(ax)$. A rule associated to r (in a proof) is for instance $\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \vdash B, \Gamma}{\vdash A \otimes B, A^\perp, \Gamma}\,(\otimes)$ or $\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \vdash B^\perp, B}{\vdash A \otimes B, A^\perp, B^\perp}\,(\otimes)$.

About contextual blocks, consider $\mathfrak{B}_{ex}$:

$$\begin{tikzcd}
\otimes,\ A^\perp \otimes B,\ \text{right},\ \dfrac{}{\vdash A^\perp, A}\,(ax) \arrow[dr] & & \oplus_1,\ (A^\perp \otimes B) \oplus A \arrow[dl] \\
& \&,\ (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A) &
\end{tikzcd}$$

We look at the following proof π_{ex}.

$$
\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,{\scriptstyle(ax)} \quad \dfrac{}{\vdash B^\perp, B}\,{\scriptstyle(ax)}}{\vdash A^\perp \otimes B, A, B^\perp}\,{\scriptstyle(\otimes)} \quad \dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,{\scriptstyle(ax)} \quad \dfrac{}{\vdash B^\perp, B}\,{\scriptstyle(ax)}}{\vdash A^\perp \otimes B, A, B^\perp}\,{\scriptstyle(\otimes)}}{\vdash (A^\perp \otimes B) \oplus A, A, B^\perp}\,{\scriptstyle(\oplus_1)}}{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp}\,{\scriptstyle(\&)} \quad \dfrac{\dfrac{}{\vdash B^\perp, B}\,{\scriptstyle(ax)}}{\vdash A^\perp \oplus B, B^\perp}\,{\scriptstyle(\oplus_2)}}{\dfrac{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp, B^\perp, A^\perp \oplus B}{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp \parr B^\perp, A^\perp \oplus B}\,{\scriptstyle(\parr)}}\,{\scriptstyle(mix_2)}
$$

The contextual block $\mathfrak{B}_{ex}$ is associated to the following rules of π_{ex}, not representing the ax-rule on A above the left premise of the $\otimes$-rule:

$$
\dfrac{\dfrac{\vdash B^\perp, B}{\vdash A^\perp \otimes B, A, B^\perp}\,{\scriptstyle(\otimes)} \quad \dfrac{\vdash A^\perp \otimes B, A, B^\perp}{\vdash (A^\perp \otimes B) \oplus A, A, B^\perp}\,{\scriptstyle(\oplus_1)}}{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp}\,{\scriptstyle(\&)}
$$

On the contrary, the following rules are not associated to any contextual block, for both premises of a $\otimes$-rule are kept:

$$
\dfrac{\dfrac{\vdash A^\perp \otimes B, A, B^\perp \quad \vdash (A^\perp \otimes B) \oplus A, A, B^\perp}{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp}\,{\scriptstyle(\&)} \quad \dfrac{\vdash B^\perp, B}{\vdash A^\perp \oplus B, B^\perp}\,{\scriptstyle(\oplus_2)}}{\vdash (A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A, B^\perp, B^\perp, A^\perp \oplus B}\,{\scriptstyle(mix_2)}
$$

Remark that given a rule in a proof, it always has one associated contextual rule (two in the case of a $\otimes$- or mix_2-rule), so it is always possible to see a rule as a contextual rule. By abuse of notations, when a rule r in a proof is associated to some contextual rule, we will often denote this contextual rule also by r.

Definition 2.55. Given a contextual rule r and a sequent $\vdash \Gamma$, the **domain** of r for $\vdash \Gamma$ is a sub-set of Γ made of the formulas (occurrences) of Γ having for sub-formulas the main formula of r (if any) as well as, if r is a $\otimes$ or mix_2, the formulas of the sequent on which the proof associated to r is.

For a contextual block $\mathfrak{B}$ and a sequent $\vdash \Gamma$, the **domain** of $\mathfrak{B}$ for $\vdash \Gamma$ is the union of the domains of each of its contextual rules for $\vdash \Gamma$.

Given r and s two contextual rules and a sequent $\vdash \Gamma$, r and s are **independent** for $\vdash \Gamma$ if the domain of r for $\vdash \Gamma$ is disjoint from the domain of s for $\vdash \Gamma$ (as sub-sets of Γ).

A rule r and a contextual block $\mathfrak{B}$ are **independent** for $\vdash \Gamma$ if the domain of r for $\vdash \Gamma$ is disjoint from the domain of $\mathfrak{B}$ for $\vdash \Gamma$ (as sub-sets of Γ). Remark this corresponds to the independence for $\vdash \Gamma$ of r with every contextual rule of $\mathfrak{B}$.

For instance, on the conclusion sequent of π_{ex}, the domain of the (contextual rule associated to) $\oplus_1$-rule is $(A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A)$, the domains of the leftmost $\otimes$-rule are $(A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), A$ or $(A^\perp \otimes B) \,\&\, ((A^\perp \otimes B) \oplus A), B^\perp \parr B^\perp$ (according to which premise is kept), the former also being the domain of $\mathfrak{B}_{ex}$. The $\oplus_2$-rule is independent with the contextual block $\mathfrak{B}_{ex}$, as well as with the mix_2-rule, whereas the rightmost $\otimes$-rule (for both choice of premises) is not independent with $\mathfrak{B}_{ex}$.

Fact 2.56. *Consider a proof containing a cut-rule c which has for conclusion sequent $\vdash \Gamma$ and has on its premises the sub-proofs π_1 and π_2. Take a contextual rule r_1 (resp. r_2) associated to some rule s_1 in π_1 (resp. s_2 in π_2). Then r_1 and r_2 are independent for $\vdash \Gamma$.*

Proof. Because the *cut*-rule c separates Γ into two sub-sequents, and the domain of r_1 for $\vdash \Gamma$ is included in the first while the one of r_2 is included in the second. ☐

Lemma 2.57. *Consider rules in a proof π of conclusion sequent $\vdash \Gamma$, seen as contextual rules.*

1. *A rule commutation does not modify the domain of any non-erased contextual rule for $\vdash \Gamma$.*
2. *A commutative cut-elimination case does not modify the domain of any non-erased contextual rule for $\vdash \Gamma$.*
3. *A key cut-elimination case does not modify the domain of any non-erased contextual rule for $\vdash \Gamma$.*

Proof. Each item to check consists in a tedious case analysis. We give here a few representative ones. Remark that, for any transformation of proofs, a domain of a non-erased rule which is not involved in the transformation is unchanged (as our transformations are local, except when erasing rules).

Item 1. Consider the following $\otimes - \&$ commutation:

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A_1,\Gamma} \quad \overset{\pi_2}{\vdash A_2,B_1,\Delta}}{\vdash A_1\otimes A_2, B_1,\Gamma,\Delta}\,(\otimes) \quad \dfrac{\overset{\pi_1}{\vdash A_1,\Gamma}\quad \overset{\pi_3}{\vdash A_2,B_2,\Delta}}{\vdash A_1\otimes A_2,B_2,\Gamma,\Delta}\,(\otimes)}{\vdash A_1\otimes A_2, B_1 \mathbin{\&} B_2,\Gamma,\Delta}\,(\&) \quad \overset{r}{\vdash\!\dashv} \quad \dfrac{\overset{\pi_1}{\vdash A_1,\Gamma} \quad \dfrac{\overset{\pi_2}{\vdash A_2,B_1,\Delta}\quad\overset{\pi_3}{\vdash A_2,B_2,\Delta}}{\vdash A_2,B_1\mathbin{\&}B_2,\Delta}\,(\&)}{\vdash A_1\otimes A_2,B_1\mathbin{\&}B_2,\Gamma,\Delta}\,(\otimes)$$

For the &-rule, its domain is (the formulas of the conclusion sequent whose sub-formulas are) $B_1 \mathbin{\&} B_2$ in both cases. For the $\otimes$-rule, its domain when keeping the right premise is $A_1 \otimes A_2, \Gamma$ in both occurrences on the left, as well as on the right. If it were to keep its left premise, then it is erased as a contextual rule (the associated proof on its right premise has changed).

Item 2. We take interest in a $\otimes - cut - 1$ commutative case.

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A,B,\Gamma}\quad\overset{\pi_2}{\vdash C,\Delta}}{\vdash A, B\otimes C,\Gamma,\Delta}\,(\otimes)\quad \overset{\pi_3}{\vdash A^\perp,\Sigma}}{\vdash B\otimes C,\Gamma,\Delta,\Sigma}\,(cut) \quad \overset{\beta}{\longrightarrow} \quad \dfrac{\dfrac{\overset{\pi_1}{\vdash A,B,\Gamma}\quad\overset{\pi_3}{\vdash A^\perp,\Sigma}}{\vdash B,\Gamma,\Sigma}\,(cut)\quad\overset{\pi_2}{\vdash C,\Delta}}{\vdash B\otimes C,\Gamma,\Delta,\Sigma}\,(\otimes)$$

The domain of the $\otimes$-rule, when keeping its left premise, stays (the formulas of the conclusion sequent whose sub-formulas are) $B \otimes C, \Delta$. While keeping its right premise, then it is erased as a contextual rule: its left sub-proof is modified – which is important, as otherwise its domain would become $B \otimes C, \Gamma, \Sigma$ from $B \otimes C, \Gamma$.

Item 3. We study here a $\& - \oplus_1$ key case:

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A_1,\Gamma}\quad\overset{\pi_2}{\vdash A_2,\Gamma}}{\vdash A_1\mathbin{\&}A_2,\Gamma}\,(\&)\quad\dfrac{\overset{\pi_3}{\vdash A_2^\perp,\Delta}}{\vdash A_2^\perp\oplus A_1^\perp,\Delta}\,(\oplus_1)}{\vdash \Gamma,\Delta}\,(cut) \quad \overset{\beta}{\longrightarrow} \quad \dfrac{\overset{\pi_2}{\vdash A_2,\Gamma}\quad\overset{\pi_3}{\vdash A_2^\perp,\Delta}}{\vdash\Gamma,\Delta}\,(cut)$$

The initial remark is enough here: the rules involved in the transformation are erased, so there is nothing to check. ☐

Lemma 2.58. *Take r and s two contextual rules and $\vdash \Gamma$ a sequent. Assume there is a proof of the shape*

$$\dfrac{\dfrac{\pi}{\quad}\; r}{\vdash \Gamma}\; s$$

(recall our convention for this notation at the beginning of this section), where:

- *the r and s appearing in this proof are abuse of notations for rules associated to r and s;*
- *the proof(s) π is (are) cut-free – including the possible proofs associated to r or s in case of $\otimes$- or mix_2-rules.*

The contextual rules r and s are independent for $\vdash \Gamma$ if and only if they commute in this proof:

$$\dfrac{\dfrac{\pi}{\quad}\; r}{\vdash \Gamma}\; s \quad \overset{r}{\vdash\!\dashv} \quad \dfrac{\dfrac{\pi}{\quad}\; s}{\vdash \Gamma}\; r$$

Proof. One has to check it for every type of rule r and s can be, among $\otimes$, $⅋$, $\perp$, $\&$, $\oplus_1$, $\oplus_2$, $\top$ and mix_2-rules. A representative case is r is a $\otimes$-rule and s a $\&$-rule, other cases being left at the charge of the reader. By symmetry, say that r keeps its right premise, with a proof π_1 on its left premise. If the rules are independent, then our proof $\dfrac{\dfrac{\pi}{\quad}\; r}{\vdash \Gamma}\; s$ must be:

$$\dfrac{\dfrac{\overset{\pi_1}{\vdash A_1, \Sigma} \qquad \overset{\pi_2}{\vdash A_2, B_1, \Delta}}{\vdash A_1 \otimes A_2, B_1, \Sigma, \Delta}\,(\otimes) \qquad \dfrac{\overset{\pi_1}{\vdash A_1, \Sigma} \qquad \overset{\pi_3}{\vdash A_2, B_2, \Delta}}{\vdash A_1 \otimes A_2, B_2, \Sigma, \Delta}\,(\otimes)}{\vdash A_1 \otimes A_2, B_1 \,\&\, B_2, \Sigma, \Delta}\,(\&)$$

We recognize a $\otimes - \&$ rule commutation, from which one gets $\dfrac{\dfrac{\pi}{\quad}\; s}{\vdash \Gamma}\; r$:

$$\dfrac{\overset{\pi_1}{\vdash A_1, \Sigma} \qquad \dfrac{\overset{\pi_2}{\vdash A_2, B_1, \Delta} \qquad \overset{\pi_3}{\vdash A_2, B_2, \Delta}}{\vdash A_2, B_1 \,\&\, B_2, \Delta}\,(\&)}{\vdash A_1 \otimes A_2, B_2 \,\&\, B_2, \Sigma, \Delta}\,(\otimes)$$

Another representative case is s a $\top$-rule and r a mix_2-rule. By symmetry again, say that r keeps its left premise, with a proof π_1 on its right premise, of conclusion sequent $\vdash \Delta$. If the rules are independent, then our proof $\dfrac{\dfrac{\pi}{\quad}\; r}{\vdash \Gamma}\; s$ must be:

$$\dfrac{}{\vdash \top, \Sigma, \Delta}\,(\top)$$

We recognize a $\top - mix_2$ rule commutation, from which one gets $\dfrac{\dfrac{\pi}{\quad}\; s}{\vdash \Gamma}\; r$:

$$\dfrac{\dfrac{}{\vdash \top, \Sigma}\,(\top) \qquad \overset{\pi_1}{\vdash \Delta}}{\vdash \top, \Sigma, \Delta}\,(mix_2)$$

For the reciprocal, it suffices to check for every instance of a rule commutation that the contextual rules are independent for the conclusion sequent, as it is the case here for a $\otimes - \&$ commutation and for a $\top - mix_2$ commutation. $\square$

Remark 2.59. It is important that in Lemma 2.58 the sequent considered is the conclusion sequent of s. Otherwise, commutation does not yield independence, as in the following where r and s are not independent for $\vdash (A \parr B) \parr (C \parr D), \Gamma$ even though they commute.

$$\dfrac{\dfrac{\dfrac{\vdash A, B, C, D, \Gamma}{\vdash A \parr B, C, D, \Gamma}\,(\parr\ r)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr\ s)}{\vdash (A \parr B) \parr (C \parr D), \Gamma}\,(\parr)$$

Lemma 2.60. *Consider a contextual rule r and a contextual block $\mathfrak{B}$ that are independent for the sequent $\vdash \Gamma$. Assume there are proofs π and ϕ of respective shapes*

$$\pi = \quad \dfrac{\begin{matrix}\tau \\ \mathfrak{B}\end{matrix}}{\vdash \Gamma}\, r \qquad\qquad \phi = \quad \begin{matrix}\dfrac{\tau}{\mathfrak{B}}\, r \\ \vdash \Gamma\end{matrix}$$

which means:

- *the r and $\mathfrak{B}$ in the proofs are rules associated to the contextual rules with the same names;*
- *above the sequent of each leaf rule of $\mathfrak{B}$ (where for a $\otimes$- or mix_2-rule we look only at its kept sequent), there is a rule (associated to) r – remark there can be $\top$-rules in $\mathfrak{B}$, with no sequent above them;*
- *in case r is a $\otimes$- or mix_2-rule, in π the rules of $\mathfrak{B}$ are above the premise kept by (the contextual) r;*
- *the proof(s) τ is (are) cut-free – including the proof associated to a contextual rule in case it is a $\otimes$ or mix_2.*

Then r commutes successively with all rules of $\mathfrak{B}$: $\pi \overset{r^*}{\vdash\!\dashv} \phi$.

Proof. By induction on the number of (contextual) rules of $\mathfrak{B}$. If it has no rule, then there is nothing to prove: $\pi = \phi$. Otherwise, set s the root rule of $\mathfrak{B}$. Either s is above r in π, or r is a $\top$-rule. In both cases, by Lemma 2.58, s and r commute in π, yielding a proof π'. In π', r and the contextual block(s) made of the contextual rules of $\mathfrak{B}$ excepted s (there may be two blocks if s is a $\&$-rule), are independent for $\vdash \Gamma$ by Item 1 of Lemma 2.57 (we keep the same choice for premises of $\otimes$-rules), and thus for the conclusion sequent of r in π', because the latter is a sub-sequent of the former (Fact 1.7). We conclude using the induction hypothesis on π' and ϕ – more exactly on their sub-proof(s) where s is removed. $\square$

Lemma 2.61. *Let π, π_1 and π_2 be proofs such that $\pi_1 \xleftarrow{\overline{\beta}} \pi \xrightarrow{\overline{\beta}} \pi_2$. Then there exist π_1' and π_2' such that $\pi_1 \xrightarrow{\overline{\beta}^*} \pi_1' \overset{r^=}{\vdash\!\dashv} \pi_2' \xleftarrow{\overline{\beta}^*} \pi_2$. Diagrammatically:*

$$\begin{tikzcd}
 & \pi \arrow[dl, "\overline{\beta}"'] \arrow[dr, "\overline{\beta}"] & \\
\pi_1 \arrow[dr, dashed, "\overline{\beta}^*"'] & & \pi_2 \arrow[dl, dashed, "\overline{\beta}^*"] \\
 & \pi'_1 \overset{r^=}{\vdash\!\dashv} \pi'_2 &
\end{tikzcd}$$

More precisely, we need a step of $\overset{r}{\vdash\!\dashv}$ *exactly when both* $\xrightarrow{\overline{\beta}}$ *steps are different commutative cases on the same* cut*-rule.*

Proof. If the $\pi \xrightarrow{\overline{\beta}} \pi_1$ and $\pi \xrightarrow{\overline{\beta}} \pi_2$ steps involve only distinct rules then, taking into account that rules of one may be duplicated or erased by the other step, they commute and we have a proof π' such that $\pi_1 \xrightarrow{\overline{\beta}^*} \pi' \xleftarrow{\overline{\beta}^*} \pi_2$, by applying one reduction after the other. With more details:

- $\pi_1 \xrightarrow{\overline{\beta}} \cdot \xleftarrow{\overline{\beta}} \pi_2$ if neither step duplicates nor erases the other
- $\pi_1 \xleftarrow{\overline{\beta}} \pi_2$ if $\pi \xrightarrow{\overline{\beta}} \pi_1$ erases a sub-proof containing the rules involved in $\pi \xrightarrow{\overline{\beta}} \pi_2$ (and symmetrically if we swap indices 1 and 2)
- $\pi_1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}} \cdot \xleftarrow{\overline{\beta}} \pi_2$ if $\pi \xrightarrow{\overline{\beta}} \pi_1$ duplicates a sub-proof containing the rules involved in $\pi \xrightarrow{\overline{\beta}} \pi_2$ (and symmetrically if we swap indices 1 and 2)

There is no more case, as to duplicate or erase rules, a $\xrightarrow{\overline{\beta}}$ step must be below these rules in a proof. In particular, both steps cannot duplicate or erase the rules of the other step, because the rules involved are distinct.

From now on, we assume both steps involve (at least) one common rule. If both reductions share all of their rules, then the two reductions are the same, so $\pi_1 = \pi_2$ and we are done (recall Remark 1.11 for our convention on the $\parr - \otimes$ key case). Hence, we will assume them not to share all of their rules. We distinguish cases according to the kinds of the $\xrightarrow{\overline{\beta}}$ steps.

If one step is a key case other than an ax one. Remark that on the three rules of a non-ax key case, no other $\xrightarrow{\overline{\beta}}$ step can be applied (only a $cut - cut$ commutation could have been applied, but this case does not belong to $\xrightarrow{\overline{\beta}}$). Whenceforth, this case cannot happen as it would lead to the two reductions sharing all of their rules.

If both steps are ax key cases. As the two reductions share one rule, but not all rules, the shared rule must be the cut-rule, with as premises two ax-rules. We can check that this critical pair leads to the same resulting proof from both reductions: $\pi_1 = \pi_2$.

If one step is an ax key case and the other a commutative case. By symmetry, assume $\pi \xrightarrow{\overline{\beta}} \pi_2$ is the ax key case. For the two reductions share a rule, and the ax-rule cannot participate in a commutative step, the shared rule must be the cut-rule. We can still do this ax key step after the commutation (maybe twice in case of duplication, or zero time in case of erasure), recovering π_2. Thus:

- $\pi_1 \xrightarrow{\overline{\beta}} \pi_2$ (ax-key case and not a $\& - cut$ nor $\top - cut$ commutative case)
- $\pi_1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}} \pi_2$ (ax-key case and $\& - cut$ commutative case)
- $\pi_1 = \pi_2$ (ax-key case and $\top - cut$ commutative case)

If both steps are commutative cases. Here again, as the two reductions share a rule, it must be the *cut*-rule, because there is at most one *cut*-rule directly below a given rule. As the reductions do not share both of their rules, in $\pi \xrightarrow{\overline{\beta}} \pi_1$ we sent a rule r_1 from a branch of the *cut* below it, and in $\pi \xrightarrow{\overline{\beta}} \pi_2$ we do similarly on a rule r_2 in the other branch. This case, more complex than the previous ones, is depicted schematically on Figure 2.2. We can in π_1 commute the *cut*-rule and r_2 – maybe twice in case of a duplication, or zero in case of an erasure – obtaining π_1'; and similarly in π_2 commute the *cut*-rule and r_1, yielding π_2'. The two resulting proofs differ exactly by a commutation of r_1 and r_2 – even if both are $\top$-rules, they differ by a $\top - \top$ commutation.

Indeed, r_1 and r_2 are independent for the conclusion sequent of the *cut*-rule we consider (Fact 2.56). They are still independent for this sequent in π_1 (resp. π_2), and then in π_1' (resp. π_2') – it is then the conclusion sequent of r_1 (resp. r_2) – by Item 2 of Lemma 2.57. For we apply rule commutation only on cut-free sub-proofs, we first eliminate all *cut*-rules above these two rules, in the same way in both sub-proofs (and in case of duplication, in the same way in all duplicates of the sub-proofs). This can be done thanks to weak normalization of $\xrightarrow{\overline{\beta}}$ (Corollary 2.43). In the resulting sub-proofs, r_1 and r_2 are still independent (or erased) as their domains do not depend on what happen above them, and cut-elimination is a local procedure (Items 2 and 3 of Lemma 2.57). So, they commute (Lemma 2.58).

Thus, $\pi_1 \xrightarrow{\overline{\beta}^*} \cdot \overset{r}{\vdash\!\dashv} \cdot \xleftarrow{\overline{\beta}^*} \pi_2$ if both steps are commutative cases. ☐

Lemma 2.62. *Let π_1, π_2 and π_3 be proofs such that $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. Then, there exist π_1' and π_2' such that $\pi_1 \xrightarrow{\overline{\beta}^+} \pi_1' \overset{r^*}{\vdash\!\dashv} \pi_2' \xleftarrow{\overline{\beta}^*} \pi_3$. Diagrammatically:*

$$\begin{tikzcd}
\pi_1 \arrow[r, phantom, "\overset{r}{\vdash\!\dashv}"] \arrow[dd, dashed, "\overline{\beta}^+"] & \pi_2 \arrow[dr, "\overline{\beta}"] & \\
& & \pi_3 \arrow[dl, dashed, "\overline{\beta}^*"] \\
\pi_1' \arrow[r, phantom, "\overset{r^*}{\vdash\!\dashv}"] & \pi_2' &
\end{tikzcd}$$

Moreover, this $\overset{r^*}{\vdash\!\dashv}$ step consists in one rule commuting with a succession of other rules; in particular it is a $\overset{r\setminus\top}{\rightsquigarrow}{}^*$ step (in one direction or the other).

Proof. An easily handled case is when the $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ and $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ steps involve only distinct rules; assume for now this is the case. A first general sub-case is when rules of one step are neither erased nor duplicated by the other. Then these steps commute and $\pi_1 \xrightarrow{\overline{\beta}} \cdot \overset{r}{\vdash\!\dashv} \pi_3$, using the same steps in the other order (because these are local transformations).

Now, consider the case where the two steps still involve distinct rules, but the $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ step duplicates a sub-proof containing the rules of $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ (which may happen if $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ is a $\& - cut$ commutative case). We apply the corresponding $\xrightarrow{\overline{\beta}}$ step first in π_1, yielding $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1$, and then the $\overset{r}{\vdash\!\dashv}$ step twice, once for each occurrence, to recover π_3: we get $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \overset{r}{\vdash\!\dashv} \cdot \overset{r}{\vdash\!\dashv} \pi_3$.

Another general case is when the rules involved in the two steps are distinct, but the $\xrightarrow{\overline{\beta}}$ step eliminates a sub-proof containing the rules of the $\overset{r}{\vdash\!\dashv}$ step (this can arise when using a $\& - \oplus_i$ key

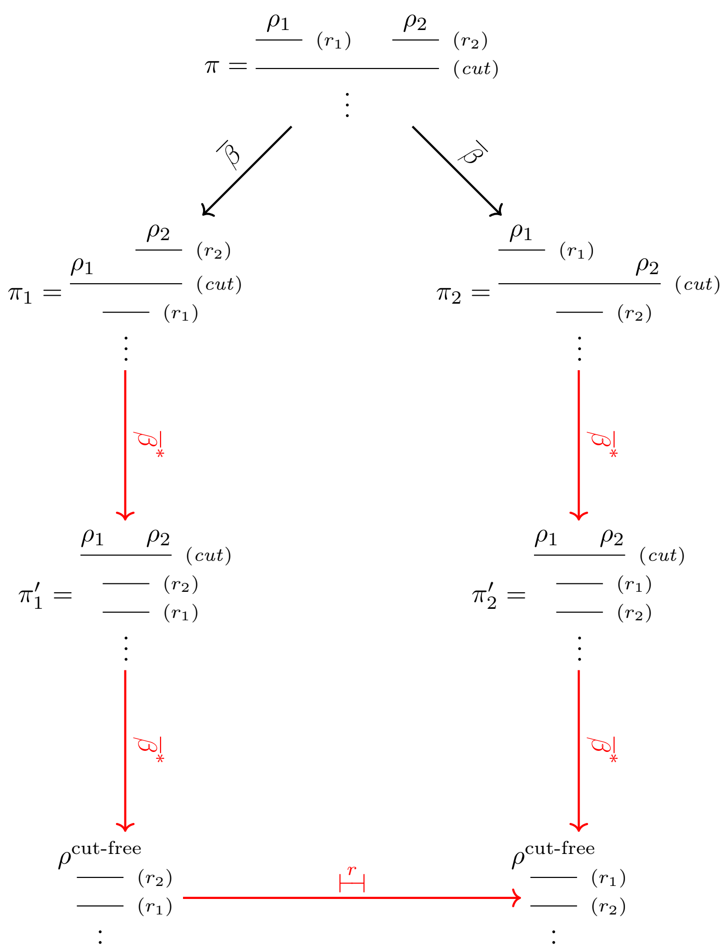


Figure 2.2: Schematic representation of the last case of the proof of Lemma 2.61

case or a $\top - cut$ commutative case). In this case, doing in π_1 the $\xrightarrow{\overline{\beta}}$ step directly yields π_3: $\pi_1 \xrightarrow{\overline{\beta}} \pi_3$.

Remark that, if these steps use distinct rules, the $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ step cannot duplicate nor erase the rules involved in $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$. Indeed, this may happen if the $\overset{r}{\vdash\!\dashv}$ step is a $\& - \otimes$, $\top - \otimes$, $\& - mix_2$ or $\top - mix_2$ commutative case, but we assumed that a sub-proof corresponding to a rule commutation is cut-free, and a $\xrightarrow{\overline{\beta}}$ step involves a *cut*-rule by definition.

From now on, we suppose both steps involve at least one common rule, which cannot be a *cut* one, for there is no commutation involving a *cut*-rule in $\overset{r}{\vdash\!\dashv}$. In fact, there is exactly one shared rule. Indeed, the rules commuting in $\overset{r}{\vdash\!\dashv}$ are successive rules, so both rules cannot be above a *cut*-rule. We distinguish cases according to the kind of $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$.

If $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ *is an ax key case.* As an *ax*-rule never commutes, the two steps share no rule, contradiction.

If $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ *is a* $\parr - \otimes$ *key case.* In this case, π_2 and π_3 are the following proofs:

$$\dfrac{\dfrac{\dfrac{\overset{\rho_1}{\vdash A,\Gamma} \qquad \overset{\rho_2}{\vdash B,\Delta}}{\vdash A\otimes B,\Gamma,\Delta}\,(\otimes) \qquad \dfrac{\overset{\rho_3}{\vdash B^\perp,A^\perp,\Sigma}}{\vdash B^\perp \parr A^\perp,\Sigma}\,(\parr)}{\vdash \Gamma,\Delta,\Sigma}\,(cut)}{\vdash \rho_4} \qquad \dfrac{\dfrac{\overset{\rho_1}{\vdash A,\Gamma} \qquad \dfrac{\overset{\rho_2}{\vdash B,\Delta} \qquad \overset{\rho_3}{\vdash B^\perp,A^\perp,\Sigma}}{\vdash A^\perp,\Delta,\Sigma}\,(cut)}{\vdash \Gamma,\Delta,\Sigma}\,(cut)}{\vdash \rho_4}$$

(up to symmetry of the *cut*-rule, the case where π_2 has a $\parr$-rule on the left and a $\otimes$ one on the right being symmetric and solved similarly).

By our assumption, $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ was a step pushing down the $\otimes$ or $\parr$-rule, and up some non *cut*-rule r. We can in π_1 commute the *cut*-rule up and r down (as r cannot be the rule of the main connective of the formula on which we cut, nor an *ax*-rule). This yields a proof π_1^1 such that $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1$ with this commutative step, with π_1^1 being π_2 except r is below the *cut*-rule and not above the $\otimes$ or $\parr$-rule (by abuse, for if r is a $\top$-rule then the *cut*-rule is not here anymore). Thus, $\pi_1^1 \xrightarrow{\overline{\beta}} \pi_1^2$ using the same step as in $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$; unless if r is a $\top$-rule, in which case there is nothing to do and we set $\pi_1^1 = \pi_1^2$; or if r is a $\&$-rule, where we have to apply this step in both occurrences, obtaining $\pi_1^1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}} \pi_1^2$. In any case, $\pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2$. Observe that π_1^2 is like π_3, except that r is above some *cut*-rule(s) in π_3 and below in π_1^2. But, using $\xrightarrow{\overline{\beta}}$ in π_3, r can commute down one or two of the *cut*-rules created by the key case, yielding π_1^2 (including if r is a $\&$ or $\top$-rule). Therefore, $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2 \xleftarrow{\overline{\beta}^+} \pi_3$, concluding this case.

If $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ *is a* $\& - \oplus_i$ *key case.* This case is similar to the previous one, in simpler as we create one new *cut*-rule and not two. We have π_2 and π_3 the following proofs:

$$\dfrac{\dfrac{\dfrac{\overset{\rho_2}{\vdash A_2,\Gamma} \qquad \overset{\rho_1}{\vdash A_1,\Gamma}}{\vdash A_2 \,\&\, A_1,\Gamma}\,(\&) \qquad \dfrac{\overset{\rho_3}{\vdash A_i^\perp,\Delta}}{\vdash A_1^\perp \oplus A_2^\perp,\Delta}\,(\oplus_i)}{\vdash \Gamma,\Delta}\,(cut)}{\vdash \rho_4} \qquad \dfrac{\dfrac{\overset{\rho_i}{\vdash A_i,\Gamma} \qquad \overset{\rho_3}{\vdash A_i^\perp,\Delta}}{\vdash \Gamma,\Delta}\,(cut)}{\vdash \rho_4}$$

(as before, the symmetric case with $\&$ on the right premise of the *cut* is handled similarly).

The $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ step was a commutation pushing down the $\&$ or $\oplus_i$-rule, and another non-*cut*-rule r up. We first commute, in π_1, r and the *cut*-rule, yielding $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1$. Applying the key case then yields $\pi_1^1 \xrightarrow{\overline{\beta}} \pi_1^2$; unless if r is a $\top$-rule, in which case there is nothing to do and we set $\pi_1^1 = \pi_1^2$; or if r is a &-rule, where we have to apply this step in both occurrences, obtaining $\pi_1^1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}} \pi_1^2$. In π_3, commuting r down the *cut*-rule created by the key case also yields π_1^2. Therefore, $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2 \xleftarrow{\overline{\beta}} \pi_3$.

If $\pi_2 \xrightarrow{\beta} \pi_3$ *is a* $\bot - 1$ *key case.* This case is also similar to the $⅋ - \otimes$ key case, in simpler as there are less sub-proofs. We have $\pi_2 = \dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1) \qquad \dfrac{\overset{\rho_1}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot)}{\vdash \Gamma}\,(cut)}{\vdash \rho_2}$ and $\pi_3 = \dfrac{\overset{\rho_1}{\vdash \Gamma}}{\vdash \rho_2}$ (up to symmetry of the *cut*-rule).

The 1-rule cannot commute, so the $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ step was a commutation pushing down the $\bot$-rule, and another non-*cut*-rule r up. We first commute, in π_1, r and the *cut*-rule, yielding $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1$. Applying the key case then yields $\pi_1^1 \xrightarrow{\overline{\beta}} \pi_3$, unless if r is a $\top$-rule, in which case we directly have $\pi_1^1 = \pi_3$, or if r is a &-rule, where we apply the key case on both occurrences, obtaining $\pi_1^1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}} \pi_3$. We conclude $\pi_1 \xrightarrow{\overline{\beta}^+} \pi_3$.

If $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ *is a commutative case.* As $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ and $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ have exactly one rule in common, the $\overset{r}{\vdash\!\dashv}$ step involves the rule r that will be commuted down in the $\xrightarrow{\overline{\beta}}$ step, and another rule that we call s (r and s are not *cut*-rules). The proof π_1 has from top to bottom r, s and *cut*, π_2 has s, r and *cut*, and π_3 has s, *cut* and r. Schematically:

$$\pi_1 = \dfrac{\dfrac{\dfrac{}{}\,(r)}{}\,(s)}{}\,(cut) \qquad \pi_2 = \dfrac{\dfrac{\dfrac{}{}\,(s)}{}\,(r)}{}\,(cut) \qquad \pi_3 = \dfrac{\dfrac{\dfrac{}{}\,(s)}{}\,(cut)}{}\,(r)$$

Suppose first that the *cut*-rule commutes with s. Our reasoning for this case is depicted on Figure 2.3. In π_1, we commute $s - cut$ then $r - cut$, yielding $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2$ (the $\xrightarrow{\overline{\beta}^*}$ being of length one, except if s is a &-rule, in which case we apply the $r - cut$ commutation for both occurrences, or if s is a $\top$-rule, in which case there is no commutation to apply). The proof π_1^2 has from top to bottom *cut*, r and s. Meanwhile, in π_3 we commute $s - cut$ (twice if r is a &-rule, or zero time if it is a $\top$-rule), yielding π_3^1 having from top to bottom *cut*, s and r. Now, both π_1^2 and π_3^1 have above r and s the same sub-proof(s), maybe duplicated or erased. We use weak normalization of $\xrightarrow{\overline{\beta}}$ (Corollary 2.43) to eliminate all *cut*-rules in this sub-proof(s), in the same way for all its occurrences in π_1^2 and π_3^1, obtaining proofs π_1^3 and π_3^2 equal up to the commutation of r and s (the very same one that was used in $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$). We thus obtain $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2 \xrightarrow{\overline{\beta}^*} \pi_1^3 \overset{r}{\vdash\!\dashv} \pi_3^2 \xleftarrow{\overline{\beta}^*} \pi_3^1 \xleftarrow{\overline{\beta}^*} \pi_3$.

To check that indeed we can apply a $\overset{r}{\vdash\!\dashv}$ step, by Lemma 2.58 r and s are independent for the conclusion sequent of s in π_1, so also for the conclusion sequent of the *cut*-rule (as their domains on the premise sequent of the *cut*-rule are their domains in the conclusion sequent, for none use the cut formula). Using commutative cut-elimination cases does not change this, nor does then eliminating all *cut*-rules above them (Items 2 and 3 of Lemma 2.57). Thus, r and s commute in π_1^3 (Lemma 2.58).

Assume now s is a rule associated to the main connective of the formula on which we cut. Remark s cannot be an *ax*-rule, for it commutes with r. This case is represented on Figure 2.4. We first reduce in the same way all *cut*-rules in the branch of the *cut*-rule not containing s, yielding π_1^1 from π_1 through $\pi_1 \xrightarrow{\overline{\beta}^*} \pi_1^1$ and π_3^1 from π_3 through $\pi_3 \xrightarrow{\overline{\beta}^*} \pi_3^1$; they share this sub-proof ρ_2, and we use weak normalization of $\xrightarrow{\beta}$ (Corollary 2.43). Denote by $s^\perp$ the rule introducing the dual formula of s (*i.e.* the other formula on which we cut), and by ρ the rules between $s^\perp$ and the *cut*-rule in π_1^1 (which are also those in π_3^1). Commuting the *cut*-rule above all rules in ρ yields $\pi_1^1 \xrightarrow{\beta^*} \pi_1^2$ with π_1^2 having the *cut*-rule between s and $s^\perp$. (Remark we do not need $\overset{c}{\vdash\!\dashv}$ steps, as we eliminated the *cut*-rules above this premise.) Doing the same commutations in π_3^1 gives $\pi_3^1 \xrightarrow{\beta^*} \pi_3^2$, with π_3^2 differing from π_1^2 by having r below ρ and not above s. Using the appropriate key case or $\top - cut$ commutative case to eliminate the *cut*-rule in π_1^2, using $s^\perp$ and maybe s, we obtain a new proof π_1^3 (as usual, if there are &-rules in ρ, we need to do so for all duplicates). In this new proof, if *cut*-rules have been introduced by the key case we used, we commute them with the rule r (which cannot introduce the formula of the *cut*-rule, for this is a sub-formula of the s rule, which commutes with the r rule). The produced proof is called π_1^4, and we have $\pi_1^2 \xrightarrow{\beta^+} \pi_1^3 \xrightarrow{\beta^*} \pi_1^4$. On the other hand, we also eliminate the *cut*-rule in the same way in π_3^2, yielding a proof π_3^3 such that $\pi_3^2 \xrightarrow{\beta^*} \pi_3^3$ (again, in all duplicates if there were &-rules in ρ, or with nothing to do if r is a $\top$-rule). Remark π_3^3 is π_1^4, except the rule r is below the rules of ρ in π_3^3 and above in π_1^4. Now we use again weak normalization of $\xrightarrow{\beta}$ (Corollary 2.43) to eliminates all *cut*-rules above r or the rules in ρ, in the same way in π_1^4 and π_3^3, obtaining respectively π_1^5 and π_3^4. These two last proofs are equal up to the commutation of r with all rules in ρ, so equal up to $\overset{r}{\vdash\!\dashv}$ now that there is no *cut*-rule above (including a $\top - \top$ commutation if both r and s are $\top$-rules). In case s or $s^\perp$ is a $\top$-rule, then $\pi_1^3 = \pi_1^4 = \pi_1^5$ and we commute s or $s^\perp$ with r in this proof, "producing" the rule r that we can then commute down with rules of ρ to recover $\pi_3^3 = \pi_3^4$.

Indeed, in π_1^1 the rule r is independent (for the sequent below the *cut*-rule) of every rule of ρ (Fact 2.56), ρ being a contextual block by definition (it consists of rules commuting with a given *cut*-rule, we keep for premises of $\otimes$- and mix_2-rules the one leading to the $s^\perp$ rules). They are still independent in π_1^2, π_1^3, π_1^4 and π_1^5 (Items 2 and 3 of Lemma 2.57). But then we remark r independent with the contextual block ρ, so it can indeed commute down (Lemma 2.60).

Finally, $\pi_1 \xrightarrow{\beta^*} \pi_1^1 \xrightarrow{\beta^*} \pi_1^2 \xrightarrow{\beta^+} \pi_1^3 \xrightarrow{\beta^*} \pi_1^4 \xrightarrow{\beta^*} \pi_1^5 \overset{r^*}{\vdash\!\dashv} \pi_3^4 \xleftarrow{\beta^*} \pi_3^3 \xleftarrow{\beta^*} \pi_3^2 \xleftarrow{\beta^*} \pi_3^1 \xleftarrow{\beta^*} \pi_3$. □

For local coherence with $\overset{c}{\vdash\!\dashv}$, we will need one intermediate result on symmetry of cut, due to the following remark.

Remark 2.63. The **symmetry of a *cut*-rule** is the identification of the following (sub-)proofs:

$$\pi_1 = \dfrac{\overset{\rho_1}{\vdash A, \Gamma} \qquad \overset{\rho_2}{\vdash A^\perp, \Delta}}{\vdash \Gamma, \Delta}\ (cut) \quad \text{and} \quad \pi_2 = \dfrac{\overset{\rho_2}{\vdash A^\perp, \Delta} \qquad \overset{\rho_1}{\vdash A, \Gamma}}{\vdash \Gamma, \Delta}\ (cut)$$

This relation is included in $=_\beta$, using a $cut - cut$ commutation. Indeed, define respectively π_1' and π_2' the two following proofs, where A_1 and A_2 are both the formula A, with indices to follow their occurrences:

$$\dfrac{\dfrac{\dfrac{}{\vdash A_1^\perp, A_2}\ (ax) \qquad \overset{\rho_1}{\vdash A_1, \Gamma}}{\vdash A_2, \Gamma}\ (cut) \qquad \overset{\rho_2}{\vdash A_2^\perp, \Delta}}{\vdash \Gamma, \Delta}\ (cut) \quad \text{and} \quad \dfrac{\dfrac{\dfrac{}{\vdash A_1^\perp, A_2}\ (ax) \qquad \overset{\rho_2}{\vdash A_2^\perp, \Delta}}{\vdash A_1^\perp, \Delta}\ (cut) \qquad \overset{\rho_1}{\vdash A_1, \Gamma}}{\vdash \Gamma, \Delta}\ (cut)$$

Then $\pi_1 \xleftarrow{\beta} \pi_1' \overset{c}{\vdash\!\dashv} \pi_2' \xrightarrow{\beta} \pi_2$, these $\xrightarrow{\beta}$ steps being ax key cases.

The next lemma allows us not to add symmetry of cut in $\overset{c}{\vdash\!\dashv}$, which would complexify our proofs where there already are a lot of cases.

Lemma 2.64. *Let π_1 and π_2 be two proofs equal up to symmetry of cut-rules. Then $\pi_1 \xrightarrow{\overline{\beta}^*} \cdot \xleftarrow{\overline{\beta}^*} \pi_2$.*

Proof. The idea is to eliminate any *cut*-rule in π_1 and π_2 in a symmetric way, and show the resulting proofs are equal up to symmetry of *cut*-rules. The wished result follows through weak normalization, as two cut-free proofs equal up to symmetry of *cut*-rules are simply equal.

We reason by induction on a sequence $\pi_1 \xrightarrow{\overline{\beta}^*} \rho$, with ρ some cut-free proof found by weak normalization (Corollary 2.43). If this sequence is empty, then π_1 is *cut*-free and so $\pi_1 = \pi_2$. Thus, take $\pi_1 \xrightarrow{\overline{\beta}} \pi_1'$ its first step. We apply the corresponding step in π_2, on the corresponding *cut*-rule which may be the symmetric version of c (for if a $\xrightarrow{\overline{\beta}}$ step can be applied, then the one with switched premises can be applied on the symmetric version). We obtain a proof π_2' with the *cut*-rule still permuted compared to π_1' in case of a commutative case (or the *cut*-rule erased when commuting with a $\top$, or permuted and duplicated with a $\&$) and 0, 1 or 2 symmetric *cut*-rules resulting from a key case. In all cases, π_1' and π_2' are equal up to symmetry of *cut*-rules, allowing us to conclude by induction hypothesis. ☐

Lemma 2.65. *Let π_1, π_2 and π_3 be proofs such that $\pi_1 \overset{c}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. Then there exist π_1' and π_2' such that $\pi_1 \xrightarrow{\overline{\beta}^+} \pi_1' \overset{c^*}{\vdash\!\dashv} \pi_2' \xleftarrow{\overline{\beta}^*} \pi_3$. Diagrammatically:*

$$
\begin{tikzcd}
\pi_1 \arrow[r, phantom, "\overset{c}{\vdash\!\dashv}"] \arrow[dd, dashed, "\overline{\beta}^+"'] & \pi_2 \arrow[dr, "\overline{\beta}"] & \\
 & & \pi_3 \arrow[dl, dashed, "\overline{\beta}^*"] \\
\pi_1' \arrow[r, phantom, "\overset{c^*}{\vdash\!\dashv}"] & \pi_2' &
\end{tikzcd}
$$

Proof. Call c the (unique) *cut*-rule of π_2 involved in $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$, and c_d and c_u the ones of π_2 from top to bottom involved in $\pi_1 \overset{c}{\vdash\!\dashv} \pi_2$ (so that in π_1 we have c_u above c_d).

If $c \notin \{c_u; c_d\}$, then there exists ρ such that $\pi_1 \xrightarrow{\overline{\beta}} \rho \overset{c^*}{\vdash\!\dashv} \pi_3$. Indeed, the commutation $\pi_1 \overset{c}{\vdash\!\dashv} \pi_2$ involves no rule of the $\xrightarrow{\overline{\beta}}$ step, meaning we can do the $\xrightarrow{\overline{\beta}}$ step first. If this $\xrightarrow{\overline{\beta}}$ step does not erase nor duplicate a sub-proof containing c_u and c_d, then we have the result by first doing the $\xrightarrow{\overline{\beta}}$ step, then the $cut - cut$ commutation: $\pi_1 \xrightarrow{\overline{\beta}} \cdot \overset{c}{\vdash\!\dashv} \pi_3$. If the $\xrightarrow{\overline{\beta}}$ step erases c_u and c_d, then $\pi_1 \xrightarrow{\overline{\beta}} \pi_3$. Finally, if $\xrightarrow{\overline{\beta}}$ duplicates the rules c_u and c_d, then we apply the $cut - cut$ commutation on each copy to recover π_3, yielding $\pi_1 \xrightarrow{\overline{\beta}} \cdot \overset{c}{\vdash\!\dashv} \cdot \overset{c}{\vdash\!\dashv} \pi_3$. Thus, if $c \notin \{c_u; c_d\}$ then $\pi_1 \xrightarrow{\overline{\beta}} \rho \overset{c^*}{\vdash\!\dashv} \pi_3$.

Whence, we now assume $c \in \{c_u; c_d\}$. Call $\mathcal{R}$ the set of the non-*cut*-rules involved in $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ (*i.e.* those other than c). Also call c' the *cut*-rule in $\{c_u; c_d\}$ other than c. Looking at the possible cases for $\xrightarrow{\overline{\beta}}$, we have $\mathcal{R}$ of cardinal 1 or 2. If all rules of $\mathcal{R}$ are just above c in π_1 then, as it is also the case in π_2, $\mathcal{R}$ must be a singleton $\{r\}$ (because c' must be just above c in π_1 or in π_2), and so $\xrightarrow{\overline{\beta}}$ is a commutative step or an ax key case. Assume r is an ax-rule; we have two sub-cases, according to $(c, c') = (c_u, c_d)$ or $(c', c) = (c_u, c_d)$. If $(c, c') = (c_u, c_d)$, then we have (up to symmetry of *cut*-rules):

- $$\pi_1 = \cfrac{\cfrac{\cfrac{}{\vdash A^\perp, A}\,(r) \qquad \overset{\rho_1}{\vdash A, B^\perp, \Gamma}}{\vdash A, B^\perp, \Gamma}\,(c) \qquad \overset{\rho_2}{\vdash B, \Delta}}{\vdash \Gamma, \Delta}\,(c')$$

- $$\pi_2 = \dfrac{\dfrac{}{\vdash A^\perp, A}\,(r) \qquad \dfrac{\begin{matrix}\rho_1\\ \vdash A, B^\perp, \Gamma\end{matrix} \qquad \begin{matrix}\rho_2\\ \vdash B, \Delta\end{matrix}}{\vdash A, \Gamma, \Delta}\,(c')}{\vdash \Gamma, \Delta}\,(c)$$

- $$\pi_3 = \dfrac{\begin{matrix}\rho_1\\ \vdash A, B^\perp, \Gamma\end{matrix} \qquad \begin{matrix}\rho_2\\ \vdash B, \Delta\end{matrix}}{\vdash A, \Gamma, \Delta}\,(c')$$

We remark $\pi_1 \xrightarrow{\beta} \pi_3$. Similarly, if $(c', c) = (c_u, c_d)$ then $\pi_1 \xrightarrow{\beta} \pi_3$.

Suppose now that $\pi_2 \xrightarrow{\beta} \pi_3$ is a commutative step. Remark that r above c' is a commutative cut-elimination case, for we have c' not below r in either π_1 or π_2. We also have two sub-cases here, according to $(c, c') = (c_u, c_d)$ or $(c', c) = (c_u, c_d)$.

In the first sub-case, $(c, c') = (c_u, c_d)$, we have schematically:

$$\pi_1 = \dfrac{\dfrac{\dfrac{}{}\,(r)}{}\,(c)}{}\,(c') \qquad \pi_2 = \dfrac{\dfrac{}{}\,(r) \quad \dfrac{}{}\,(c')}{}\,(c) \qquad \pi_3 = \dfrac{\dfrac{\dfrac{}{}\,(c')}{}\,(c)}{}\,(r)$$

(with in π_3 rules above r erased if r is a $\top$-rule, or duplicated if r is a $\&$-rule). We apply in π_1 the commutative cut-elimination case between r and c, yielding $\pi_1^1 = \dfrac{\dfrac{\dfrac{}{}\,(c)}{}\,(r)}{}\,(c')$, then the same one between r and c' to get $\pi_1^2 = \dfrac{\dfrac{\dfrac{}{}\,(c)}{}\,(c')}{}\,(r)$ and finally one $\overset{c}{\vdash\!\dashv}$ step (or zero if $r = \top$, or two if $r = \&$), obtaining π_3. Thence, $\pi_1 \xrightarrow{\beta} \pi_1^1 \xrightarrow{\beta} \pi_1^2 \overset{c^*}{\vdash\!\dashv} \pi_3$.

In the second sub-case, $(c', c) = (c_u, c_d)$, we have:

$$\pi_1 = \dfrac{\dfrac{}{}\,(r) \quad \dfrac{}{}\,(c')}{}\,(c) \qquad \pi_2 = \dfrac{\dfrac{\dfrac{}{}\,(r)}{}\,(c)}{}\,(c') \qquad \pi_3 = \dfrac{\dfrac{\dfrac{}{}\,(c)}{}\,(r)}{}\,(c')$$

Similarly to the previous sub-case, $\pi_1 \xrightarrow{\beta} \pi_1^1 = \dfrac{\dfrac{\dfrac{}{}\,(c')}{}\,(c)}{}\,(r)$ and $\pi_3 \xrightarrow{\beta} \pi_3^1 = \dfrac{\dfrac{\dfrac{}{}\,(c)}{}\,(c')}{}\,(r)$. Then, $\pi_1^1 \overset{c^*}{\vdash\!\dashv} \pi_3^1$ by commuting c and c' (using one $cut - cut$ commutation, or zero if $r = \top$, or two if $r = \&$). Thus, $\pi_1 \xrightarrow{\beta} \pi_1^1 \overset{c^*}{\vdash\!\dashv} \pi_3^1 \xleftarrow{\beta} \pi_3$. This solves the case where all rules of $\mathcal{R}$ are above c in π_1.

We suppose now that a rule $r \in \mathcal{R}$ is above c' in π_1, and then above c in π_2, thus $c' = c_u$ and $c = c_d$. In this case, the possible other rule in $\mathcal{R}$ cannot be above c' in π_1, for only one sub-proof may be shared between the two *cut*-rules. Schematically:

$$\pi_1 = \dfrac{\dfrac{\dfrac{}{}\,(r)}{}\,(c')}{}\,(c) \qquad \pi_2 = \dfrac{\dfrac{\dfrac{}{}\,(r)}{}\,(c)}{}\,(c') \qquad \pi_3 = \dfrac{\dfrac{}{}\,(\gamma)}{}\,(c')$$

with γ the resulting rules of applying the $\xrightarrow{\beta}$ step, which may be a key case.

If r and c' commute (with a commutative $\xrightarrow{\beta}$ step), then we apply this commutation in π_1, yielding $\pi_1^1 = \dfrac{\dfrac{\dfrac{}{}\,(c')}{}\,(r)}{}\,(c)$. Then we apply the $\xrightarrow{\beta}$ step with the rules corresponding to the ones of $\pi_2 \xrightarrow{\beta} \pi_3$, yielding $\pi_1^2 = \dfrac{\dfrac{}{}\,(c')}{}\,(\gamma)$. At this point, either this step was a commutative step with r, meaning γ is composed of c above r, and we can in π_3 commute c' and r with a $\xrightarrow{\beta}$ step, then c' and c with one $\overset{c}{\vdash\!\dashv}$ step, in order to recover π_1^2 (or no $\overset{c}{\vdash\!\dashv}$ needed if $r = \top$, or two if $r = \&$). Otherwise, this reduction step was a key case, producing γ which consists in 0, 1 or 2 *cut*-rules. We commute them with c' so as to obtain π_3. Therefore, if r and c' commute then $\pi_1 \xrightarrow{\overline{\beta}} \pi_1^1 \xrightarrow{\overline{\beta}} \pi_1^2 \overset{c^*}{\vdash\!\dashv} \cdot \xleftarrow{\overline{\beta}^=} \pi_3$.

Suppose now that r and c' do not commute. This means r is a rule corresponding to the main connective of the formula A on which c' cuts, a $\top$-rule or an ax-rule on A. If it is an ax-rule, then r and c, as well as r and c', make an ax key case. One can check that applying this key case with c' in π_1 yields π_3 (in this case γ is empty), maybe up to symmetry of the *cut*-rule. Thus $\pi_1 \xrightarrow{\overline{\beta}} \pi_3$ or, using Lemma 2.64, $\pi_1 \xrightarrow{\overline{\beta}} \cdot \xrightarrow{\overline{\beta}^*} \cdot \xleftarrow{\overline{\beta}^*} \pi_3$.

Assume now r is not an ax-rule, so it is a rule corresponding to the main connective of the formula A on which c' cuts, or a $\top$-rule. Thus $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$ is not a key step, because r cannot also correspond to the main connective of the formula on which c cuts (only the ax-rule has two principal formulas). Hence in the schema of π_3, γ consists of c followed below by r. A schematic representation of the reasoning we will do is depicted on Figure 2.5.

Call ρ the sub-proof of π_1 above c' in the branch not leading to r. This proof ρ is also the sub-proof of π_3 above c' in the same branch (c does not belong to this branch as it commuted with r). We reduce all *cut*-rules in ρ using only $\xrightarrow{\overline{\beta}}$ steps (thanks to Corollary 2.43), in the same way in both π_1 and π_3, obtaining a cut-free proof ρ', which is a sub-proof of the resulting π_1^1 and π_3^1 respectively. In particular, we have $\pi_1 \xrightarrow{\overline{\beta}^*} \pi_1^1$ and $\pi_3 \xrightarrow{\overline{\beta}^*} \pi_3^1$.

Call $r^\perp$ the rule in ρ' introducing the formula $A^\perp$ (which is a rule of the main connective of $A^\perp$ or an ax or $\top$-rule), and τ the sequence of rules in ρ' between c' and $r^\perp$. Remark that rules in τ cannot be $\top$ or *cut*-rules. We commute in π_1^1 (resp. π_3^1) the rules of τ with c', yielding $\pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2$ (resp. $\pi_3^1 \xrightarrow{\overline{\beta}^*} \pi_3^2$) as τ is cut-free. In the two proofs obtained, c' has on its premises the rules r and $r^\perp$ (or all duplicated c', as τ may contain &-rules).

We now apply a key or $\top - cut$ commutative case on c', using r and $r^\perp$ (or just $r^\perp$ if it is an ax or $\top$-rule), in all slices as there can be &-rules in τ, yielding $\pi_1^2 \xrightarrow{\overline{\beta}^+} \pi_1^3$ and $\pi_3^2 \xrightarrow{\overline{\beta}^+} \pi_3^3$, producing rules γ' (in each slice), which are 0, 1 or 2 *cut*-rules, or the $\top$-rule $r^\perp$. Observe that π_1^3 and π_3^3 differ only by the fact that in π_1^3 the *cut*-rule c is below γ' and τ, while it is above (or erased in case of a $\top - cut$ commutative case) in π_3^3 (in all slices). We can commute c up in π_1^3 until going above τ, using $\xrightarrow{\overline{\beta}}$ steps, obtaining π_1^4; this is because rules in τ cannot introduce the formula on which c cuts, looking at π_3^3. Then we commute c with γ' to recover π_3^3, using 0, 1 or 2 $\overset{c}{\vdash\!\dashv}$ steps or a $\top - cut$ commutation $\xrightarrow{\overline{\beta}}$, in all slices as they can be &-rules in τ. Whence $\pi_1 \xrightarrow{\overline{\beta}^*} \pi_1^1 \xrightarrow{\overline{\beta}^*} \pi_1^2 \xrightarrow{\overline{\beta}^+} \pi_1^3 \xrightarrow{\overline{\beta}^*} \pi_1^4 (\xrightarrow{\overline{\beta}^+} \cup \overset{c^*}{\vdash\!\dashv}) \pi_3^3 \xleftarrow{\overline{\beta}^+} \pi_3^2 \xleftarrow{\overline{\beta}^*} \pi_3^1 \xleftarrow{\overline{\beta}^*} \pi_3$, and in particular $\pi_1 \xrightarrow{\overline{\beta}^+} \cdot \overset{c^*}{\vdash\!\dashv} \cdot \xleftarrow{\overline{\beta}^*} \pi_3$. ∎

We can now prove the wished result.

Theorem 2.47. *In* $\mathsf{MALL}^{0,2}$*:*

- $\xrightarrow{\overline{\beta}}$ *is Church-Rosser modulo* $(\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$*.*
- $\xrightarrow{\beta}$ *is Church-Rosser modulo* $\overset{r^*}{\vdash\!\dashv}$*.*

Proof. For the first point, apply Theorem 2.7 with $\vdash\!\dashv\ = (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})$, $\rightarrow\ = \xrightarrow{\overline{\beta}}$ and $\rightsquigarrow\ = (\overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv})$. We deduce Item (i) from Proposition 2.41, Item (ii) from Lemma 2.61 and Item (iii) from Lemmas 2.62 and 2.65.

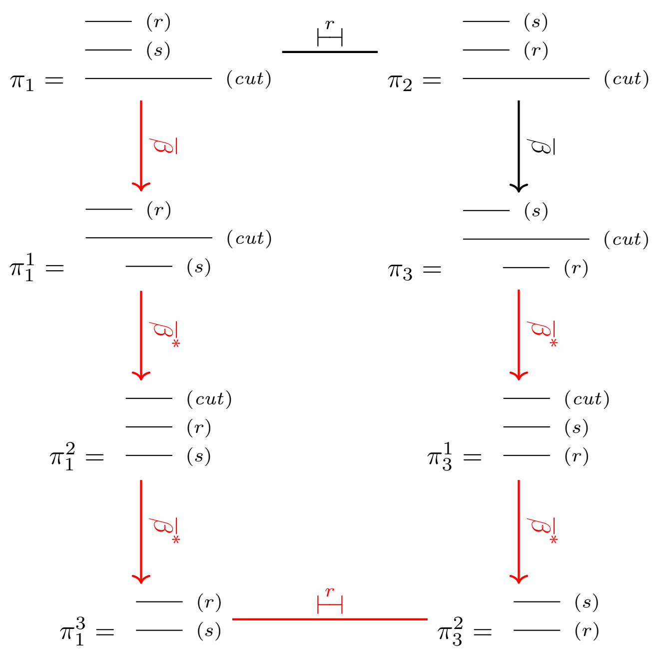


Figure 2.3: Schematic representation of the second-to-last case in the proof of Lemma 2.62

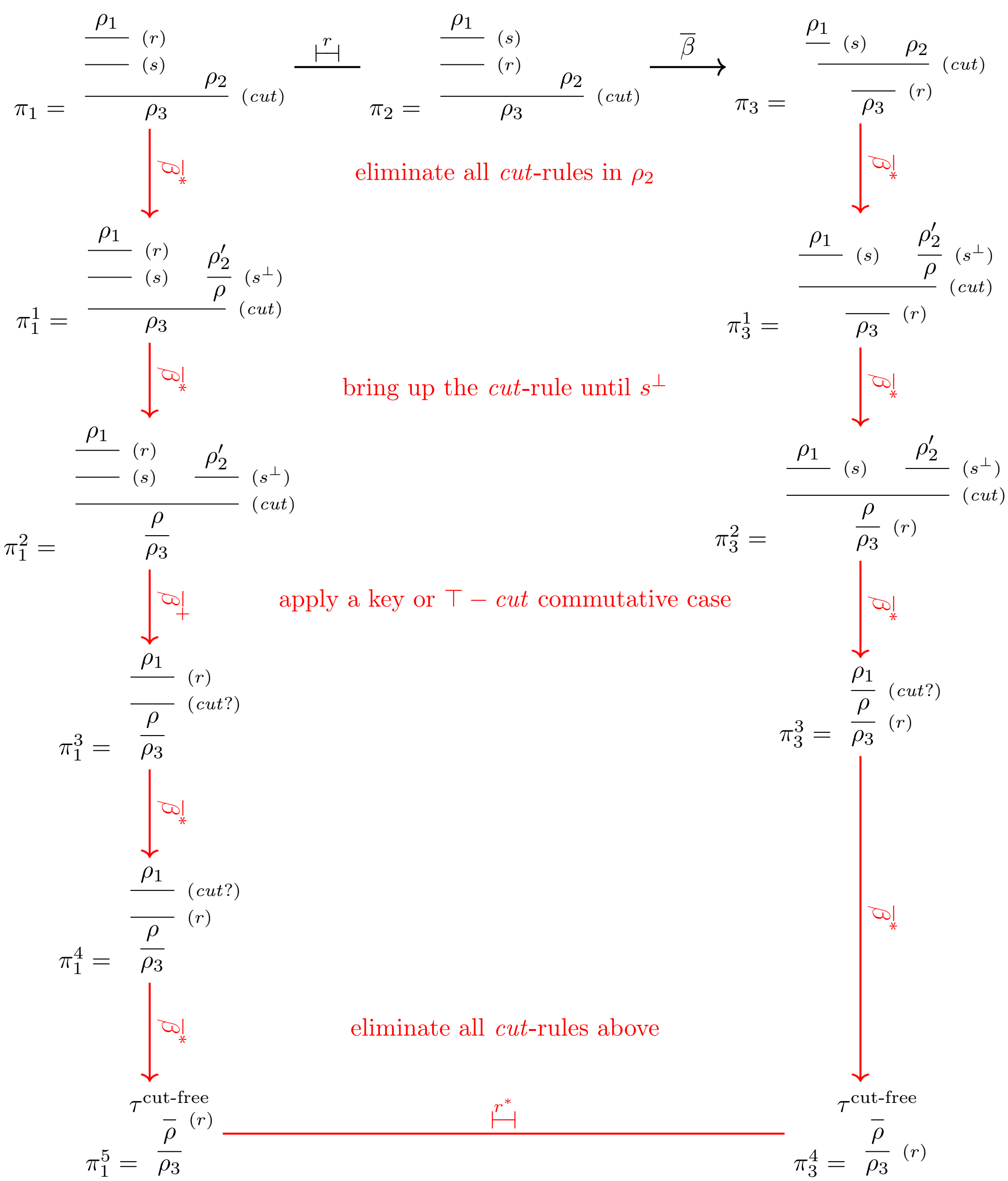


Figure 2.4: Schematic representation of the last case in the proof of Lemma 2.62

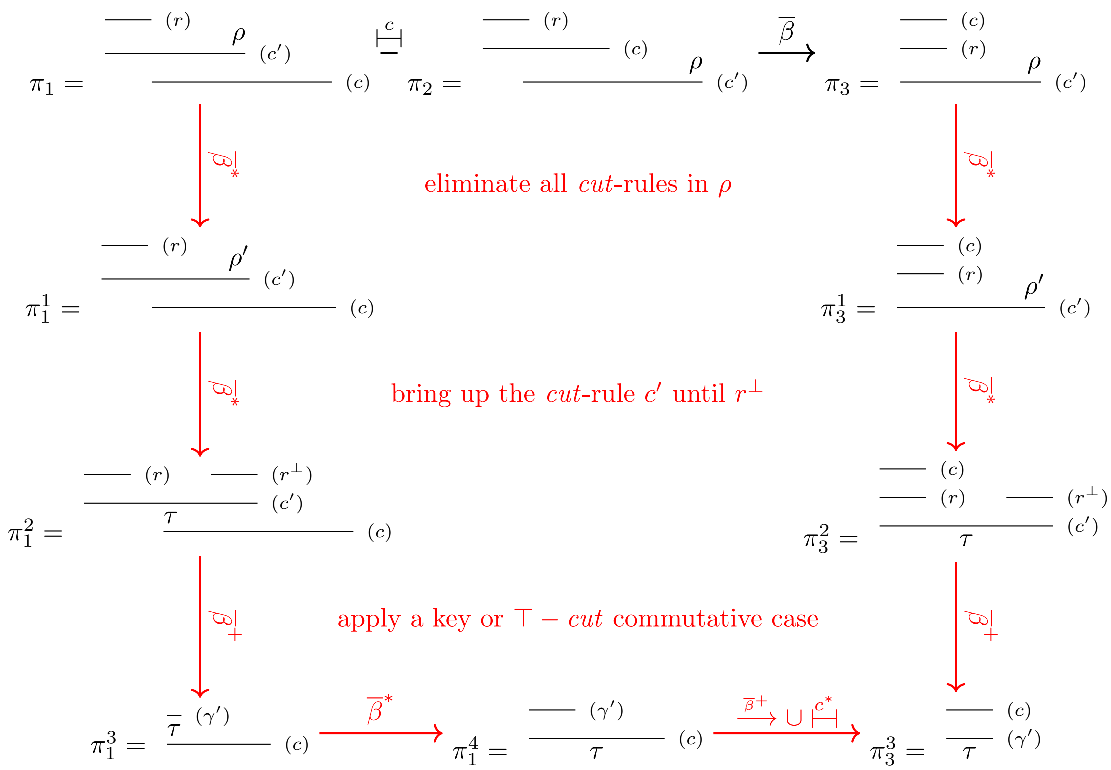


Figure 2.5: Schematic representation of the last case in the proof of Lemma 2.65

We deduce the second item from the first one. Consider two proofs π and ϕ equal up to cut-elimination and rule commutation. Take π' and ϕ' respective β-normal forms for π and ϕ (Proposition 2.41). Thus, π' and ϕ' are equal up to cut-elimination and rule commutation. By the first item, $\pi'\ (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*\ \phi'$. Being cut-free, they can only be related by $\overset{r^*}{\vdash\!\dashv}$, for these relations cannot create *cut*-rules. Therefore, $\xrightarrow{\beta}$ is Church-Rosser modulo $\overset{r^*}{\vdash\!\dashv}$. □

Remark 2.66. Extending the results of this section to the full system, with all optional rules, is easy. It suffices to extend the definition of a contextual rule (Definition 2.54) to the $?_d$-, $?_c$-, $?_w$-, $\forall$-, $\exists$-, $\cup$- and $\emptyset$-rules trivially, with a !-rule not considered (as it is the case for *ax*- and 1-rules for instance). The first 5 rules have at most one child in a contextual block, $\cup$ has at most two and $\emptyset$ none.

Considering the definition of domains (Definition 2.55), there is some care to take for the $?_c$- and $\exists$-rules. For a $?_c$-rule of the shape $\dfrac{\vdash ?A_1, ?A_2, \Delta}{\vdash ?A_1, \Delta}\ (?_c)$, we want the contracted occurrence $?A_2$ to be identified with the occurrence $?A_1$ in the sequent $\vdash \Gamma$ for which we look at, so that a rule using either of $?A_1$ and $?A_2$ cannot be independent with the $?_c$-rule. In particular, remark that another $?_c$-rule on $?A_1$ or $?A_2$ does *not* commute with the $?_c$-rule above. For the $\exists$-rule $\dfrac{\vdash A[B/X], \Delta}{\vdash \exists X A, \Delta}\ (\exists)$, we need the occurrence $A[B/X]$ to be assimilated to $\exists X A$, as was $?A_2$ to $?A_1$ in the $?_c$ case.

The rest is simply a matter of checking that our proofs of Fact 2.56 and Lemmas 2.57, 2.58 and 2.60 stay valid in this extended framework. One can then prove Lemmas 2.61, 2.62 and 2.65 as we did here. Remark a $! - cut$ commutation poses no problem: such a commutation can only be done if one premise of the *cut*-rule is a !-rule on the cut formula, in which case no commutative case can be done using the rule on the other premise (and there is no commutation rule involving a !-rule).

Therefore, the main difficulty when proving Theorem 2.47 is to prove strong normalization. In particular, provided Conjecture 2.45 holds one readily deduces that cut-elimination is Church-Rosser modulo rule commutations in the full system.

2.3.6.3 Cut-elimination and *mix*-Rétoré is Church-Rosser

Once Theorem 2.47 proven, it is easy to prove Theorem 2.48 by reusing lemmas from this proof. The next lemma lists and proves all hypotheses needed to apply Theorem 2.7 in this case, including those already proved last section.

Lemma 2.67.

(i) *The relation* $\xrightarrow{\overline{\beta}} \cdot (\overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$ *is strongly normalizing.*

(ii) (a) *Let* π, π_1 *and* π_2 *be proofs such that* $\pi_1 \xleftarrow{\overline{\beta}} \pi \xrightarrow{\overline{\beta}} \pi_2$. *Then* $\pi_1 \xrightarrow{\overline{\beta}^*} \cdot \overset{r^=}{\vdash\!\dashv} \cdot \xleftarrow{\overline{\beta}^*} \pi_2$.

(b) *Let* π, π_1 *and* π_2 *be proofs such that* $\pi_1 \overset{om}{\leftsquigarrow} \pi \xrightarrow{\overline{\beta}} \pi_2$. *Then* $\pi_1 \xrightarrow{\overline{\beta}^=} \cdot \overset{om^*}{\leftsquigarrow} \pi_2$.

(c) *The relation* $\overset{om}{\rightsquigarrow}$ *has the diamond property.*

(iii) (a) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. *Then, there exist* π_1' *and* π_2' *such that* $\pi_1 \xrightarrow{\overline{\beta}^+} \pi_1' \overset{r^*}{\vdash\!\dashv} \pi_2' \xleftarrow{\overline{\beta}^*} \pi_3$. *Moreover, this* $\overset{r^*}{\vdash\!\dashv}$ *step consists in one rule commuting with a succession of other rules; in particular it is a* $\overset{r\setminus\top}{\rightsquigarrow}{}^*$ *step (in one direction or the other).*

(b) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{c}{\vdash\!\dashv} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. *Then* $\pi_1 \xrightarrow{\overline{\beta}+} \cdot \overset{c^*}{\vdash\!\dashv} \cdot \xleftarrow{\overline{\beta}*} \pi_3$.

(c.1) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{om}{\rightsquigarrow} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. *Then* $\pi_1 \xrightarrow{\overline{\beta}+} \cdot \overset{om^*}{\rightsquigarrow} \pi_3$.

(c.2) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{om}{\leftsquigarrow} \pi_2 \xrightarrow{\overline{\beta}} \pi_3$. *Then* $\pi_1 \xrightarrow{\overline{\beta}+} \cdot \overset{om^*}{\leftsquigarrow} \pi_2$.

(d) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2 \overset{om}{\rightsquigarrow} \pi_3$. *Then* $\pi_1 \overset{r\backslash\top}{\rightsquigarrow} \cdot \overset{om}{\rightsquigarrow} \pi_3$ *or* $\pi_1 \overset{om^{=}}{\rightsquigarrow} \cdot \overset{r\backslash\top^{=}}{\leftsquigarrow} \pi_3$.

(e) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{c}{\vdash\!\dashv} \pi_2 \overset{om}{\rightsquigarrow} \pi_3$. *Then* $\pi_1 \overset{om}{\rightsquigarrow} \cdot \overset{c}{\vdash\!\dashv} \pi_3$.

(f.1) *Let* π_1, π_2 *and* π_3 *be proofs such that* $\pi_1 \overset{om}{\rightsquigarrow} \pi_2 \overset{om}{\rightsquigarrow} \pi_3$. *Then* $\pi_1 \overset{om}{\rightsquigarrow} \cdot \overset{om}{\rightsquigarrow} \pi_3$.

(f.2) *The relation* $\overset{om}{\rightsquigarrow}$ *has the diamond property.*

Proof.

(i) This is Proposition 2.41.

(ii) (a) This is Lemma 2.61.

(b) There are two cases here. If the $\xrightarrow{\overline{\beta}}$ step is not a $mix_2 - cut$ commutative case on the mix_2-rule erase in the $\overset{om}{\rightsquigarrow}$ step, then the two steps commute, and $\pi_1 \xrightarrow{\overline{\beta}+} \cdot \overset{om^*}{\leftsquigarrow} \pi_2$ using the very same steps (with $\overset{om}{\leftsquigarrow}$ done once, except if its rules are erased by the cut-elimination step in which case it is not done, or if its rules have been duplicated then it is done twice). Otherwise, we directly have $\pi_1 \overset{om}{\leftsquigarrow} \pi_3$, applying this steps on the same rule as those of $\pi_1 \overset{om}{\leftsquigarrow} \pi$.

(c) If two steps of $\overset{om}{\rightsquigarrow}$ acts on the same mix_2-rule, then the results are equal as the two ways of reducing $\dfrac{\dfrac{}{\vdash}(mix_0) \qquad \dfrac{}{\vdash}(mix_0)}{\vdash}(mix_2)$ give $\dfrac{}{\vdash}(mix_0)$. Otherwise, the steps commute and $\overset{om}{\leftsquigarrow} \cdot \overset{om}{\rightsquigarrow} = \overset{om}{\rightsquigarrow} \cdot \overset{om}{\leftsquigarrow}$.

(iii) (a) This is Lemma 2.62.

(b) This is Lemma 2.65.

(c.1) In π_1, one can apply the cut-elimination step corresponding to $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$, maybe doing a $cut - mix_2$ commutative case first if the mix_2-rule erased in the $\pi_1 \overset{om}{\rightsquigarrow} \pi_2$ step is above the (bottom) cut-rule of $\pi_2 \xrightarrow{\overline{\beta}} \pi_3$. This yields a proof π', in which one can then apply a $\overset{om}{\rightsquigarrow}$ step on the mix_2-rule of $\pi_1 \overset{om}{\rightsquigarrow} \pi_2$ – maybe twice if its rules have been duplicated by the $\pi_1 \xrightarrow{\overline{\beta}} \pi'$ step, or with nothing to do if its rules have been erased by this cut-elimination step.

(c.2) This is the same as (ii)(b).

(d) If the $\pi_1 \overset{r}{\vdash\!\dashv} \pi_2$ is a $\pi_1 \overset{r\backslash\top}{\rightsquigarrow} \pi_2$ step, then we are done. Otherwise, it is a $\top$ commutation adding rules in π_2, and in particular a $\overset{r\backslash\top}{\leftsquigarrow}$ step. As it cannot add a mix_0-rule without adding a rule below it, either both rules of the $\pi_2 \overset{om}{\rightsquigarrow} \pi_3$ step are rules created by the commutation, or they were already in π_1. In the latter case, one can apply the $\overset{om}{\rightsquigarrow}$ step first and then the $\overset{r\backslash\top}{\leftsquigarrow}$ step: $\pi_1 \overset{om}{\rightsquigarrow} \cdot \overset{r\backslash\top}{\leftsquigarrow} \pi_3$. In the former case, we must have a $\top - \otimes$ or $\top - mix_2$ commutation introducing a $\dfrac{\vdash \qquad \dfrac{}{\vdash}(mix_0)}{\vdash}(mix_2)$ in its sub-proof. We can directly introduce the sub-proof without these two rules, leading to $\pi_1 \overset{r\backslash\top}{\leftsquigarrow} \pi_3$, or $\pi_1 = \pi_3$ in case this was the whole sub-proof introduced.

(e) Two such steps always commute, as doing one cannot prevent doing the other: $\pi_1 \overset{om}{\rightsquigarrow} \cdot \overset{c}{\vdash\!\dashv} \pi_3$.

(f.1) Nothing to prove here.

(f.2) This is the same as (ii)(c).

□

Theorem 2.48. *In* $\mathsf{MALL}^{0,2}$*:*

- $\xrightarrow{\overline{\beta}} \cup \overset{om}{\rightsquigarrow}$ *is Church-Rosser modulo* $(\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*$.
- $\xrightarrow{\beta} \cup \overset{om}{\rightsquigarrow}$ *is Church-Rosser modulo* $(\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*$.

Proof. For the first point, we apply Theorem 2.7 with $\vdash\!\dashv\; = \;(\overset{\rho m}{\vdash\!\dashv} \cup \overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})$, $\rightarrow\; = \;(\xrightarrow{\overline{\beta}} \cup \overset{om}{\rightsquigarrow})$ and $\rightsquigarrow\; = \;(\overset{om}{\rightsquigarrow} \cup \overset{r \setminus \top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv})$. All needed hypotheses can be deduced from Lemma 2.67.

We deduce the second item from the first one, as in the proof of Theorem 2.47. Consider two proofs π and ϕ equal up to cut-elimination and rule commutation. Take π' and ϕ' respective β-normal forms for π and ϕ (Proposition 2.41). They are equal up to cut-elimination and rule commutation, so by the first item π' $(\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ ϕ'. Being cut-free, they can only be related by $\overset{r^*}{\vdash\!\dashv}$, for these relations cannot create *cut*-rules. □

Again, extending this result to the full system, with any of the Rétoré transformations, should be doable (*c.f.* Remark 2.66).

2.3.7 Comparison with the work of Cockett and Pastro

Willem Heijltjes, one of the rapporteurs of this thesis, brought to our attention some earlier work on MALL by Cockett and Pastro, with a term language for this logic on which has been showed the Church-Rosser property [CP05]. While some results of this paper are really close to what has been done in this Section 2.3, there are some key differences.

- Whereas here is considered the unilateral sequent calculus, in [CP05] everything is done through the lens of a term calculus corresponding to a bilateral sequent calculus for MALL. Still, this should not be a big difference as it is quite easy to check that the two transformations of the term calculus indeed correspond to cut-elimination and rule commutation in the bilateral sequent calculus, which is “equivalent” to the unilateral one. Nonetheless, our approach directly in the sequent calculus makes the result explicit, without the need of an implicit result linking the term and sequent calculi. A bigger difference is that in [CP05] are considered generalized axioms (on a list of atoms), that we do not have in this thesis.

- There seems to be a small mistake in their proof, for [CP05, Proposition B.2 (ii)] is false: an interchange (*i.e.* rule commutations in the term calculus) not involving a unit does not necessarily preserve their measure Λ on terms (*i.e.* proofs). Indeed, the authors took as measure a multiset of a quantity for each cut, but what corresponds to a $\& - \otimes$ commutation in their calculus – rule (18) – may duplicate a *cut*-term, and thus increase the measure of the term. This error could be corrected as done here, by considering only rule commutations with no *cut*-term inside the rewritten part of the term.

- The authors of [CP05] do not consider directly the $cut - cut$ commutation in their cut-elimination, using smartly that this commutation is easily seen as included in $\overline{\beta}$-equality ([CP05, Lemmas C.2 and C.3]). This allows for a simpler proof, with no more considerations about $\overset{c}{\vdash\!\dashv}$. Unfortunately, this approach can hardly be extended in presence of the exponentials. This is because their demonstration goes by induction on the terms, and in MALL a cut-elimination step reduces the size of the terms or proofs above a cut. This is false with the $?_c - !$ key elimination case where the proof above the !-rule is duplicated, hence the sizes of the proofs above the lower cut-rule increase. Meanwhile, our approach should extend to full linear logic (see Remark 2.66).

- Some of their proofs are simpler, for they do not consider a commutation with a linear block. We could have done the same here, but as we would still need such commutations in the presence of exponentials, where key cases may introduce many rules, we chose to keep them.

- We prove some more results that those present in [CP05], namely that $\overset{rc}{\vdash\!\dashv} \subseteq =_\beta$ (Proposition 2.46), the strong normalization in MALL of $\xrightarrow{\overline{\beta}} \cdot (\overset{r}{\vdash\!\dashv} \cup \overset{c}{\vdash\!\dashv})^*$ (using that our measure is preserved by rule commutations, while theirs is preserved only by rule commutations not involving a unit rule, and thanks to looking directly at $cut - cut$ commutations) and we take into account the possible presence of the mix_2- and mix_0-rules as well as the mix-Rétoré transformation $\overset{om}{\rightsquigarrow}$ whereas they do not.

2.4 Perspectives

Two results were proved in this chapter. The first is that equality up to identification, *i.e.* up to axiom-expansion, cut-elimination and some Rétoré transformations, can be reduced to the same problem without axiom-expansion – except with second order quantifiers, where one should expand on the fly proofs after using a $\forall - \exists$ key cut-elimination step.

The second, and more important, proof of this chapter is that cut-elimination is Church-Rosser modulo rule commutation in $\mathsf{MALL}^{0,2}$, using as an intermediate lemma that composing commutations with non $cut - cut$ cut-elimination steps is strongly normalizing. The generalization of this result to full linear logic, with all optional rules and possibly Rétoré transformations, seems more than doable. Indeed, the proof techniques used in Sections 2.3.6.2 and 2.3.6.3 seem to be general enough for the full logic, so that the main difficulty when going to the general case is strong normalization. For this, it suffices first to turn the comprehensive proof sketch of Conjecture 2.45 into a proof. One could then conclude as done here, using Theorem 2.7.

Part II

Proof-nets

A usual representation of a formal proof in most logics, including classical, intuitionistic or linear logic, is a proof from sequent calculus, *i.e.* a tree with as vertices rules. A major contribution from linear logic is another syntax for proofs: *proof-nets*. In this syntax, proofs are no more trees but more general graphs, allowing more flexibility and leading to simpler reasonings that focus on key cases without having to bother with commutative cases. The keystone of the proof-net syntax is that it quotients proofs by rule commutation, which was the source of many problems in Chapter 2. Consequently, cut-elimination is convergent in this framework. Therefore, the proof-net syntax is more adequate than the sequent calculus one when considering proofs up cut-elimination, or equivalently up to rule commutation. Intuitively, this is a similar phenomenon to what happens in intuitionistic logic with natural deduction: a natural or canonical representation is often preferable over others. Nonetheless, proof-nets for full linear logic do not exist at the time this manuscript is written – see Section 4.5 for more details. In this part, we will use the notion of proof-nets from Hughes & van Glabbeek [HG05] for unit-free MALL, with and without the mix_0- and mix_2-rules.

Our main contributions here are the following two. First is a new proof of *sequentialization* for proof-nets, which is the hard direction in proving that these graphs correspond to proofs from sequent calculus: recovering a proof of sequent calculus given a proof-net. As many demonstrations in the literature, ours is based on the search of a *splitting* vertex, corresponding to a last possible rule and allowing to conclude by induction. This problem is strongly linked to more usual graph theoretical results, with recent advances showing even equivalence between sequentialization of MLL^2_{uf} proof-nets and perfect matchings [Ret03; Ngu20]. Indeed, special classes of graphs for which an inductive syntax can be given is common sight in graph theory, which is exactly the subject of sequentialization. Our proof continues in this spirit, for it is based on an intermediate result purely in graph theory. We deduce the existence of a splitting vertex by means of a simple lemma about edge-coloring in graph theory, the *bungee jumping* lemma, leading to a generalization of *Yeo's theorem* [Yeo97]. This approach permits to separate the pure reasonings about colored graphs from the particularities of proof-nets. Besides, it gives a graph theoretical analog to $\mathsf{MALL}^{0,2}_{uf}$ sequentialization, while previous works only consider $\mathsf{MLL}^{0,2}_{uf}$ – even if hypotheses of this analog are quite unusual. Not only our result holds with the mix_0- and mix_2-rules, it is simpler than previous proofs of sequentialization, even when restricted to the more known proof-nets of MLL_{uf}. Our second contribution is a formalization of MLL_{uf} proof-nets in the *Coq* proof assistant, with a definition of these graphs and proofs of some key results including our new proof of sequentialization. This formalization is quite striking, for manipulating graphs on computer is notoriously hard.

This part is thus divided into three. First, Chapter 3 is purely about graph theory. It introduces the graph notions we will need, and its major part is about a new generalization of Yeo's theorem, with its (quite involved) statement and its proof. We also discuss the reach of our generalization, looking at results equivalent to Yeo's theorem as well as another known generalization [Gal+22]. Then, in Chapter 4 are defined proof-nets from [HG05] and links between this new syntax and sequent calculus. In particular, we give a simple proof of sequentialization based on our generalization of Yeo's theorem, which is as far as we know the second proof of sequentialization for $\mathsf{MALL}^{0,2}_{uf}$ proof-nets from [HG05]. Notably, it is the first proof of "bottom-up" sequentialization, *i.e.* building an associated sequent calculus proof from the last rule towards terminal rules – the original proof in [HG05] finds all $\parr$- and &- rules and then files in the other rules. We finish this part with Chapter 5 on the formalization of proof-nets in Coq, explaining which part of the theory has been ported on computer, which crucial choices were made, as well as the encountered difficulties.

Chapter 3

Graph Theory

In the present chapter, we define the graph theoretic notions we will need for introducing proof-nets (paths, …). We also prove a graph theoretical theorem (Theorem 3.19) from which will be deduced a key but often difficult result for proof-nets, the *sequentialization theorem* (Theorem 4.18): a proof-net corresponds to a proof of sequent calculus. This result from graph theory is about edge-colored graphs, and is a generalization of Yeo's theorem [Yeo97]. An *edge-colored graph* is a graph with a function of codomain the edges of the graph, and the elements of whose domain are called *colors*. We say a cycle is *alternating* when all its adjacent edges have different colors. Yeo's theorem states that an edge-colored graph with no alternating cycle has a *splitting* vertex, allowing to decompose the graph in a "nice" way with its edges to a connected component all of the same color – see Figure 3.1 for an intuition, the precise definition is given in Section 3.2.1. This decomposition can be carried on, so as to give an inductive representation of a graph with no alternating cycle. We generalize it in several directions. First, we define a more general notion of edge-coloring than the standard one, this new notion being used in all this chapter. Second, we give more control on the splitting vertex found, thanks to a parameter (which will be a set of edges) and an order for which the splitting vertex given by the theorem is maximal. Lastly but most importantly, instead of outright forbidding alternating cycles we allow some of them respecting precise conditions.

One of the major advantages of having our key result directly in graph theory is to be able to apply it in various settings. For instance, corollaries of our main result include a proof of

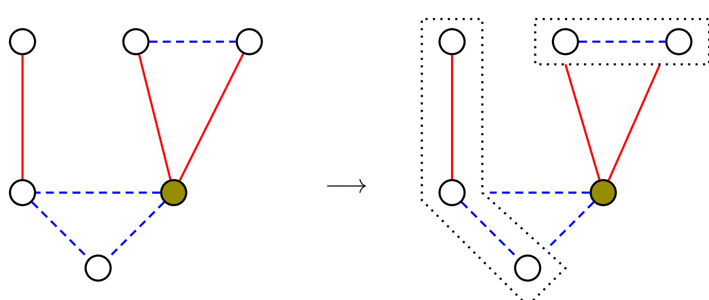


Figure 3.1: Example of Yeo's theorem with a filled splitting vertex

sequentialization for different definitions of $\mathsf{MLL}^{0,2}_{uf}$ proof-nets (as well as for $\mathsf{MELL}^{0,2}$ proof-nets as their sequentialization can be handled similarly), one for $\mathsf{MALL}^{0,2}_{uf}$ proof-nets, as well as sufficient conditions for similar syntaxes – in particular it permits to have a proof of sequentialization for a version where we put jump arcs differently in the definition of proof-net (Section 4.1.1).

It is quite natural to look at what results on proof-nets tell us more generally about graphs, as a proof-net is a particular kind of graph representing a proof. Links between $\mathsf{MLL}^{0,2}_{uf}$ proof-nets and graph theory have been explored for instance in [Ret03; Ngu18a; Ngu20; Ngu18b], showing a correspondence between sequentialization for proof-nets and perfect matchings in graphs. The parallel is intuitively quite simple: sequentializing a proof-net is giving a proof, *i.e.* a sequence of rules, corresponding to it. Meanwhile, some results in graph theory such as Yeo's theorem give a grammar generating all graphs in certain classes. Here, we explore the links between sequentialization for $\mathsf{MALL}^{0,2}_{uf}$ proof-nets and a new notion of edge-coloring for multigraphs.

Outline This chapter is organized as follows. First are given definitions from the theory of graphs: graphs, paths,... (Section 3.1). We then define our notion of coloring, and prove a simple but key result: the *Bungee Jumping lemma* (Section 3.2). The core of this chapter, with our main contribution, is a generalization of Yeo's theorem [Yeo97] (Section 3.3). Finally, some folklore results that will be of use in this thesis are presented (Section 3.4).

3.1 Graphs & Paths

We define here graphs and paths. Compared to what can usually be found in the literature, our graphs are directed partial multigraphs, on which we will consider undirected paths, *i.e.* paths in the underlying undirected total multigraph. We start with a quick reminder of the usual vocabulary on graphs (Section 3.1.1) before presenting our own version (Section 3.1.2). We then define paths on these graphs (Section 3.1.3).

3.1.1 Undirected graphs

In the literature, *e.g.* [Die05], are defined what we call in this thesis **undirected graphs**. We recall here quickly some of the associated terminology to differentiate afterwards with ours. These graphs are made of vertices and arcs between them, and are total undirected multigraphs: each arc is between a pair of vertices (possibly the same one taken twice), these pairs not being ordered; there may be several arcs between the same pair of vertices. An **arc-coloring** is a function with domain the set of arcs of the graph, and an arbitrary codomain, whose elements are called **colors**. Even though such a coloring is usually called an *edge-coloring*, we prefer to reserve this term for another notion of coloring that we will define later. To be particularly clear on this point, when speaking of edge-coloring in this thesis, we exclusively mean a more general notion that the usual one, the latter being renamed arc-coloring.

3.1.2 Graphs

We define here the notions of graphs we will use. Be careful: some notions and names given here do not correspond exactly to those usually used in the literature.

A (directed finite partial multi) **graph** is a 4-tuple $(\mathcal{V}, \mathcal{A}, \mathsf{s}, \mathsf{t})$ where $\mathcal{V}$ is a finite set whose elements we call **vertices**, $\mathcal{A}$ one whose elements are **arcs**, and the **source** s and **target** t functions are *partial* functions from $\mathcal{A}$ to $\mathcal{V}$.

The **endpoints** of an arc a are its source $\mathsf{s}(a)$ and target $\mathsf{t}(a)$ (if defined). An arc is **incident** to a vertex if this vertex is either its source or its target, in other words one of its endpoints. An arc a is **total** if both $\mathsf{s}(a)$ and $\mathsf{t}(a)$ are defined. A **loop arc** is a total arc a such that $\mathsf{s}(a) = \mathsf{t}(a)$.

A graph is **total** if s and t are total functions and **empty** if $\mathcal{V} = \mathcal{A} = \emptyset$. It is **simple** if no two arcs have the same endpoints, in other words there is at most one arc between two given vertices. A graph is **loop-free** if it has no loop arc.

Many notions lift immediately from graphs to partial graphs and moreover from any partial graph, one can recover an **underlying** (directed total multi) graph by restricting $\mathcal{A}$ to the **total arcs**. As an example, one can consider isomorphism of graphs: two graphs $(\mathcal{V}, \mathcal{A}, \mathsf{s}, \mathsf{t})$ and $(\mathcal{V}', \mathcal{A}', \mathsf{s}', \mathsf{t}')$ are **isomorphic** if there are bijections f and g respectively from $\mathcal{V}$ to $\mathcal{V}'$ and from $\mathcal{A}$ to $\mathcal{A}'$ such that for any arc $a \in \mathcal{A}$, $\mathsf{s}(a)$ (resp. $\mathsf{t}(a)$) is defined if and only if $\mathsf{s}'(g(a))$ (resp. $\mathsf{t}'(g(a))$) is and in this case $\mathsf{s}'(g(a)) = f(\mathsf{s}(a))$ (resp. $\mathsf{t}'(g(a)) = f(\mathsf{t}(a))$).

An **edge** is an arc a together with a **direction** in $\{+, -\}$; we use the notations a^+ and a^- for these edges, and call a the **support** of the edge. We note $\mathcal{E}$ the set of edges of the graph. The functions source and target are extended to edges by $\mathsf{s}(a^+) = \mathsf{s}(a) = \mathsf{t}(a^-)$ and $\mathsf{t}(a^+) = \mathsf{t}(a) = \mathsf{s}(a^-)$ – which can be defined or not. As for arcs, we call endpoints the source and target of an edge. If ε is a direction, $\overline{\varepsilon}$ is the opposite one. If e is an edge a^ε then its **reverse** $\overline{e}$ is $a^{\overline{\varepsilon}}$. We say e is total when a is. Two edges e_1 and e_2 are **composable** if the target of e_1 is defined and equal to the source of e_2. A **loop edge** is an edge supported by a loop arc; equivalently, it is an edge that is composable with itself.

Given a graph $G = (\mathcal{V}, \mathcal{A}, \mathsf{s}, \mathsf{t})$, a **sub-graph** of G is a graph $G' = (\mathcal{V}', \mathcal{A}', \mathsf{s}', \mathsf{t}')$ such that $\mathcal{V}' \subseteq \mathcal{V}$, $\mathcal{A}' \subseteq \mathcal{A}$ and for any $a \in \mathcal{A}'$, if $\mathsf{s}(a)$ (resp. $\mathsf{t}(a)$) is defined and belongs to $\mathcal{V}'$ then $\mathsf{s}'(a) = \mathsf{s}(a)$ (resp. $\mathsf{t}'(a) = \mathsf{t}(a)$), and otherwise $\mathsf{s}'(a)$ (resp. $\mathsf{t}'(a)$) is undefined. An **induced sub-graph** is a sub-graph whose set of arcs $\mathcal{A}'$ is made exactly of arcs whose both endpoints are in its set of vertices $\mathcal{V}'$. A sub-graph is **proper** if it is not the graph itself, nor the empty graph. Given sub-graphs S and R, their **union** $S \cup R$ (resp. their **intersection** $S \cap R$) is the sub-graph whose sets of vertices and of arcs are the union (resp. intersection) of those of S and R.

There are two natural notions of removing a vertex v in a graph G. The graph $G \setminus v$ is obtained by removing v from the vertices of G. As a consequence arcs stay the same but the source and target functions become less defined: arcs incident to v lose a source or target. Meanwhile, $G - v$ also has for vertices those of G except v but erases all edges incident to v. Remark that if G is total, then $G - v$ also is, but not necessarily $G \setminus v$.

The cardinal of a set S will be denoted $\#|S|$.

Definition 3.1 (Degree)**.** The **in-degree** $deg^{in}(v)$ of a vertex v is the number of *arcs* of target v: $deg^{in}(v) = \#|\{a \in \mathcal{A} \mid \mathsf{t}(a) = v\}|$. Similarly, the **out-degree** $deg^{out}(v)$ of a vertex v is the number of *arcs* of source v: $deg^{out}(v) = \#|\{a \in \mathcal{A} \mid \mathsf{s}(a) = v\}|$.

We set **the degree** $deg(v)$ of a vertex v as $deg(v) = deg^{in}(v) + deg^{out}(v)$.

The **maximal degree** $\Delta(G)$ of a graph G is the maximum of the degrees of its vertices: $\Delta(G) = \max_{v \in \mathcal{V}} deg(v)$.

3.1.3 Paths

A **path** p is a pair $(v, \vec{e})$ of a vertex v and a finite sequence $\vec{e}$ of total and consecutively composable edges such that $\vec{e}$ is empty – in which case p is an **empty** path – or v is the source of the first edge of $\vec{e}$. The vertex v is the **source** of p and the **target** of p is the target of the last edge in $\vec{e}$, if $\vec{e}$ is not empty, and v otherwise. Since we require all edges of a path to be total, we are in fact considering undirected paths (since arcs can be crossed in both directions) in the underlying total graph. Again, the source and target of a path are its endpoints.

By considering v followed by the targets of the edges in $\vec{e}$, a path p induces a *non-empty* sequence of vertices. A vertex u **is a vertex of** p if it belongs to this sequence. Since a given vertex may occur more than once in this sequence, we may have to talk about **occurrences of vertices in a path** to distinguish these equal values. An occurrence of a vertex in a path is **internal** if it is neither the source nor the target occurrence. An **arc of a path** is the support of one of its edges.

Two paths $p_1 = (v_1, \vec{e}_1)$ and $p_2 = (v_2, \vec{e}_2)$ are **composable** if $\mathsf{t}(p_1) = \mathsf{s}(p_2)$ – recall that both are always defined – and, in that case, we define their **concatenation** $p_1 \cdot p_2$ as $(v_1, \vec{e}_1 \cdot \vec{e}_2)$. The **reverse** $\overline{p}$ of a path p is obtained by reversing the order of edges and taking the reverse of each edge. Moreover its source is the target of p, and conversely.

A path q is a **sub-path** of a path p if its edges are a contiguous sub-sequence of the edges of p, or q is an empty path and its source is a vertex of p. Equivalently, q is a sub-path of p if there exist two paths q_1 and q_2 such that $p = q_1 \cdot q \cdot q_2$. In the same spirit, q is a **prefix** of p if q_1 is empty, and a **suffix** of p if q_2 is empty. If v_1 and v_2 are two occurrences of vertices of a path p, with v_1 occurring before v_2, there is a unique sub-path of p with source v_1 and target v_2, that we denote by $p_{v_1 \to v_2}$.

A path is **arc-simple** if its arcs are pairwise distinct. A path is **vertex-simple**, that we will simply call **simple**, if it is arc-simple and its vertices are pairwise distinct except possibly its endpoints which may be equal. A path is **bouncing** if it contains two consecutive equal arcs; equivalently, a path is bouncing if it contains a sub-path of the shape $e \cdot \overline{e}$, or $e \cdot e$ – in the latter case, e must be a loop edge.

A path is **closed** if it has equal endpoints, otherwise it is **open**. A **cycle** is a non-empty simple closed path. A graph is **acyclic** if it has no cycle.

Our notion of simple path corresponds to the one usually used in the literature. Nonetheless, proving that the concatenation of two simple paths is simple is harder than doing the same for arc-simple paths.

Fact 3.2 (Concatenation of arc-simple paths)**.** *If p_1 and p_2 are two arc-simple paths such that $\mathsf{t}(p_1) = \mathsf{s}(p_2)$ and the arcs of p_1 are disjoint from those of p_2, then $p_1 \cdot p_2$ is arc-simple.*

Lemma 3.3. *A non-bouncing path with pairwise distinct vertices except possibly its endpoints is simple.*

Proof. We have to prove that we cannot have equal arcs in the path. If so they cannot be consecutive, but in this case their endpoints provide two disjoint pairs of occurrences of equal vertices in the path which contradicts the fact that only the endpoints may be equal. □

Lemma 3.4 (Concatenation of simple paths)**.** *If p_1 and p_2 are two simple open paths and their unique common vertices are the target of p_1 and the source of p_2, and possibly the target of p_2 and the source of p_1, and if the last arc of p_1 is different from the first arc of p_2, then $p_1 \cdot p_2$ is simple.*

Proof. If two vertices of $p_1 \cdot p_2$ are equal, they must be its endpoints: if one of them is v_1 in p_1 for example (and not the source of p_1), it is not equal to another vertex in p_1 as p_1 is simple and open, and it is not equal to a vertex v_2 in p_2 as the only possibility would be the source of p_2 and in this case the occurrences v_1 and v_2 are the same (identified in the concatenation). Now, by Lemma 3.3, it is enough to prove that $p_1 \cdot p_2$ is not bouncing. As the two paths are simple, the only possibility would be if the last arc of p_1 is equal to the first arc of p_2, a contradiction. □

Lemma 3.5 (Concatenation of disjoint simple paths)**.** *If p_1 and p_2 are two simple open or empty paths with $\mathsf{t}(p_1) = \mathsf{s}(p_2)$ being their unique common vertex then $p_1 \cdot p_2$ is simple and open or empty.*

Proof. If p_1 or p_2 is empty, the result is immediate. Otherwise one can apply Lemma 3.4 since the source of the last edge of p_1 is not in p_2. Moreover $p_1 \cdot p_2$ is open since the source of p_1 does not belong to p_2. □

Connectivity is not immediate to define in partial graphs because paths contain total edges only. Two vertices v and u are **connected** if there exists a path with source v and target u. Two arcs are **connected** if they are equal[1] or incident to two connected vertices. Two edges are **connected** if their supports are. A (sub-)graph is **connected** if all its pairs of arcs are. A **connected component** is a largest sub-set of $\mathcal{V}$ such that for any pair of vertices inside are connected.

3.2 Edge-coloring & Bungee Jumping

We define here our notion of coloring, named *(local) edge-coloring*. It is a generalization of the arc-coloring from the literature (*i.e.* the usual edge-coloring) that, at the best of the author's knowledge, is a novel concept. In this section is also proved a simple but useful result: the Bungee Jumping lemma.

3.2.1 Edge-coloring, Bridge & Splitting vertex

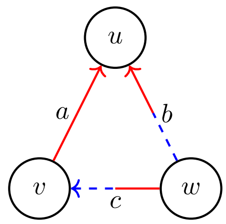
An **edge-coloring** of a graph G is any function c with as domain the set of *edges* $\mathcal{E}$ of G. We can see an edge-coloring equivalently as a coloring of half-arcs: by convention, we consider that $\mathsf{c}(a^+)$ is the color of the target half-arc of a, and $\mathsf{c}(a^-)$ is the color of the source half-arc of a – without presuming that a source or target vertex is defined. We recover the standard notion of arc-coloring when $\mathsf{c}(e) = \mathsf{c}(\overline{e})$ for every edge e. An example of edge-colored graph is given on Figure 3.2. On this drawing, as for all other illustrations, we represent an edge-coloring as an "half-arc coloring" for it is the intuitive way to understand it. Nonetheless, we will always speak of edge-coloring as we mainly consider (undirected) paths, and edges occur naturally in those contrary to half-arcs.

A **bridge** of **pier** v is a pair of (possibly equal) edges (e, f) such that $\mathsf{t}(e) = \mathsf{s}(f) = v$, $e \neq \overline{f}$ and $\mathsf{c}(e) = \mathsf{c}(\overline{f})$ – which is called the **color of the bridge**. Equivalently, $(e, \overline{f})$ is a bridge of pier v if and only if $\mathsf{t}(e) = \mathsf{t}(f) = v$, $e \neq f$ and $\mathsf{c}(e) = \mathsf{c}(f)$. Note that (e, f) forms a bridge if and only if the reverse $(\overline{f}, \overline{e})$ does, and they share the same pier and color. The only bridges of the edge-colored graph in Figure 3.2 are (a^+, b^-) and its reverse (b^+, a^-).

Fact 3.6. *Let v be a vertex and e_1, e_2, e_3 three edges of target v with $e_1 \neq e_2$.*

- *If $(e_1, \overline{e_2})$ is not a bridge, then at least one of $(e_3, \overline{e_1})$ and $(e_3, \overline{e_2})$ is not a bridge.*

[1] This first condition is necessary for arcs with undefined source and target.



Edge-coloring c:

$\mathsf{c}(a^+) =$ solid $\qquad \mathsf{c}(a^-) =$ solid

$\mathsf{c}(b^+) =$ solid $\qquad \mathsf{c}(b^-) =$ dashed

$\mathsf{c}(c^+) =$ dashed $\qquad \mathsf{c}(c^-) =$ solid

Figure 3.2: Example of edge-colored graph

- *If both* $(e_1, \overline{e_3})$ *and* $(e_3, \overline{e_2})$ *are bridges, then* $(e_1, \overline{e_2})$ *is one as well.*

Proof.

- We have $\mathsf{c}(e_1) \neq \mathsf{c}(e_2)$, thus we cannot have both $\mathsf{c}(e_3) = \mathsf{c}(e_1)$ and $\mathsf{c}(e_3) = \mathsf{c}(e_2)$.
- By contrapositive, if $(e_1, \overline{e_2})$ is not a bridge, then applying the previous item we would get that either $(e_3, \overline{e_1})$ or $(e_3, \overline{e_2})$ is not a bridge. $\square$

An **internal bridge of a path** p is (an occurrence of) a length-2-sub-path $e \cdot f$ of p, such that (e, f) forms a bridge. More generally, a **bridge of a path** p is either an internal bridge of p, or the pair (e, f) with e the last edge of p and f the first edge of p, in case this pair forms a bridge – and then p is a closed path, and possibly a loop when $e = f$. Remark that the reverse of a path contains the same number of bridges as this path, as well as the same piers. A **bridge-free path** (resp. **internal-bridge-free path**) is one without bridge (resp. internal bridge). Note that in the case of a cycle reduced to a loop, say e with $\mathsf{s}(e) = \mathsf{t}(e)$, this loop is bridge-free if and only if $\mathsf{c}(e) \neq \mathsf{c}(\overline{e})$, and a loop is always internal-bridge-free since it has no length-2-sub-path.

We say that e is a **bridge arch** when e is not a loop edge and (e, f) is a bridge for some non-loop edge f which has a different support than e – in other words, when $e \cdot \overline{f}$ is a simple internal-bridge-free path.

We call **splitting** a vertex v such that any cycle containing it has a bridge of pier v. We will show in Section 3.3.5.1 that this notion of splitting vertex fits the usual notion at play in the conclusion of Yeo's theorem [Yeo97] (illustrated on Figure 3.1).

Remark 3.7. This notion of edge-coloring does not seem to be standard in the literature. Note that, when used only through the notions of bridges and splitting vertices, the fact that a given color is given for edges with different targets has no impact. This means that we could use different sets of colors depending on each target vertex. Plus, this would be equivalent to defining equivalence relations on the sets of edges with the same target. We keep the idea of edge-coloring for it comes as a direct generalization of arc-coloring.

3.2.2 Bungee Jumping

We prove here our key intermediate result, and two of its corollaries we use in the proof of our main result. The idea of this lemma is to show that some notion of backtracking cannot happen, ensuring some kind of progress. Typically, it will be used to prove that, in a graph with no bridge-free cycle, if one leaves a cycle with a minimal number of bridges from one of its bridge pier using a bridge-free path, then it is impossible for this path to cross the cycle it started from. As it shows that when leaving a bridge, it is not possible to crash back on this bridge, we call it *Bungee Jumping*.

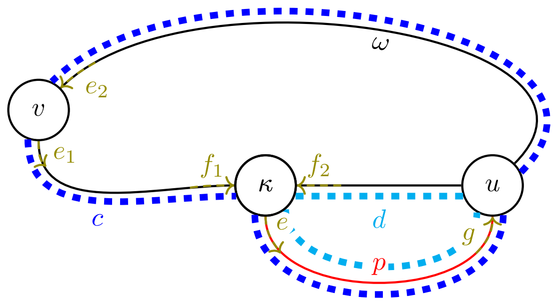


Figure 3.3: Illustration of Lemma 3.8

Lemma 3.8 (Bungee Jumping)**.** *Fix an edge-coloring* c *of a graph* G *and* v *a vertex of* G*. Assume* ω *is a cycle starting from* v *which contains an internal bridge* $(f_1, \overline{f_2})$ *of pier* κ *and of color* $a = \mathsf{c}(f_1) = \mathsf{c}(f_2)$*. Also suppose there exists a simple open bridge-free path* p *starting from* κ *with an edge* e *such that* $\mathsf{c}(\overline{e}) \neq a$*, ending on a vertex* u *of* ω*, and whose vertices in common with* ω *are exactly its endpoints* κ *and* u*. Then there exists a bridge-free cycle containing* p *or there exists a cycle* c *with source* v *and strictly less bridges than* ω*, in which case* v *is a bridge pier of* c *only if it is a bridge pier of* ω*.*

Proof. Use Figure 3.3 as a reference for notations. We denote by v_1 the occurrence of v at the source of ω, by v_2 its occurrence at the target of ω. We also set e_1 the first edge of ω, e_2 its last edge, and g the last edge of p.

By symmetry (considering the reverse of ω if necessary), we can assume that u is in $\omega_{\kappa\to v_2}$ and if $u = v_2$ then (g, e_1) is a bridge only if (e_2, e_1) is a bridge. Indeed, if $u \notin \omega_{\kappa\to v_2}$, we reverse ω. And if $u = v_2$ and (g, e_1) is a bridge but (e_2, e_1) is not a bridge, then necessarily $(g, \overline{e_2})$ is not a bridge (Fact 3.6) and we reverse ω.

Consider the closed path $c = \omega_{v_1\to\kappa} \cdot p \cdot \omega_{u\to v_2}$ (see Figure 3.3): c is non-empty (because p is non-empty) and simple (because $\omega_{u\to v_2} \cdot \omega_{v_1\to\kappa} \cdot p$ is simple by Lemma 3.4), hence it is a cycle. Moreover v is bridge pier of c only if it is a bridge pier of ω.

Let us now count the number of bridges in our two cycles. Since p is bridge-free and $\mathsf{c}(\overline{e}) \neq \mathsf{c}(f_1)$, c contains $b^c_v + n_1 + b^c_u + n_3 - \delta_{v,u} b^c_v$ bridges, while the number of bridges in ω is $b^\omega_v + n_1 + 1 + n_2 + b^\omega_u + n_3 - \delta_{v,u} b^\omega_v$, where:

- b^c_v (resp. b^c_u) is 1 if v (resp. u) is a bridge pier of c and 0 otherwise;
- b^ω_v (resp. b^ω_u) is 1 if v (resp. u) is a bridge pier of ω and 0 otherwise;
- n_1 (resp. n_2, n_3) is the number of bridges of $\omega_{v_1\to\kappa}$ (resp. $\omega_{\kappa\to u}$, $\omega_{u\to v_2}$);
- $\delta_{v,u}$ is 1 if $v = u$ and 0 otherwise.

Comparing these two numbers, the number of bridges of ω minus the number of bridges of c is $(b^\omega_v - b^c_v)(1 - \delta_{v,u}) + 1 + n_2 + b^\omega_u - b^c_u$.

If c has strictly less bridges than ω, meaning the above quantity is positive, then we are done. Otherwise, as by hypotheses $0 \leq b_v^c \leq b_v^\omega \leq 1$, we must have $b_u^c \geq 1 + n_2 + b_u^\omega$. This is only possible if $b_u^c = 1$, $n_2 = 0$ and $b_u^\omega = 0$. In this case, we consider the closed path $d = p \cdot \overline{\omega_{\kappa \to u}}$. It is bridge-free since $\mathsf{c}(\overline{e}) \neq \mathsf{c}(f_2)$ and u is not a bridge pier of d – by $b_u^c = 1$ and $b_u^\omega = 0$, using Fact 3.6. Moreover d is simple by Lemma 3.4 and thus is a bridge-free cycle containing p. □

Remark 3.9. The proof of Lemma 3.8 is obviously constructive: changing if necessary ω's orientation, the cycles we are looking for are $p \cdot \overline{\omega_{\kappa \to u}}$ and $\omega_{v_1 \to \kappa} \cdot p \cdot \omega_{u \to v_2}$.

Definition 3.10. For a vertex v, we denote by $\mathcal{M}_v$ the set of cycles:

- with source (and target) v;
- whose last and first edges do not make a bridge of pier v;
- with a minimal number of bridges among all cycles respecting the previous two conditions.

The set $\mathcal{M}_v$ is empty if and only if v is splitting.

Corollary 3.11 (Small Bungee Jumping)**.** *Fix an edge-coloring* c *of a graph* G*,* v *a vertex of* G*, and* $\omega \in \mathcal{M}_v$*. Assume* ω *contains a bridge* $(f_1, \overline{f_2})$ *of pier* κ *and of color* $a = \mathsf{c}(f_1) = \mathsf{c}(f_2)$*, and there is a simple open bridge-free path* q *starting from* κ *with an edge* e *such that* $\mathsf{c}(\overline{e}) \neq a$*, and ending on a vertex* u *of* ω*. Then there exists a (non-loop) bridge-free cycle containing* κ*.*

Proof. By taking a prefix of q if necessary, we can assume that q does not share any vertex with ω other than its (distinct) endpoints κ and u. We apply Lemma 3.8 to ω and q. Since $\omega \in \mathcal{M}_v$, we cannot find a cycle with source v, no bridge at v and strictly less bridges than ω. We thus have a bridge-free cycle containing e, so a non-loop one containing κ. □

Corollary 3.12 (Big Bungee Jumping)**.** *Let* G *be a graph with an edge-coloring* c*. Assume* ω *is a cycle with source* v*, and* x *and* y *are two internal vertices of* ω *with* x *occurring before (or equal to)* y*, and with at least one pier between* x *and* y *(possibly* x *or* y*). Let* f_x *be the edge of* ω *with target* x *and* f_y *be the reverse of the edge of* ω *with source* y*. Suppose* l *is a vertex such that we have a path* ρ *from* x *to* l*, a path* χ *from* y *to* l *and an edge* e *with source* l *such that:*

- $\rho \cdot e$ *and* $\chi \cdot e$ *are simple paths;*
- $f_x \cdot \rho \cdot e$ *and* $f_y \cdot \chi \cdot e$ *are internal-bridge-free paths;*
- *the only vertices of* ρ *or* χ *which may belong to* $\omega_{y \to v} \cdot \omega_{v \to x}$ *are* x *and* y *(which might be equal to* l *when one of these paths is empty).*

If there is a path p *such that:*

- $e \cdot p$ *is a simple open bridge-free path;*
- *its target* u *belongs to* $\omega_{y \to v} \cdot \omega_{v \to x}$*;*
- *both* ρ *and* χ *have no vertex in common with* p*;*

then there exists a bridge-free cycle containing e *or there exists a cycle* c *with source* v *and strictly less bridges than* ω*, in which case* v *is a bridge pier of* c *only if it is a bridge pier of* ω*.*

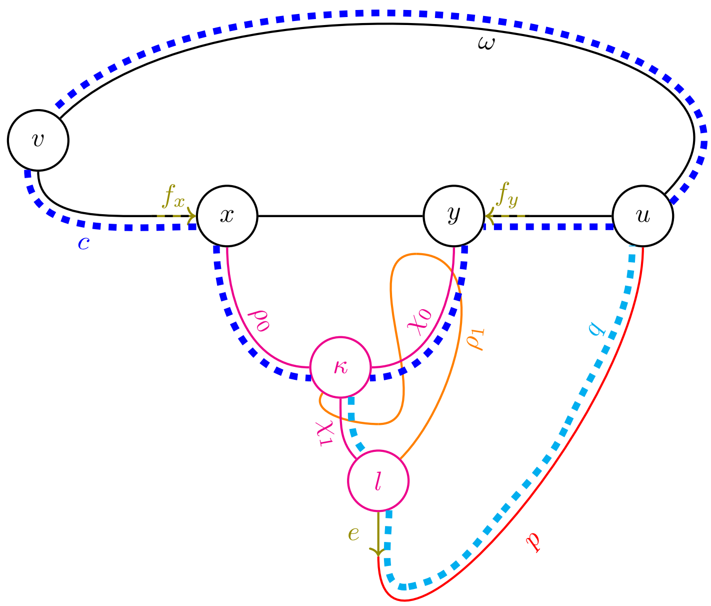


Figure 3.4: Illustration of Corollary 3.12

Proof. Use Figure 3.4 as a reference for notations. By taking a prefix of p if necessary, we can assume that p does not share any vertex with $\omega_{y\to v} \cdot \omega_{v\to x}$ other its target u (in case $\mathsf{t}(e)$ is in $\omega_{y\to v} \cdot \omega_{v\to x}$, p is the empty path). Consider κ the first vertex of ρ which belongs to χ. We have $\rho = \rho_0 \cdot \rho_1$ and $\chi = \chi_0 \cdot \chi_1$ with κ as target of ρ_0 and χ_0. We consider the closed path $c = \omega_{v\to x} \cdot \rho_0 \cdot \overline{\chi_0} \cdot \omega_{y\to v}$ which starts with v and has v as bridge pier only if it is also the case for ω. The path c is simple since $(\omega_{y\to v} \cdot \omega_{v\to x}) \cdot (\rho_0 \cdot \overline{\chi_0})$ is, by Lemmas 3.4 and 3.5, thus it is a cycle. Remark that, by hypotheses, the only bridge of c which is not a bridge of ω, is a possible bridge of pier κ. If κ is not a bridge pier of c then c has strictly less bridges than ω, the latter containing a pier between x and y. Otherwise c contains a bridge of pier κ (and at most as many bridges as ω) and we apply Lemma 3.8 on this cycle with the path $q = \chi_1 \cdot e \cdot p$.

We have already seen that c satisfies the hypotheses of Lemma 3.8. Now concerning $\chi_1 \cdot e \cdot p$: it is a simple path by Lemma 3.5 since $e \cdot p$ is simple and χ has no vertex in common with p, and it is bridge-free since $e \cdot p$ is bridge-free and $f_2 \cdot \chi \cdot e$ has no internal bridge. Moreover, since κ is a bridge pier of c and $f_y \cdot \chi \cdot e$ is bridge-free, the color of the reverse of the first edge of $\chi_1 \cdot e \cdot p$ cannot be the color of the bridge at κ in c. Finally, $\chi_1 \cdot e \cdot p$ shares only its endpoints with c. Indeed, the only vertex of χ_1 in $\omega_{y\to v} \cdot \omega_{v\to x}$ is possibly its source[2] and χ_1 is disjoint from ρ_0 by definition of κ; meanwhile p has only its target u inside $\omega_{y\to v} \cdot \omega_{v\to x}$, and share no vertex with ρ and χ, hence with their respective prefixes ρ_0 and χ_0.

Thus, Lemma 3.8 yields a wished cycle containing v, or a bridge-free cycle containing $\chi_1 \cdot e \cdot p$, and in particular containing e. ☐

3.3 A generalization of Yeo's theorem

We give here our main contribution for this chapter: a generalization of Yeo's theorem [Yeo97]. Yeo's theorem states that an arc-colored graph with no bridge-free cycle has a splitting vertex. We generalize it in the context of edge-coloring and by weakening its hypothesis through allowing some bridge-free cycles, as well as by giving more control on which vertex is found splitting: the found vertex will be maximal for a particular ordering. This last property will be particularly useful because we also add to this theorem a parameter E, and the given splitting vertex will be the target of one of these edges. When applying our theorem to the theory of proof-nets, we recover different proofs of sequentialization from the literature (namely all proofs by splitting vertex) by adjusting the value of this parameter.

The allowed values for this parameter depends on the order from which the found splitting vertex will be maximal: this order is defined in Section 3.3.1. Once this relation given, we can state our main result in Section 3.3.2, and prove it in Section 3.3.3. As the hypotheses of this theorem are quite involved, we show using counter-examples in Section 3.3.4 that they are all needed. Afterwards, in Section 3.3.5 are given corollaries of our result, among which Yeo's theorem itself as well as one of its generalization to H-coloring; this shows our notion of edge-coloring is useful even outside the sequentialization of proof-nets. Then, we consider in Section 3.3.6 theorems equivalent to Yeo's theorem: we show that Yeo's theorem on edge-colored graph is equivalent to its original version from [Yeo97] on arc-coloring, but prove that a series of theorems known to be equivalent to it can all be easily deduced from our edge-colored version *without the need of an encoding modifying vertices or arcs.* The last section, Section 3.3.7, describes the well-known use of

[2] As the only vertices of χ in $\omega_{y\to v} \cdot \omega_{v\to x}$ are its source y and possibly x, if the latter is a vertex of χ then $x = \kappa = \mathsf{s}(\chi_1)$.

Yeo's theorem to obtain an inductive definition of graphs with no bridge-free cycle, as well as links with proof-nets, the objects of the next chapter.

3.3.1 Order on edges

Here, we fix a graph G with an edge-coloring c. We define two relations on edges, that will be of use in all this section about Yeo's theorem.

Definition 3.13 ($\Rsh$, $\lhd$)**.** Given an edge e and a vertex v, we write $e \overset{p}{\Rsh} v$ if p is a simple, open, bridge-free path from $\mathsf{t}(e)$ to v such that the reverse of its first edge is not colored $\mathsf{c}(e)$. We moreover write $e \overset{p}{\Rsh} f$ when the last edge of p is f. We simply write $e \Rsh v$ (resp. $e \Rsh f$) whenever such a path exists.

We then write $e \overset{p}{\lhd} f$ when $e \overset{p}{\Rsh} f$ and there is no vertex u of p such that $f \Rsh u$. Again, we may simply write $e \lhd f$ when such a path exists.

Remark 3.14. If $e \overset{p}{\Rsh} u$, then the first edge e' of p not only satisfies $\mathsf{c}(\overline{e'}) \neq \mathsf{c}(e)$ (thus $e \neq \overline{e'}$ and (e, e') is not a bridge), but e' has not the same support as e. Indeed, if it were the case we would have $e' = e$ being a loop; contradiction as p is simple and open. Also remark that asking the reverse of the first edge to not be colored $\mathsf{c}(e)$ is stronger than asking the first edge not to make a bridge of pier $\mathsf{t}(e)$ with e, for $(e, \overline{e})$ is not a bridge.

Lemma 3.15. *Let e, f and g be edges, v a vertex and p and q paths. If $e \overset{p}{\lhd} f$ and $f \overset{q}{\Rsh} v$ (resp. $f \overset{q}{\Rsh} g$) then $e \overset{p\cdot q}{\Rsh} v$ (resp. $e \overset{p\cdot q}{\Rsh} g$).*

Proof. Assume $e \overset{p}{\lhd} f \overset{q}{\Rsh} v$, and consider the path $p \cdot q$, with source $\mathsf{t}(e)$ and target v. The reverse of its first edge is not colored $\mathsf{c}(e)$, and it is bridge-free as f and the first edge of q do not make a bridge of pier $\mathsf{t}(f) = \mathsf{t}(p) = \mathsf{s}(q)$. To have $e \overset{p\cdot q}{\Rsh} v$, it remains only to show that $p \cdot q$ is open and simple.

Let q' be any prefix of q such that $u = \mathsf{t}(q')$ occurs in p: q' is simple and bridge-free by construction, and the reverse of its first edge, if any, is not colored $\mathsf{c}(f)$. If q' is not empty, then it is open because q is open and simple: it follows that $f \overset{q'}{\Rsh} u$, contradicting $e \overset{p}{\lhd} f$. The only occurrence of a vertex of p in q is thus at its source, hence $p \cdot q$ is simple (by Lemma 3.5) and open.

If we moreover assume that the last edge of q is g, then this also holds for $p \cdot q$, and we obtain $e \overset{p\cdot q}{\Rsh} g$. $\square$

Lemma 3.16. *The relation $\lhd$ is a strict partial order on edges.*

Proof. The relation $\lhd$ is irreflexive: $e \overset{p}{\Rsh} e$ implies $\mathsf{s}(p) = \mathsf{t}(e) = \mathsf{t}(p)$ with p open, a contradiction.

Now assume $e \overset{p}{\lhd} f \overset{q}{\lhd} g$. We obtain $e \overset{p\cdot q}{\Rsh} g$ by Lemma 3.15. To get $e \overset{p\cdot q}{\lhd} g$, it only remains to show that if $g \overset{\rho}{\Rsh} v$ then v does not occur in $p \cdot q$. First observe that v cannot occur in q as $f \overset{q}{\lhd} g$. And, by Lemma 3.15, $f \overset{q\cdot\rho}{\Rsh} v$, so $e \overset{p}{\lhd} f$ implies that v does not occur in p either. $\square$

Remark 3.17. If $e \lhd f$ then f is total and is not a loop.

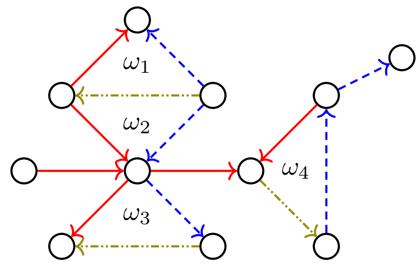


Figure 3.5: Graph with $\mathfrak{D} = \{\Omega_1; \Omega_2\}$ where $\Omega_1 = \omega_1 \cup \omega_2 \cup \omega_3$ and $\Omega_2 = \omega_4$.

We say a set of edges F **dominates** a set of edges E if for all $e \in E$, one has $e \in F$ or $\exists f \in F$, $e \lhd f$. Notice that if $E \subseteq F$, then F dominates E. Moreover, if F dominates the maximal elements of E, then it dominates E.

3.3.2 Parameterized Local Yeo with Cycles

Definition 3.18. Given a graph G with an edge-coloring c, we note $\mathfrak{D}$ the set of non-empty maximal (for the inclusion) connected unions of bridge-free cycles of G.

As an example, see Figure 3.5 with an edge-colored graph and the value of $\mathfrak{D}$ in its caption.

Consider the following generalization of Yeo's theorem, which will be the basis of our proof of sequentialization.

Theorem 3.19 (Parameterized Local Yeo with Cycles)**.** *Take G a graph with an edge-coloring c, and a function $\mathfrak{e}$ which for every $\Omega \in \mathfrak{D}$ gives an edge $\mathfrak{e}(\Omega)$ such that:*

(H^0_G) *for every $\Omega \in \mathfrak{D}$, $\mathsf{s}(\mathfrak{e}(\Omega)) \in \Omega$ and $\mathsf{t}(\mathfrak{e}(\Omega)) \notin \Omega$;*

(H^1_G) *for every $\Omega \in \mathfrak{D}$ and for all edges f such that $\mathsf{t}(f) = \mathsf{s}(\mathfrak{e}(\Omega))$ and $f \neq \overline{\mathfrak{e}(\Omega)}$, we have $\mathsf{c}(f) \neq \mathsf{c}(\overline{\mathfrak{e}(\Omega)})$.*

Pose E a set of edges dominating $E_{in} = \{e \in \mathcal{E} \mid e \text{ is a bridge arch}\} \cup \{\mathfrak{e}(\Omega) \mid \Omega \in \mathfrak{D}\}$ but disjoint *with $E_{out} = \{\overline{\mathfrak{e}(\Omega)} \mid \Omega \in \mathfrak{D}\}$. The target (if it is defined) of any element of E maximal for $\lhd$ (*i.e.* for $\lhd$ restricted to E) is a splitting vertex.*

The parameter gives some control over which splitting vertex can be found, allowing in the context of proof-nets to obtain both a proof of sequentialization by splitting negative vertex and one by splitting terminal vertex (see Section 4.3.6). For some visual intuition, consider the illustration on Figure 3.6: the key point is that while maximal elements of E_{in} are splitting, they may not be maximal in E, and edges above them have splitting targets.

The requirements on E of Theorem 3.19 may be contradictory, in particular if for $\Omega \neq \Omega' \in \mathfrak{D}$, it holds that $\mathfrak{e}(\Omega) = \overline{\mathfrak{e}(\Omega')}$. Notwithstanding this case, the conditions can be satisfied.

Lemma 3.20. *Let G be a graph respecting (H^1_G) as well as:*

(H^2_G) *for $\Omega \neq \Omega' \in \mathfrak{D}$, $\mathfrak{e}(\Omega) \neq \overline{\mathfrak{e}(\Omega')}$.*

Then E_{in} and E_{out} are disjoint.

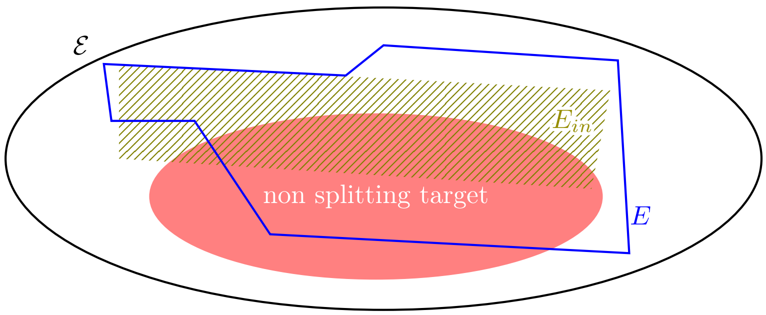


Figure 3.6: Illustration of the interest of the parameter E in Theorem 3.19; maximum elements for $\lhd$ are on top

Proof. By (H^1_G), any $\overline{\mathfrak{e}(\Omega)}$ for $\Omega \in \mathfrak{D}$ cannot be a bridge arch. Moreover, for any $\Omega \in \mathfrak{D}$, $\mathfrak{e}(\Omega)$ cannot be some $\overline{\mathfrak{e}(\Omega')}$ for a $\Omega' \in \mathfrak{D}$. Indeed, it is not the case for $\Omega' = \Omega$ (an edge is never equal to its reverse), and if $\Omega' \neq \Omega$ then by (H^2_G) $\mathfrak{e}(\Omega) \neq \overline{\mathfrak{e}(\Omega')}$. $\square$

3.3.3 Proof of Parameterized Local Yeo with Cycles

The idea behind our proof of Theorem 3.19 is quite simple: an edge of target a non-splitting vertex cannot be maximal for $\lhd$, even when restricted to E, excepted if this edge is in E_{out}. To show this, we apply the Bungee Jumping lemma through its two corollaries when the edge we consider is not in a bridge-free cycle; otherwise, we demonstrate some $\mathfrak{e}(\Omega)$ is greater.

To this end, we first need a study on properties of connected unions of bridge-free cycles, that are the central objects here (Section 3.3.3.1). Once this study done, finding a splitting vertex is easy (Section 3.3.3.2).

3.3.3.1 $\Rsh$-connectivity

Definition 3.21 ($\Rsh$-connectivity)**.** A sub-graph S is said to be **$\Rsh$-connected** if for all v and u distinct vertices in S, for any edge e of target v (e not necessarily in the sub-graph S), there exists inside S a path p such that $e \overset{p}{\Rsh} u$.

The goal of this section is proving Corollary 3.24: connected unions of bridge-free cycles are $\Rsh$-connected.

Lemma 3.22. *A bridge-free cycle is $\Rsh$-connected.*

Proof. Take a bridge-free cycle ω, $v \neq u$ vertices in ω and e an edge of target v. As ω is a cycle containing v, and not a loop for $v \neq u$, it contains two different edges f and g, respectively with target v and with source v. Then (f, g) is not a bridge, for ω is bridge-free. Thus, one of f and $\overline{g}$ does not make a bridge of pier v with $\overline{e}$ (Fact 3.6). Say $(e, \overline{f})$ is not a bridge (up to swapping f and $\overline{g}$). Call p the (maybe reversed) sub-path of ω from v to u starting with f: $e \overset{p}{\Rsh} u$ as ω is a non-loop bridge-free cycle and by hypothesis on f. $\square$

Lemma 3.23. *Let S and R be $\Rsh$-connected sub-graphs having at least one vertex in common. Then $S \cup R$ is $\Rsh$-connected.*

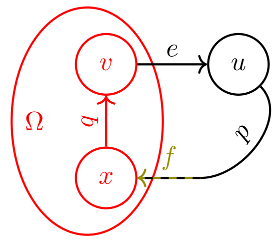


Figure 3.7: Illustration of the proof of Lemma 3.25

Proof. Take vertices $v \neq u \in S \cup R$ and e an edge of target v. If $v, u \in S$ (resp. $v, u \in R$), then we conclude using $\Rsh$-connectivity of S (resp. R). Without any loss of generality, assume $v \in R \backslash S$ and $u \in S \backslash R$.

By hypothesis, there exists a vertex $x \in S \cap R$; necessarily $v \neq x$. By $\Rsh$-connectivity of R, there is a simple, open, bridge-free path p from v to x inside R, such that the reverse of its first edge is not colored $\mathsf{c}(e)$. Consider p' a minimal prefix of p ending in $S \cap R$, so that p' is a simple, bridge-free path with no vertex in S safe for its target $\mathsf{t}(p') \in S \cap R$. Remark p' is open (and not empty) for its source $v \notin S$ cannot be its target. Furthermore, the first edge of p' is the first edge of p.

Call f the last edge of p'. By $\Rsh$-connectivity of S, and as $R \ni \mathsf{t}(p') \neq u \notin R$, there is a simple, open, bridge-free path q in S from $\mathsf{t}(p')$ to u, such that the reverse of its first edge is not colored $\mathsf{c}(f)$. Thus, $p' \cdot q$ is a simple, open, bridge-free path from v to u (Lemma 3.5). Its first edge is the first edge of p', so of p, thence $e \overset{p' \cdot q}{\Rsh} u$. □

In other words, the previous lemma tells us that a connected (*i.e.* non-disjoint) finite union of $\Rsh$-connected is $\Rsh$-connected. An immediate corollary is that for union of bridge-free cycles, being $\Rsh$-connected is the same as being connected.

Corollary 3.24. *A finite union of bridge-free cycles is $\Rsh$-connected if and only if it is connected.*

Proof. The direct implication is trivial. The reverse one follows from Lemmas 3.22 and 3.23, with an induction on the number of bridge-free cycles. Let $\Omega = \bigcup_{i=1}^{n} \omega_i$ be a connected union of $n \in \mathbb{N}$ bridge-free cycles, denoted ω_i for $i \in [\![1; n]\!]$. The empty case $n = 0$ is trivial. Otherwise, $n \geq 1$ and $\Omega = S \cup \omega_n$ with $S = \bigcup_{i=1}^{n-1} \omega_i$ a union of $n-1$ bridge-free cycles. By connectivity of Ω, each connected component S_j of S respects $S_j \cap \omega_n \neq \emptyset$, with S_j a connected union of bridge-free cycles. By induction hypothesis, each S_j is $\Rsh$-connected, and ω_n also is by Lemma 3.22. Then, Ω is $\Rsh$-connected by repeated applications of Lemma 3.23 on $\omega_n \cup S_1$, $\omega_n \cup S_1 \cup S_2$, …, $\omega_n \cup S = \Omega$. □

3.3.3.2 Finding a splitting vertex

Lemma 3.25. *Take $\Omega \in \mathfrak{O}$ and e an edge from some vertex v in Ω to a vertex u not inside, such that for all edges f of target v, if $f \neq \overline{e}$ then $\mathsf{c}(f) \neq \mathsf{c}(\overline{e})$. For all vertices $x \in \Omega$, $e \Rsh x$ cannot hold.*

Proof. An illustration of this proof is given on Figure 3.7. Towards a contradiction, assume $e \overset{p}{\Rsh} x$ for some $x \in \Omega$. Up to taking a prefix, p has for only vertex in Ω its target x, with $\Omega \not\ni u \neq x \in \Omega$. Call f the last edge of the non-empty path p. As Ω is $\Rsh$-connected by Corollary 3.24, there exists a path q in Ω from x to v with either q empty or $f \overset{q}{\Rsh} v$.

By Lemmas 3.4 and 3.5 and Remark 3.14, $\omega = e \cdot p \cdot q$ is a cycle since p cannot start with $\overline{e}$. Moreover, it is bridge-free using the hypotheses on the paths and the one on the coloration of e to obtain $\mathsf{c}(\overline{e})$ different from the color of the last edge of $p \cdot q$.[3] Therefore, $\Omega \cup \omega \supset \Omega$ is a non-empty connected union of bridge-free cycles, contradicting the maximality of $\Omega \in \mathfrak{O}$. $\square$

Using the last result and hypothesis (H^1_G), we prove a maximal element for $\lhd$ cannot belong to a bridge-free cycle.

Lemma 3.26. *Take G a graph with an edge-coloring c respecting (H^0_G) and (H^1_G), and $\Omega \in \mathfrak{O}$. For any edge f with a target in Ω, $f = \overline{\mathfrak{e}(\Omega)}$ or $f \overset{p\cdot\mathfrak{e}(\Omega)}{\lhd} \mathfrak{e}(\Omega)$ with p inside Ω.*

Proof. Assume $f \neq \overline{\mathfrak{e}(\Omega)}$, for otherwise we are done. Call $v = \mathsf{t}(f) \in \Omega$, $u = \mathsf{s}(\mathfrak{e}(\Omega)) \in \Omega$ and $x = \mathsf{t}(\mathfrak{e}(\Omega)) \notin \Omega$, thanks to (H^0_G). By Corollary 3.24, there exists a path p in Ω from v to u with either p empty or $f \overset{p}{\Rsh} u$. The path $p \cdot \mathfrak{e}(\Omega)$ is simple and open as all vertices of p belong to Ω while the target x of $\mathfrak{e}(\Omega)$ is outside (Lemma 3.4). Moreover, $p \cdot \mathfrak{e}(\Omega)$ is bridge-free as p is bridge-free and using (H^1_G). Hence, if p is non-empty then $f \overset{p\cdot\mathfrak{e}(\Omega)}{\Rsh} \mathfrak{e}(\Omega)$ follows as the first edge of $p \cdot \mathfrak{e}(\Omega)$ is the first edge of p. If p is empty, then $v = u$ and $f \overset{\mathfrak{e}(\Omega)}{\Rsh} \mathfrak{e}(\Omega)$ using $f \neq \overline{\mathfrak{e}(\Omega)}$ and (H^1_G). In both cases, we get $f \overset{p\cdot\mathfrak{e}(\Omega)}{\Rsh} \mathfrak{e}(\Omega)$.

Furthermore, take any vertex y and assume $\mathfrak{e}(\Omega) \overset{q}{\Rsh} y$. Then $y \notin \Omega$ by Lemma 3.25 and (H^1_G). In particular, y cannot be a vertex of p, and the only other vertex of $p \cdot \mathfrak{e}(\Omega)$ is its target x, which cannot be the target of q as q is simple, open and of source x. Hence, $f \overset{p\cdot\mathfrak{e}(\Omega)}{\lhd} \mathfrak{e}(\Omega)$. $\square$

This handles edges whose targets are in bridge-free cycles. For the others, we can apply the Small Bungee Jumping if the pier we find does not belong to a bridge-free cycle, and the Big Bungee Jumping if it does. In this last case, we need some study of paths, making the proof of the next result a bit long.

Proposition 3.27. *Take G a graph with an edge-coloring c respecting (H^0_G) and (H^1_G). Let v be a non-splitting vertex of G, and $e \notin E_{out}$ an edge of target v. There exists $f \in E_{in}$ such that $e \lhd f$.*

Proof. If v belongs to a bridge-free cycle, then the result follows by Lemma 3.26. Therefore, we assume it is not the case.

As v is not splitting, $\mathcal{M}_v \neq \emptyset$. Take some $\omega \in \mathcal{M}_v$. This cycle contains at least one bridge – and in particular is not a loop, for v is not a pier of this cycle. Up to reversing ω, assume it starts with an edge not of support e nor making a bridge with e. This is possible as the first and last edges of ω do not make a bridge, applying Fact 3.6. Notice we use here that ω is not a loop, for otherwise we could have ω being the loop edge e. Denote by κ the first pier of ω. Remark $v \neq \kappa$ as κ is internal in ω of source v. We have two cases, according to whether κ belongs to a bridge-free cycle or not.

If κ does not belong to a bridge-free cycle, call f the last edge of $\omega_{v\to\kappa}$, which is one of the two edges making a bridge at κ; in particular, f is a bridge arch. By Corollary 3.11, any simple open

[3]We need here that $\overline{e}$ is colored differently by c than all other edges of target v not only in Ω but in the whole graph, because we may have $v = x$ and q empty, therefore we need our hypothesis for f the last edge of p – which cannot be $\overline{e}$ by Remark 3.14.

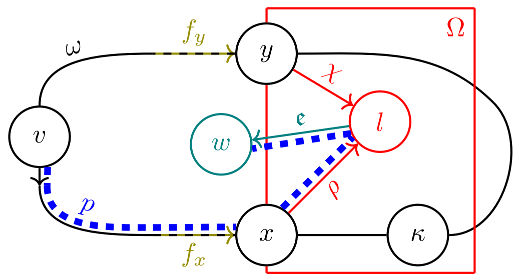


Figure 3.8: Illustration of named elements in the proof of Proposition 3.27: case where v is not in a bridge-free cycle but κ is

bridge-free path, starting from κ with an edge g such that $\mathsf{c}(\overline{g}) \neq \mathsf{c}(f)$, cannot end on a vertex of ω. So $e \overset{\omega_{v\to\kappa}}{\lhd} f$.

We suppose from now on that κ belongs to a bridge-free cycle, and set Ω the maximal connected union of bridge-free cycles containing κ. The set Ω is non-empty by hypothesis, so $\Omega \in \mathfrak{O}$. We will now name some vertices, edges and paths; see Figure 3.8 for an illustration. We use the shorthand notation $\mathfrak{e}$ for $\mathfrak{e}(\Omega)$. Set $l = \mathsf{s}(\mathfrak{e})$ and $w = \mathsf{t}(\mathfrak{e})$; by (H_G^0), $l \in \Omega \not\ni w$. Pose x (resp. y) the first (resp. last) vertex of ω belonging to Ω. Remark $v \notin \{x; y\}$ as v is not in any bridge-free cycle thus not in Ω, and $\kappa \in \omega_{x\to y}$. Let f_x be the last edge of $\omega_{v\to x}$, and f_y be the reverse of the first edge of $\omega_{y\to v}$. By Lemma 3.26, there exists a path ρ in Ω between x and l such that either ρ is empty or $f_x \overset{\rho\cdot\mathfrak{e}}{\lhd} \mathfrak{e}$. Similarly, by Lemma 3.26, there exists a path χ in Ω between y and l such that either χ is empty or $f_y \overset{\chi\cdot\mathfrak{e}}{\lhd} \mathfrak{e}$.

If there is a path q with source w and target some vertex $u \in \omega_{y\to v} \cdot \omega_{v\to x}$ such that q is empty or $\mathfrak{e} \overset{q}{\Rsh} u$, we will find a contradiction. Note that no vertex of q belongs to Ω otherwise we contradict Lemma 3.25 by (H_G^1). In particular $u \notin \{x; y\}$, q and ρ are disjoint, q and χ are disjoint, and $\mathfrak{e} \cdot q$ is simple and open. We then have a contradiction by Corollary 3.12: either $\mathfrak{e}$ belongs to a bridge-free cycle and it contradicts the maximality of Ω, or there is a cycle starting with v, with no bridge at v and with strictly less bridges than ω, contradicting $\omega \in \mathcal{M}_v$. We thus conclude that there is no such path as q. It means that w does not belong to $\omega_{v\to x}$ and that $\mathfrak{e} \Rsh u$ with $u \in \omega_{v\to x}$ cannot hold.

Since $w \notin \omega_{v\to x}$, by Lemma 3.5, $p = \omega_{v\to x} \cdot \rho \cdot \mathfrak{e}$ is a simple open path (see Figure 3.8). Moreover, it is bridge-free by construction. We thus get $e \overset{p}{\Rsh} \mathfrak{e}$. Finally $e \overset{p}{\lhd} \mathfrak{e}$: otherwise $\mathfrak{e} \Rsh u$ with u in p but we have already seen that it cannot happen with u in $\omega_{v\to x}$, and not with u in ρ as well since $f_x \overset{\rho\cdot\mathfrak{e}}{\lhd} \mathfrak{e}$ or ρ is empty. $\square$

Proof of Theorem 3.19. Take $e \in E$ maximal for $\lhd$ (restricted to E): $\mathsf{t}(e)$ is splitting. Indeed, otherwise there would be some $f \in E_{in}$ such that $e \lhd f$ by Proposition 3.27. As E dominates E_{in}, we would get $f \in E$ or $e \lhd f \lhd g$ for some $g \in E$, contradicting the maximality of e. $\square$

3.3.4 All hypotheses are needed

We give here examples showing all hypotheses of Theorem 3.19 are needed, even with an arc-coloring. The hypothesis (H_G^0) will not be considered, as it is more on the side of the definition of

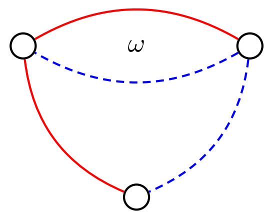


Figure 3.9: Graph without a splitting vertex, with a sole bridge-free cycle ω, having an arc out of it not of the color of any adjacent arc out of ω

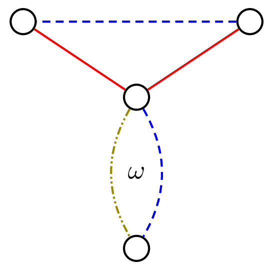


Figure 3.10: Graph without a splitting vertex, with a sole bridge-free cycle ω, having an arc out of it not of the color of any adjacent arc in ω

the function $\mathfrak{e}$ than a condition it should respect. On all figures of this section, colors are identified by the coloration and the shapes of the arcs, for all edge-colorings given here are arc-coloring.

We need hypothesis (H_G^1), namely that $\overline{\mathfrak{e}(\Omega)}$ is colored differently by c than all other edges of the same target in the whole graph. In particular, it is important to have this hypothesis for all other edges f in the whole graph, not only for those in Ω. Indeed, we need it to be colored differently than these edges in Ω looking at the graph on Figure 3.9. This graph has no splitting vertex, and has for only bridge-free cycle ω, which has an out-going edge (both the solid and dashed edges work), colored differently than other edges outside ω.

We also need (H_G^1) for edges from a vertex out of Ω and towards v, as shown by the graph on Figure 3.10. This graph also has no splitting vertex, has for only bridge-free cycle ω, which has an out-going edge (both solid edges work) colored differently from all other edges in the cycle.

To have a valid E, one needs the hypothesis (H_G^2). A counter-example is the graph on Figure 3.11, with exactly two disjoint bridge-free cycles ω and ω', whose edges $\mathfrak{e}(\{\omega\})$ and $\mathfrak{e}(\{\omega'\})$ share a same support (the dash-dotted arc). This graph has no splitting vertex, but for all non-empty maximal connected union of bridge-free cycles Ω, meaning $\Omega \in \{\{\omega\};\{\omega'\}\}$, we have $\mathfrak{e}(\Omega)$ between a vertex in Ω and a vertex outside, with $\overline{\mathfrak{e}(\Omega)}$ colored differently than all other edges with the same target. This is not a counter-example of Theorem 3.19, for no E respecting the statement of the lemma can be defined: a set dominating $\{\mathfrak{e}(\{\omega\});\mathfrak{e}(\{\omega'\})\}$ must contains this set as both elements are maximal for $\lhd$; but then it contains $\overline{\mathfrak{e}(\{\omega\})} = \mathfrak{e}(\{\omega'\})$.

Lastly, there is a counter-example in the case where the chosen set of edges E is not disjoint from E_{out}. Consider the graph on Figure 3.12, with e the dash-dotted edge taken towards the (sole) bridge-free cycle ω. Then this graph respects (H_G^0), (H_G^1) and (H_G^2), e is maximal for $\lhd$ but its target is not splitting.

Figure 3.11: Graph without a splitting vertex and respecting (H_G^0) and (H_G^1)

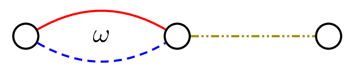


Figure 3.12: Graph respecting (H_G^0), (H_G^1) and (H_G^2) with an edge in E_{out} maximal for $\lhd$ but whose target is not splitting

As a last remark, consider a parallel with the single switching cycle conjecture in [HG05] (Conjecture 4.5): when replacing "non-empty maximal connected unions of bridge-free cycles" simply by "bridge-free cycles" (in the definition of $\mathfrak{O}$, Definition 3.18), does the theorem still hold? The answer here is no: the graph depicted on Figure 3.13 is a counter-example, having no splitting vertex but such that for every bridge-free cycle, there exists an arc going out of it, not of the color of any other adjacent arc. Nonetheless, this example seems hard to adapt as a counter-example of the single switching cycle conjecture in the context of proof-nets. It still shows that if this conjecture holds, then to prove it one must use the criterion (P3$^-$) from Conjecture 4.5 not only for the set of linkings considered, but also for some sub-sets of it.

3.3.5 Corollaries of the main result

In this section, we give various results Theorem 3.19 implies. In particular, we prove Yeo's theorem (Theorem 3.31) as well as one of its known generalizations in the framework of H-coloring (Theorem 3.33).

3.3.5.1 Yeo's theorem and other generalizations as corollaries

Here are given a succession of generalization of Yeo's theorem that Theorem 3.19 implies, restricting iteratively until reaching Yeo's theorem itself. To begin with, one can remove the parameter E, up to adding the hypothesis (H_G^2) from Lemma 3.20: for $\Omega \neq \Omega' \in \mathfrak{O}$, $\mathfrak{c}(\Omega) \neq \overline{\mathfrak{c}(\Omega')}$.

Theorem 3.28 (Local Yeo with Cycles)**.** *Take G a graph with an edge-coloring c and at least one vertex. If G respects (H_G^0), (H_G^1) and (H_G^2), then it contains a splitting vertex.*

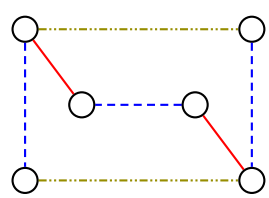


Figure 3.13: Graph without a splitting vertex but with every bridge-free cycle (all three cycles are bridge-free) having an arc out of it not of the color of any adjacent arc

Proof. If there is a vertex which is not the target of any edge, it is splitting. Otherwise, the set E of edges with a defined target and not in E_{out} is finite and non-empty – either $\mathfrak{O}$ is empty or there is at least one $\mathfrak{e}(\Omega)$, not in E_{out} by (H^2_G) – and thus contains a maximal element e with respect to $\lhd$ (Lemma 3.16). Using Lemma 3.20 to get $E_{in} \subseteq E$, one can apply Theorem 3.19 to conclude that the vertex $\mathsf{t}(e)$ is splitting. $\square$

On the other hand, one can simplify by asking for no bridge-free cycle.

Theorem 3.29 (Parameterized Local Yeo)**.** *Take G a graph with an edge-coloring c and pose E a set of edges of G dominating $E_{in} = \{e \mid e \text{ is a bridge arch}\}$. If G has no bridge-free cycle, the target (if it is defined) of any element e of E maximal for $\lhd$ (*i.e.* for $\lhd$ restricted to E) is a splitting vertex.*

Proof. As G has no bridge-free cycle, it trivially respects (H^0_G) and (H^1_G), which are about non-empty union of bridge-free cycles, and $E_{out} = \emptyset$. By Theorem 3.19, the result follows. $\square$

This last result can be used as the basis for sequentialization of $\mathsf{MLL}^{0,2}_{uf}$ proof-nets, in the same spirit Theorem 3.19 will be used for sequentialization of $\mathsf{MALL}^{0,2}_{uf}$ proof-nets in Section 4.3.

Theorem 3.30 (Local Yeo)**.** *Take G a graph with an edge-coloring c and at least one vertex. If G has no bridge-free cycle, then it contains a splitting vertex.*

Proof. As G has no bridge-free cycle, it trivially respects (H^0_G), (H^1_G) and (H^2_G), which are about non-empty unions of bridge-free cycles. We conclude by Theorem 3.28. $\square$

As an example, the graph depicted on Figure 3.2 respects the hypotheses of this last theorem, and u is its only splitting vertex.

Let us now move to more standard terminology on undirected total graphs and show that Yeo's theorem is a direct consequence of our local version. The definitions of connectivity, $G \setminus v$ and $G - v$ in Section 3.1 provide an alternative characterization of splitting vertices in edge-colored graphs: a vertex v is splitting if and only if any two edges with target v and connected in $G \setminus v$ have the same color. We recover the more standards notions for (total) undirected graphs from those we defined for (partial) graphs:

- Connectivity as defined for (partial) graphs leads to the standard notion when restricted to total graphs.
- In an undirected graph, there is no point in making a distinction between arcs and edges, and the natural notion of edge-coloring is arc-coloring.
- In an arc-colored undirected graph, an **alternating cycle** is a cycle where all consecutive arcs are of different colors, and its first and last arcs also respect this property; *i.e.* it is a bridge-free cycle.

Theorem 3.31 (Yeo's Theorem)**.** *Let G be a non-empty arc-colored undirected graph with no alternating cycle. There exists a vertex v of G such that no connected component of $G - v$ is joined to v with arcs of more than one color.*

Proof. Let c be the arc-coloring of G, we can see G as a (directed partial multi) graph by choosing an arbitrary direction on arcs, and define an edge-coloring c' by $\mathsf{c}'(a^+) = \mathsf{c}'(a^-) = \mathsf{c}(a)$. Bridge-free cycles with respect to c' are exactly alternating cycles with respect to c. By Theorem 3.30, we get a splitting vertex, that is a vertex v such that any two edges with target v and connected in $G \setminus v$ have the same color given by c'. This means in particular that no connected component of $G - v$ is joined to v with arcs of more than one color. □

There is a simple proof of Theorem 3.29, hence of Theorem 3.31, as a direct corollary from the Small Bungee Jumping (Corollary 3.11) – following the corresponding part in the proof of Proposition 3.27. This is not the first very simple proof of Theorem 3.31: see [Nen14; LN22].

Alternative proof of Theorem 3.29. Take e a maximal element of E, and call $v = \mathsf{t}(e)$. Towards a contradiction, assume v is not splitting.

Then, $\mathcal{M}_v \neq \emptyset$: take some $\omega \in \mathcal{M}_v$, considered as starting from v. This cycle contains at least one bridge – and in particular is not a loop, for v is not a pier of this cycle. Up to reversing ω, assume it starts with an edge not of support e nor making a bridge with e. This is possible as the first and last edges of ω do not make a bridge, applying Fact 3.6. Notice we use here that ω is not a loop, for otherwise we could have ω being the loop-edge e. Denote by κ the first pier of ω. Remark $v \neq \kappa$ as κ is internal in ω of source v. Call f the last edge of $\omega_{v\to\kappa}$, which is one of the two edges making a bridge at κ; in particular, f is a bridge arch. By Corollary 3.11 and the absence of bridge-free cycles, there is no path p such that $f \overset{p}{\Rsh} u$ for u a vertex of ω: hence $e \overset{\omega_{v\to\kappa}}{\lhd} f$.

By hypothesis on E, either $f \in E$ or we can find $g \in E$ with $f \lhd g$. In both cases, e is not maximal for $\lhd$ in E: contradiction. □

3.3.5.2 A generalization to H-coloring as a corollary

Actually, Theorem 3.30 also implies another generalization of Yeo's theorem to H-colored graphs from [Gal+22]. Given an undirected simple graph H, an **H-coloring** c of an undirected graph G is an arc-coloring with colors the vertices of H. An **H-cycle** in an H-colored graph is a cycle in which the colors of consecutive arcs (including the last and first ones) are adjacent in H.[4]

A **complete multipartite** graph G is an undirected simple graph which can be decomposed into fully connected independent sets of vertices, *i.e.* its vertices are $\mathcal{V} = S_1 \sqcup \ldots \sqcup S_k$ (disjoint union) where each S_i is an independent set of vertices (no arcs in G between vertices of S_i) and if $v \in S_i$ and $u \in S_j$ with $i \neq j$, then v and u are adjacent in G.

Definition 3.32. Given (G, c) an H-colored graph, and v a vertex of G, G_v is the undirected simple graph with vertices the arcs of G incident to v, and an arc between e and f if and only if their colors $\mathsf{c}(e)$ and $\mathsf{c}(f)$ are adjacent in H.

Note that G_v only depends on the neighborhood of v (arcs incident to v) and on the sub-graph of H induced by the colors of these arcs.

Theorem 3.33 ([Gal+22, Theorem 2]). *Take H an undirected simple graph and G a non-empty H-colored graph. Assume:*

- *G has no H-cycle;*

[4] When H is a complete graph, we recover the standard notion of arc-coloring, and H-cycles then correspond to alternating cycles.

- *for every vertex v of G, G_v is a complete multipartite graph.*

Then there exists a vertex v of G such that every connected component D of $G-v$ satisfies that the set of arcs of G between v and vertices of D is an independent set in G_v.

Proof. Call c the H-coloring of G. We equip G with an arbitrary direction for arcs. We define an edge-coloring c' of G, with codomain (*i.e.* colors) the sets of arcs of G: $\mathsf{c}'(a^\varepsilon)$ is the independent set in $G_{\mathsf{t}(a^\varepsilon)}$ to which a belongs. The result follows by Theorem 3.30, for G (with c) having no H-cycle implies it (with c') has no bridge-free cycle. □

What happens is that, given the assumption that G_v is a complete multipartite graph, one can consider its independent sets of vertices as corresponding to a given color. Considered locally on the neighborhood of v, these colors define an edge-coloring. Edge-coloring seems to be a more natural tool to describe the situation than the complete multipartite structure of some induced sub-graphs of H. Remark a version of the theorem with $G \setminus v$ instead of $G - v$ also holds, with the same proof.

Our version, Theorem 3.33, is a slight generalization of [Gal+22, Theorem 2] as they require H to be loop-free, and H and G to have no isolated vertices. The presence of exactly one or of more than one arc between two vertices of H has no impact on H-coloring, hence on the theorem. Remark our simple proof of Theorem 3.29 yields a simple proof of Theorem 3.33. The proof of Yeo's theorem from [LN22] is also written as generalizable to a simple proof of [Gal+22, Theorem 2].

As a last remark, one can also generalizes Theorem 3.33 in the presence of some H-cycles using Theorem 3.28.

Theorem 3.34. *Take H an undirected simple graph and G a non-empty H-colored graph. Call $\mathfrak{O}'$ the set of non-empty maximal (for the inclusion) connected unions of H-cycles in G. Equip G with a function $\mathfrak{e}$ which for every $\Omega \in \mathfrak{O}'$ gives an arc $\mathfrak{e}(\Omega)$. Assume:*

($H_G^0{}'$) *for every $\Omega \in \mathfrak{O}'$, the arc $\mathfrak{e}(\Omega)$ is between a vertex s_Ω in Ω and some t_Ω outside;*

($H_G^1{}'$) *for every $\Omega \in \mathfrak{O}'$ and for every arc f adjacent to s_Ω and distinct from $\mathfrak{e}(\Omega)$, we have $\mathsf{c}(f)$ and $\mathsf{c}(\mathfrak{e}(\Omega))$ adjacent in H;*

($H_G^2{}'$) *for $\Omega \neq \Omega' \in \mathfrak{O}'$, $\mathfrak{e}(\Omega) \neq \mathfrak{e}(\Omega')$.*

($H_G^3{}'$) *for every vertex v of G, G_v is a complete multipartite graph.*

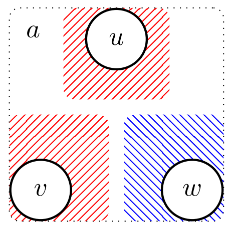
Then there exists a vertex v of G such that every connected component D of $G-v$ satisfies that the set of arcs of G between v and vertices of D is an independent set in G_v.

Proof. We define the same coloring c' as in the proof of Theorem 3.33, using ($H_G^3{}'$), so as to apply Theorem 3.28. This is possible as elements of $\mathfrak{O}'$ for c correspond to those of $\mathfrak{O}$ for c', which directly implies (H_G^0) and (H_G^2) from ($H_G^0{}'$) and ($H_G^2{}'$). Furthermore, ($H_G^1{}'$) also implies (H_G^1) for if $\mathsf{c}(f)$ and $\mathsf{c}(\overline{\mathfrak{e}(\Omega)})$ are adjacent in H then $\mathsf{c}'(f) \neq \mathsf{c}'(\overline{\mathfrak{e}(\Omega)})$ (assuming f directed towards s_Ω and $\mathfrak{e}(\Omega)$ from it, up to reversing these edges). □

3.3.5.3 A generalization to hypergraphs as a corollary

Let G be an undirected hypergraph: it is composed of a set of vertices and a set of (hyper)arcs, with an (hyper)arc being a set of at least two vertices.[5] A **local coloring** c of G is a function

[5]Hypergraphs generalize multigraphs except we lose the notion of loop, but this does not really matter here as a loop is a bridge-free cycle.



Local coloring c:

$\mathsf{c}(u)(a) =$ red
$\mathsf{c}(v)(a) =$ red
$\mathsf{c}(w)(a) =$ blue

Figure 3.14: Example of locally colored hypergraph

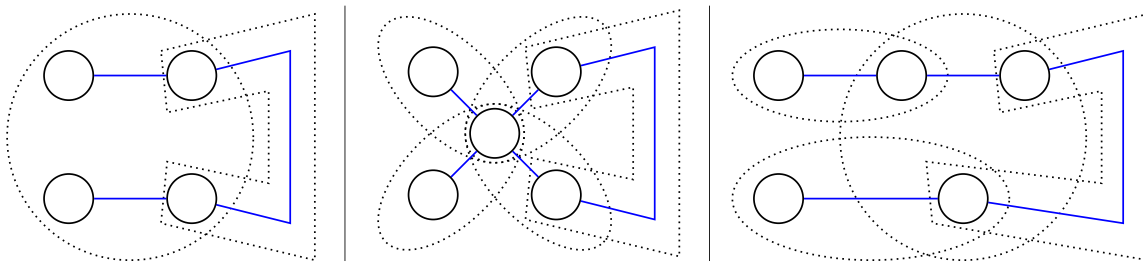


Figure 3.15: Examples of a path simple of type 1 but not of type 2 (left), of type 2 but not of type 1 (middle) and of type 3 but not of type 4 (right)

taking as arguments a vertex v, an hyperarc a incident to (*i.e.* containing) v and returning a color $\mathsf{c}(v)(a)$. An example of locally colored hypergraph is given on Figure 3.14. On our illustrations, hyperarcs will be represented as dotted shapes.

A **path** is an alternating sequence of vertices and arcs $(v_1, a_1, v_2, a_2, \dots, v_n)$ (with at least one vertex) such that for all $i \in [\![1; n]\!]$, the arc a_i is incident to the vertex v_i as well as to the vertex v_{i+1} if $i \neq n$. Its **source** is v_1 and its **target** v_n. A path is **closed** if $v_1 = v_n$ and **open** otherwise. As in Section 3.1.3, one can define the concatenation of paths by identifying the source and target occurrences and concatenating the two lists.

We can define several notions of **simple** paths in an hypergraph, by generalizing those on graphs.

1. Simple of type 1: does not contain the same vertex twice (except possibly as its endpoints), and is not of the shape (v, a, u, a, v) – this is the adaptation to hypergraphs of non-bouncing.

2. Simple of type 2: does not contain the same arc twice.

3. Simple of type 3: simple of type 1 and of type 2.

4. Simple of type 4: simple of type 2 such that any two non-consecutive arcs (modulo this path, with the last and first arcs consecutive in the case of a closed path) share no vertex.

We have simple of type 4 implies simple of type 3, which implies simple of type 1 and of type 2. There exist simple paths of type 1 which are not simple of type 2, and vice-versa; there also are simple paths of type 3 not simple of type 4 – see Figure 3.15 for counter-examples.

A **cycle** (of type i) is a non-empty, closed, simple path (of type i). A **bridge** of **pier** v is a pair of (hyper)arcs (a, b) both incident to v such that $\mathsf{c}(v)(a) = \mathsf{c}(v)(b)$. The **number of bridges of a**

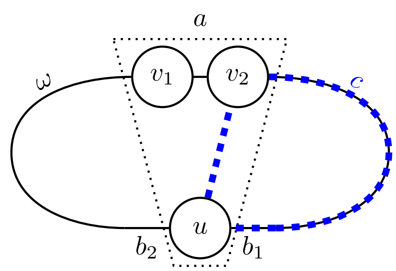


Figure 3.16: Illustration of the proof of Lemma 3.35

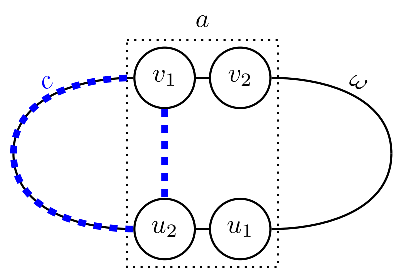


Figure 3.17: Illustration of the proof of Lemma 3.36

path is the number of consecutive arcs inside, say $(v_i, a_i, v_{i+1}, a_{i+1}, v_{i+2})$, such that (a_i, a_{i+1}) is a bridge of pier v_{i+1}. Remark that the reverse of a path contains the same bridges as this path. A **bridge-free** path is one without bridge. For a cycle to be **bridge-free**, we further ask its first and last arcs to not form a bridge of pier v, where v is the source (and target) of the cycle.

We prove that a weaker hypothesis among all "G contains no bridge-free cycle of type i" is the version with $i = 1$.

Lemma 3.35. *If a locally colored hypergraph G contains a bridge-free cycle of type* 3*, then it contains a bridge-free cycle of type* 4.

Proof. Take ω a bridge-free cycle of type 3. If it is of type 4 the lemma holds, so assume it is not the case. Thus, it contains arcs a, b_1, b_2 and vertices v_1, v_2 and u such that a contains these three vertices and ω is $(v_1, a, v_2) \cdot \omega_1 \cdot (b_1, u, b_2) \cdot \omega_2$ (up to changing the endpoint of ω by rotating it), see Figure 3.16. As ω is bridge-free, we have $\mathsf{c}(u)(b_1) \neq \mathsf{c}(u)(b_2)$. By symmetry (up to reversing and rotating ω), $\mathsf{c}(u)(b_1) \neq \mathsf{c}(u)(a)$. We set $c = (v_2) \cdot \omega_1 \cdot (b_1, u, a, v_2)$, which is a bridge-free (as we do not create a bridge at u, and there is none at v_2 which has the same incident arcs in c and ω) cycle of type 3 whose size (*i.e.* number of arcs) is strictly smaller than the one of ω. By finite induction, we obtain a bridge-free cycle of type 4. $\square$

Lemma 3.36. *If a locally colored hypergraph G contains a bridge-free cycle of type* 1*, then it contains a bridge-free cycle of type* 3.

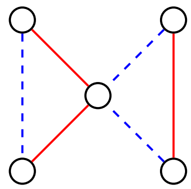


Figure 3.18: Simple graph with a bridge-free closed arc-simple path, but no bridge-free closed vertex-simple path

Proof. Take ω a bridge-free cycle of type 1. If it is of type 3 the lemma holds, so assume it is not the case. Thus, it contains an arc a and vertices v_1, v_2, u_1 and u_2 such that a contains these four vertices and ω is $(v_1, a, v_2) \cdot \omega_1 \cdot (u_1, a, u_2) \cdot \omega_2$ (up to changing the endpoint of ω by rotating it), see Figure 3.17. We set $c = (v_1, a, u_2) \cdot \omega_2$, which is a bridge-free (as we do not create a bridge at v_1 nor u_2 for we use the same incident arcs in c as in the bridge-free ω) cycle of type 1 whose size (*i.e.* number of arcs) is strictly smaller than the one of ω. By finite induction, we obtain a bridge-free cycle of type 3. □

By Lemmas 3.35 and 3.36, a weaker hypothesis among "G contains no bridge-free cycle of type i" is for $i = 1$ – which is equivalent to $i = 3$ and $i = 4$, and implied by $i = 2$. This hypothesis is stronger for $i = 2$ than for the other values of i, with the same example as in graphs between vertex-simple and arc-simple – see Figure 3.18. Thus, in what follows simple means simple of type 1, allowing to use Theorem 3.30 so as to obtain a similar result on hypergraphs.

Theorem 3.37. *Take G a non-empty undirected hypergraph, with a local coloring c. If G has no bridge-free cycle, then there exists a vertex v of G such that every connected component D of $G - v$ satisfies that all arcs containing v and a vertex of D have the same color for $\mathsf{c}(v)$.*

Proof. Consider G' the graph built from G with as vertices the vertices of G and an arc $a_{\{v;u\}}$ between distinct vertices v and u for each hyperarc a of G containing v and u – choosing arbitrarily its source and target among the two possibilities. We define a local coloring c' of G' thanks to the coloring c of G, by $\mathsf{c}'(v)(a^{\varepsilon}_{\{v;u\}}) = \mathsf{c}(\mathsf{t}(a^{\varepsilon}_{\{v;u\}}))(a)$.

Remark that any path is simple in G if and only if it is simple in G', and that it has the same number of bridges in both graphs. Thus, G' has no bridge-free cycle as G does not, and by Theorem 3.30 there exists a vertex v such that every cycle containing v has a bridge of pier v. This property of v, holding in G', also holds in G by the previous remark, which proves the theorem. □

3.3.6 Results equivalent to Yeo's theorem

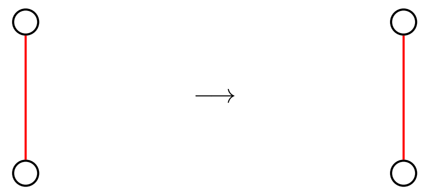
Since the first version of this thesis, and thanks to some discussions with Lê Thành Dũng Nguyễn and Colin Geniet, I discovered that Theorem 3.30 and Theorem 3.31 are **equivalent** – they can be deduced from each other – and found some links with previously existing results, including that the bungee jumping lemma is more a rediscovered result than a new one.

More precisely, while at first glance Theorem 3.30 seems stronger than Theorem 3.33 which seems itself stronger than Theorem 3.31, they are in fact equivalent.

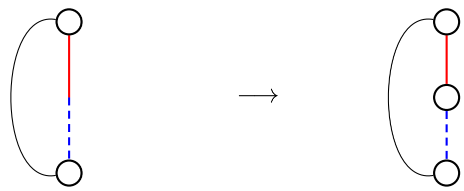
Lemma 3.38. *Theorems 3.30, 3.31 and 3.33 are mutually equivalent.*

Proof. We already saw in the proof of Theorem 3.33 how to deduce Theorem 3.33 from Theorem 3.30 (simply by defining an edge-coloring), and Theorem 3.31 is a particular case of Theorem 3.33 with H being a complete graph. Thence, the only non-trivial direction is proving Theorem 3.30 from Theorem 3.31. A simple encoding of G with an edge-coloring c into G' with an arc-coloring c' does the trick:

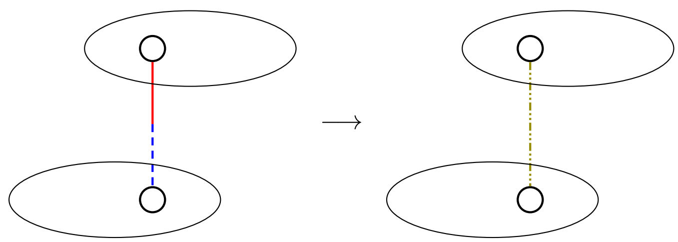
- for each arc a such that $\mathsf{c}(a^+) = \mathsf{c}(a^-)$, set $\mathsf{c}'(a) = \mathsf{c}(a^+) = \mathsf{c}(a^-)$;



- for each arc a such that $\mathsf{c}(a^+) \neq \mathsf{c}(a^-)$ and a belongs to a cycle, replace it with two arcs a_- and a_+ respectively from $\mathsf{s}(a)$ to v_a and from v_a to $\mathsf{t}(a)$, where v_a is a new vertex, and set $\mathsf{c}'(a_-) = \mathsf{c}(a^-)$ and $\mathsf{c}'(a_+) = \mathsf{c}(a^+)$;



- for each arc a such that $\mathsf{c}(a^+) \neq \mathsf{c}(a^-)$ and a does not belong to a cycle (*i.e.* a is a cut-arc), set $\mathsf{c}'(a)$ to be an arbitrary color (it could be $\mathsf{c}(a^+)$ for instance).



On Figure 3.19 is given an example of this encoding (with colors also given by the shape of the arcs, and representing an edge-coloring by a coloring of half-arcs instead of edges). Then one can check that:

- bridge-free cycles in the obtained graph G' correspond to those of G, and in particular G' has no bridge-free cycle if and only if G has none;
- no added vertex v_a is splitting in G';
- a vertex v of G is splitting in G if and only if it is splitting in G'.

Thence, Theorem 3.31 applied on G' yields a splitting vertex of G, proving Theorem 3.30. ☐

Remark 3.39. The encoding in the proof of Lemma 3.38 is not stable by sub-graph, *e.g.* after removing the leftmost splitting vertex in the graph depicted on Figure 3.19, the leftmost solid-dashed arc should not be encoded with a vertex in its middle anymore, because it is no more in a cycle. An encoding stable by sub-graph seems hard to come by. In particular, a trick such as adding a same "gadget" graph in the middle of each (bicolored) arc (graph which may simply be a single vertex or a more complex graph), so as to duplicate each arc and color them as the corresponding edges, cannot work. Indeed, the gadget to add must not have any bridge-free cycle so as to be able

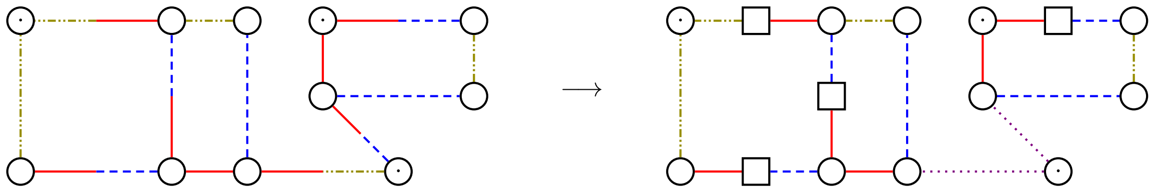


Figure 3.19: Example of encoding of local coloring as edge-coloring, where pointed vertices are splitting ones and square vertices represent added ones

to apply Theorem 3.30, nor should have any splitting vertex as one wants to find as splitting a vertex in the original graph. Such a graph cannot exist by Theorem 3.30 itself! This is the reason why we need a third case in our encoding in the above proof.

Moreover, it is proved in [Sze04, Theorem 6] that Theorem 3.31 is equivalent to each of the four following theorems – we slightly reformulated them – using smart encodings, one of these four being Kotzig's theorem, which has been showed to be equivalent to the sequentialization of $\mathsf{MLL}_{uf}^{0,2}$ in [Ngu20]. One can check that Theorem 3.29 allows to deduce each of these four theorems *without the need of any encoding*, *i.e.* only by defining an edge-coloring on the graph and without any modification of the vertices or arcs. Thence, Theorem 3.29 (or its unparameterized version Theorem 3.30) is a sort of unifying formulation of these equivalent results. Furthermore, this implies that our demonstration, with the use of the bungee jumping lemma, also proves all of these theorems by only adapting the definition of a bridge. We give below the statement of each of these theorems how to prove them using Theorem 3.29.

A **cut-arc** in a graph is an arc a such that the removal of a increases the number of connected components of the graph. A **perfect matching** (or **1-factor**) of a graph G is a set of arcs F such that every vertex is incident to a unique arc in F. It is well known that a perfect matching F in a graph G is unique if and only if G contains no F**-alternating cycle**, which is a cycle whose arcs are alternatively in and out of F, including the last and first ones (it is *e.g.* a simple variant of [Ber57, Theorem 1] which considers F-alternating open paths).

Theorem 3.40 (Kotzig [Kot59])**.** *If a graph G has a unique perfect matching F, then G has a cut-arc which belongs to F.*

Proof. It suffices to define an arc-coloring c of G by $\mathsf{c}(a) = \begin{cases} 1 & \text{if } a \in F \\ 0 & \text{otherwise} \end{cases}$. Then F-alternating cycles are exactly bridge-free cycles, so by Theorem 3.30 (here even Theorem 3.31 would suffice) there is a splitting vertex v. The unique arc of F incident to v is a cut-arc as it is the only edge of target v with color 1. ☐

Given a map $\phi : \mathcal{V} \to \mathcal{A}$ such that $\phi(v)$ is incident to v for all $v \in \mathcal{V}$, a cycle ω is called ϕ**-conformal** if for every vertex v in ω, $\phi(v)$ is in ω.

Theorem 3.41 ([Sey78])**.** *Let G be a graph and let $\phi : \mathcal{V} \to \mathcal{A}$ be a map such that $\phi(v)$ is incident to v for all $v \in \mathcal{V}$. If G has no ϕ-conformal cycle, then there exists a vertex v such that $\phi(v)$ is a cut-arc.*

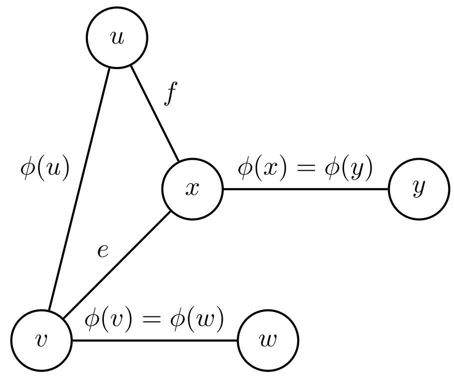


Figure 3.20: No arc-coloring for Theorem 3.41

Proof. Set $\mathsf{c}(e) = \begin{cases} 1 & \text{if } e \text{ has for support } \phi(\mathsf{t}(e)) \\ 0 & \text{otherwise} \end{cases}$. Then ϕ-conformal cycles are exactly bridge-free cycles, so by Theorem 3.30 there is a splitting vertex v. The arc $\phi(v)$ is a cut-arc as it supports the only edge of target v with color 1. ☐

Remark 3.42. Remark that an arc-coloring is not enough to prove Theorem 3.41 without an encoding: consider the graph drawn on Figure 3.20. To have the equivalence between ϕ-conformal cycles and bridge-free cycles, one would need

$$\begin{cases} \mathsf{c}(\phi(x)) \neq \mathsf{c}(e) = \mathsf{c}(e) & \text{looking at } x \\ \mathsf{c}(\phi(v)) \neq \mathsf{c}(\phi(u)) = \mathsf{c}(e) & \text{looking at } v \\ \mathsf{c}(\phi(u)) \neq \mathsf{c}(f) & \text{looking at } u \end{cases}$$

thence $\mathsf{c}(f) = \mathsf{c}(e) = \mathsf{c}(\phi(u)) \neq \mathsf{c}(f)$, absurd.

Theorem 3.43 ([GH83])**.** *Let G be a 2-edge-colored graph. Then either G has a splitting vertex, or G has a bridge-free cycle.*

Proof. There is nothing to do here as this result is Theorem 3.31 restricted to the case with 2 colors, and so is directly a particular case of Theorem 3.30. ☐

A vertex v of a cycle ω is a **turning vertex** of ω if all arcs incident to v in ω are either all of source v or all of target v.

Theorem 3.44 ([SS79])**.** *If a nonempty set S of vertices of a graph G contains a turning vertex of each cycle of G, then S contains a vertex which is a turning vertex of every cycle it belongs to.*

Proof. Define $\begin{cases} \mathsf{c}(a^+) = 1 & \text{if } \mathsf{t}(a) \in S \\ \mathsf{c}(a^-) = 0 & \text{if } \mathsf{s}(a) \in S \\ \mathsf{c}(e) = e & \text{otherwise} \end{cases}$. Then cycles with no turning vertex in S are exactly bridge-free cycles, so by Theorem 3.29 instantiated with E being all edges whose target is in S (which indeed contains all bridge archs as other edges have their own colors), there is a splitting vertex $v \in S$. By definition of the coloring, v is a turning vertex of every cycle it belongs to. ☐

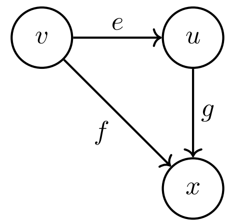


Figure 3.21: No arc-coloring for Theorem 3.44

Remark 3.45. We need the parameterized version here as this theorem has a parameter S itself. Here again, an arc-coloring is not enough for proving Theorem 3.44 without an encoding: consider the graph drawn on Figure 3.21. To have the equivalence between cycles with no turning vertex in S and bridge-free cycles, one would need $\mathsf{c}(f) = \mathsf{c}(g) \neq \mathsf{c}(e) = \mathsf{c}(f)$, absurd.

An important observation is that Shoesmith and Smiley's proof of this theorem in [SS79] is very similar to our alternative proof of Theorem 3.29 by bungee jumping: the key idea of both proofs is to look at cycles with a minimal number of bridges. Still, there are key differences: the absence of an explicit order – the proof of [SS79] builds an infinite bridge-free path – as well as the parameter being on vertices instead of edges. We did not know about [SS79] before writing our proof, so that we rediscovered what we call the bungee jumping; it only came to our attention *via* Szeider's equivalence results [Sze04].

3.3.7 Inductive definitions of graphs and links with proof-nets

Thanks to Yeo's theorem, one finds the following inductive definition for graphs with no bridge-free cycle. A graph with no bridge-free cycle is (isomorphic to a graph) generated by the following grammar:

$$G \ ::= \ \emptyset \mid \ltimes_i G_i$$

where $\emptyset$ is the empty graph, and $\ltimes_i G_i$ denotes taking the graphs G_i (generated by the grammar) and adding a new vertex v with some arcs (or not) between v and these G_i, such that all edges from G_i to v share a same color. Indeed, graphs generated by this grammar have no bridge-free cycle, for the added vertex v cannot belong to such a cycle. Reciprocally, Yeo's theorem gives a splitting vertex, meaning a decomposition into $\ltimes_i G_i$, and removing a vertex cannot create a bridge-free cycle.

One can then see the link with proof-nets, which are graphs corresponding to proofs: from a graph of a particular shape, one finds a corresponding inductive proof. These links between (sequentialization of) proof-nets and general results in graph theory are not new: Yeo's theorem implies Kotzig's theorem (Theorem 3.40), showed to be equivalent to sequentialization of $\mathsf{MLL}_{uf}^{0,2}$ proof-nets in [Ngu20].

This characterization also explains the slightly weird requirements in the definition of $\mathsf{MALL}_{uf}^{0,2}$ proof-nets next chapter, in particular for (P3): as we remove not only one vertex when sequentializing (*i.e.* splitting) a &-vertex, but separate a graph into two graphs in a non-trivial way, one has to preserve the correctness criterion for these sub-graphs. Just asking for splitting vertices can only create graphs with no bridge-free cycles, corresponding to $\mathsf{MLL}_{uf}^{0,2}$ proof-nets, thence the need for a new kind of condition.

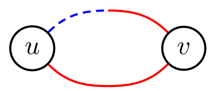


Figure 3.22: Graph where Theorem 3.30 is applicable but not Theorem 3.31 (without an encoding)

Getting an inductive description of a graph was also the goal underlying Theorem 3.44. It is noteworthy to remark that Theorem 3.44 from Shoesmith and Smiley [SS79] can be used to obtain directly a splitting $\parr$ in a $\mathsf{MLL}_{uf}^{0,2}$ proof-net, simply by taking S the set of all $\parr$-vertices. Indeed, such vertices have only one out-going arc, so having each cycle containing a $\parr$ as a turning vertex is equivalent to have each cycle containing two in-arcs of such a vertex; and a $\parr$-vertex is a turning vertex of every cycle it belongs to if and only if its out-going arc is in no cycle. Nevertheless, Theorem 3.44 seems limited to the search of a splitting $\parr$ and hardly applicable for the search of *e.g.* a splitting terminal vertex in a $\mathsf{MLL}_{uf}^{0,2}$ proof-net, while our Theorem 3.29 can do it thanks to its parameter on edges (see Section 4.3.6).

Besides, Theorem 3.44 has been motivated and used by its authors Shoesmith and Smiley in [SS78] to handle proofs represented as graphs, sharing notable similarities to the proof-nets of multiplicative linear logic. Some striking considerations in [SS78], reminiscent of proof-nets, are the interdiction of some kinds of cycles [SS78, Theorems 10.3 and 10.4] (the ones without any turning vertices), a notion of *residues* reminiscing of correctness graphs [SS78, Section 10.2], with more properties when these residues are not only acyclic but also connected [SS78, Section 10.7]. Some parts may even point at some linearity constraints, *e.g.* "But any solution along these lines would require us to take account of multiplicity of occurrence in rules of inference" [SS78, Page 164]. It could be interesting to understand how much about the "static" ideas of $\mathsf{MLL}_{uf}^{0,2}$ proof-nets was already present in [SS78] – the "dynamic" aspects with the cut-elimination procedure do not appear to be a topic of consideration (at least in this book).

As a last remark, we originally needed edge-coloring instead of arc-coloring due to the graph depicted on Figure 3.22 and on which we wanted to apply Theorem 3.30: v is splitting but not u, but if we color this graph with an arc-coloring (without modifying its sets of vertices and arcs) then either both u and v are splitting or neither is. This graph appears naturally in usual definitions of MLL_{uf} proof-nets: it corresponds to a proof with an *ax*-rule above a $\parr$-rule.

3.4 Folklore results

Our main theorem, Theorem 3.19, is not the only result we need from graph theory in order to study proof-nets. The others, given here, are well-known folklore results about the number of connected components in a graph and about simple graphs of maximal degree two – the first result might not be folklore, but can be easily obtained from the inductive syntax given by Yeo's theorem, *c.f.* Section 3.3.7.

Lemma 3.46. *Let G be an arc-colored undirected graph without alternating cycle. The number c of connected components of G respects $c \leq \#|\mathcal{V}| - \#|\mathcal{C}|$ where $\mathcal{V}$ is the vertices of G and $\#|\mathcal{C}|$ the number of colors used on its arcs.*

Proof. By induction on the number of vertices of G.

An undirected graph with no vertex has no connected component and no arc, thus $c = 0 = 0 - 0 = \#|\mathcal{V}| - \#|\mathcal{C}|$.

Otherwise, by Theorem 3.31, there is a splitting vertex v. Call c' the number of connected components of $G - v$, $\mathcal{V}'$ its vertices and $\mathcal{C}'$ the set of colors used on arcs in its arc-coloring. By induction hypothesis, the removal of v preserving being without alternating cycle, $c' \leq \#|\mathcal{V}'| - \#|\mathcal{C}'|$. The removal of v in $G - v$ turns 1 connected component into n components, for some $n \in \mathbb{N}$; *i.e.* $c' = c + n - 1$. We trivially have $\#|\mathcal{V}'| = \#|\mathcal{V}| - 1$. Lastly, as v is splitting, the number of colors used on arcs incident to him is at most n. Thence, $\#|\mathcal{C}'| \geq \#|\mathcal{C}| - n$. We conclude $c = c' - n + 1 \leq \#|\mathcal{V}'| - \#|\mathcal{C}'| - n + 1 \leq \#|\mathcal{V}| - \#|\mathcal{C}|$. □

Lemma 3.47. *For any undirected graph G with vertices $\mathcal{V}$, arcs $\mathcal{A}$ and c connected components, $c \geq \#|\mathcal{V}| - \#|\mathcal{A}|$.*

Proof. By induction on the number of arcs. In a graph with no arc, $c = \#|\mathcal{V}|$. When adding an arc, one either decreases c by one (if the incident vertices were not in the same connected component) or preserves it (if both incident vertices were in a same connected component), allowing to conclude by induction. □

Corollary 3.48. *Let G be an* acyclic *undirected graph. The number c of connected components of G respects $c = \#|\mathcal{V}| - \#|\mathcal{A}|$, where $\mathcal{V}$ is the vertices of G and $\mathcal{A}$ its arcs.*

Proof. Consider an arc-coloring of G where each arc has its own color. Then G seen as an arc-colored graph has no bridge-free cycle, and we obtain $c \leq \#|\mathcal{V}| - \#|\mathcal{A}|$ by Lemma 3.46. We conclude using Lemma 3.47. □

Lemma 3.49. *A total simple loop-free graph G of maximal degree two is a disjoint union of simple paths – meaning cycles, simple open paths and empty paths – where:*

- *vertices of degree* 2 *are internal in a simple path (whether open or a cycle);*
- *vertices of degree* 1 *are endpoints of a simple open path;*
- *vertices of degree* 0 *are isolated,* i.e. *endpoints of an empty path.*

Proof. We prove it by induction on the number of vertices of G.

If G has no vertex, *i.e.* is the empty graph, then the result trivially holds as it is the empty (disjoint) union.

Assume the result holds for all graphs whose number of vertices is n, for n some natural number. Consider a graph G with $n + 1$ vertices, and an arbitrary vertex v inside. By induction hypothesis $G - v$ is a disjoint union of simple paths and cycles. We distinguish cases according to the degree of v.

If $deg(v) = 0$, then G is the disjoint union of $G - v$ and the empty path on v, hence the result.

Suppose $deg(v) = 1$: v is adjacent to a unique u of $G - v$, with an edge e from u to v. Then u is of degree at most one in $G - v$, thus in this graph u is the target of a path p (if $deg(u) = 1$ in G, then p is the empty path). Thus, G is the same union as $G - v$ up to replacing p with $p \cdot e$.

Assume now $deg(v) = 2$, so v is linked to u_1 and u_2, with edges e_1 and e_2 respectively from u_1 and u_2 to v. Then u_1 (resp. u_2) is of degree at most one in $G - v$, so u_1 (resp. u_2) must be the target of a path p_1 (resp. p_2) in $G - v$. If $p_1 \neq \overline{p_2}$, then G is the same union as $G - v$ up to replacing p_1 and p_2 with the path $p_1 \cdot e_1 \cdot \overline{e_2} \cdot \overline{p_2}$. Otherwise, it is the same union as $G - v$ up to replacing p_1 with the cycle $p_1 \cdot e_1 \cdot \overline{e_2}$. □

Lemma 3.50. *A total simple loop-free graph G containing only vertices of degree one or two is a disjoint union of non-empty simple paths and cycles.*

Proof. By Lemma 3.49, G is a disjoint union of simple paths and cycles. As G contains no vertex of null degree, this union cannot contain an empty path. □

3.5 Perspectives

We introduced in this chapter a more general notion of edge-coloring, and a generalization of Yeo's theorem to this setting as well as in the presence of bridge-free cycles respecting some conditions. While these conditions on unions of bridge-free cycles do not seem usual for graph theory, they look very much like those given on unions of switching cycles for proof-nets, allowing us to apply our new theorem in this framework. It could be interesting to see if this novel notion of edge-coloring is useful outside this scope, or more generally outside considerations looking like Yeo's theorem.

More usual considerations of graph theory, such as the study of complexity, seems similar to the standard arc-coloring case. For instance, finding a bridge-free cycle for edge-coloring can be reduced to finding a bridge-free cycle for arc-coloring, simply by putting a vertex in the middle of each arc – which is splitting, but that does not matter for this problem. Likewise, checking a given vertex is splitting seems to be of the same complexity in the edge-coloring case as in the arc-coloring one, at least when considering the naive algorithm that for a vertex check if it belongs to a cycle where it is not a pier.

Chapter 4

Proof-Nets

We will at present change the syntax when considering proofs of linear logic, no longer using sequent calculus proofs but *proof-nets*. A proof-net syntax is interesting to "patch up" the not so good properties of sequent calculus from Chapter 2: as a key example cut-elimination in proof-nets is strongly normalizing and confluent, thus leads to a unique normal form. The key feature of proof-nets enabling these properties while still being about the proofs considered in the sequent calculus syntax is to be a more *canonical* representation of proofs, by identifying proofs up to rule commutation (*i.e.* by being a quotient of sequent calculus proofs up to $\overset{r}{\vdash\!\dashv}$) [HG16].[1] This is what yields better properties, while also sparing us tedious case studies on rule commutations – like the one we will consider in the proof of Proposition 6.17, due to the need to relate the different possible cut-free proofs obtained by cut-elimination. Therefore, an atomic-axiom cut-free proof-net is a canonical representation of a proof up to $=_{\beta\eta}$, which is an identification we wish for in most settings – in the categorical framework, for semantics, when studying isomorphisms, ... Such a syntax also comes with other better properties such as dealing with cut-elimination in a parallel manner. This is why a syntax canonical up to $=_{\beta\eta}$ matters, not speaking about the "bureaucracy" aspect that the order of rules should not matter in some cases [Gir01].

Proof-nets are defined for many sub-systems, among which the first notion given for MLL_{uf} in [Gir87] as well as many other definitions and proofs of sequentialization for this well studied system [DF08; DF06], proof-nets for MELL [Gir87; DR95; Di18], for ALL [Hei11b; Hei11a], for MLL_{uf} and ALL_{uf} with first-order quantifiers [Gir91; Hug18; HHS19], for MLL_{uf} with second-order quantifiers [Str09], for polarized linear logic [Lau99], for differential linear logic [ER06; PT17], for MLL_{uf} with fixpoints [DPS21],... Proof-nets also exist for noncommutative systems [Abr95; AM19], and there have even been some works on proof-nets for classical logic [Rob03; Str10]. Nevertheless, the existence of proof-nets – with the canonicity properties we want, namely identifying proofs up to $\overset{r}{\vdash\!\dashv}$ – is at best doubtful for some sub-systems, such as MLL [HH16a]; see Section 4.5 for more details. Furthermore, proofs that they indeed identify proofs exactly up to rule commutation are scarce: it is done for instance in [HG16] and in [HHS19]. We give here the definition of proof-nets for $\mathsf{MALL}^{0,2}_{uf}$ from [HG05]. Other definitions of proof-nets for MALL_{uf} or $\mathsf{MALL}^{0,2}_{uf}$ exist, such as the original one from Girard [Gir96], or others like [Di11; HH16b; Mai07]. Still, the definition we take is one of the most satisfactory from the point of view of canonicity and cut-elimination (see [HG05;

[1] The authors of [HG16] consider as commutations $\overset{rc}{\vdash\!\dashv}$ and not $\overset{r}{\vdash\!\dashv}$, but both definitions coincide on cut-free proof-nets, which are the objects we want canonical.

HG16], or the introduction of [HH16b] for a comparison of alternative definitions).

Outline This chapter is composed as follows. First is defined the syntax of proof-nets introduced by Hughes & Van Glabbeek in [HG05] (Section 4.1). After are given links between sequent calculus and proof-nets: a proof can be seen as a proof-net and cut-elimination in proofs yields cut-elimination in proof-nets (Section 4.2). We then prove the key result of [HG05], the *sequentialization theorem*: a proof-net is the image of a proof (Section 4.3). This is simply done thanks to our generalization of Yeo's theorem from Section 3.3. We end this chapter by giving some general and easy results on proof-nets that will be of use later (Section 4.4).

4.1 Proof-net for unit-free MALL

We give here the definition of proof-nets from Hughes & Van Glabbeek for unit-free MALL, taken from [HG05], which is defined for both MALL_{uf} and MALL^2_{uf}, and has a trivial extension to $\mathsf{MALL}^{0,2}_{uf}$ with the $\overset{om}{\rightsquigarrow}$ transformation. We also recall composition by cut and cut-elimination for this syntax. Please refer to [HG05] for more details, examples, as well as the intuitions behind the definition. In all that follows, we consider only the $\mathsf{MALL}^{0,2}_{uf}$ sub-system, unless stated otherwise.

4.1.1 Proof-net

A **cut pair** is $A^\perp * A$, for a given formula A; the connective $*$ is unordered, *i.e.* $A^\perp * A = A * A^\perp$. A **cut sequent** $[\Sigma]\ \Gamma$ is composed of a set Σ of (occurrences of) cut pairs and a sequent Γ. When $\Sigma = \emptyset$ is empty, $[\,]\ \Gamma$ may simply be denoted by Γ. The idea is that a proof-net on $[\Sigma]\ \Gamma$ corresponds to a proof of $\vdash \Gamma$ with a *cut*-rule on each cut pair of Σ, which may or may not be shared across slices. For instance, $[X_5^+ * X_6^-]\ X_1^+ \,\&\, X_2^-, X_3^+ \oplus X_4^-$ (where each X_i is an occurrence of the same atom X) is a cut sequent, on which we will instantiate the concepts defined in this part.

We always identify a formula (or a cut pair) A with its **syntactic tree** $T(A)$, having as internal vertices its connectives and as leaves its atoms, with arcs directed from the leaves to the root.[2] In a syntactic tree, a vertex v is a **descendant** of a vertex u if there is a directed path (with edges of sign $+$) from u to v. In this case, we also say u is an **ancestor** of v. As an example, the root is a descendant of the leaves. For all our connective are binary, we will also talk about **left-ancestor** and **right-ancestor**. The **syntactic forest** $T([\Sigma]\ \Gamma)$ of a cut sequent $[\Sigma]\ \Gamma$ is the union of the syntactic trees of the formulas in Γ and of the cut pairs in Σ.

An **additive resolution** of a cut sequent $[\Sigma]\ \Gamma$ is any result of taking the syntactic forest $T([\Sigma]\ \Gamma)$, then deleting zero or more cut pairs from Σ, and one argument sub-tree of each additive connective ($\&$ or $\oplus$). A **&-resolution** of a cut sequent $[\Sigma]\ \Gamma$ is any result of taking the syntactic forest $T([\Sigma]\ \Gamma)$, then deleting one argument sub-tree of each &-connective. For example, $[\,]\ X_1^+\&\ ,\ \oplus X_4^-$ is one of the eight additive resolutions of $[X_5^+ * X_6^-]\ X_1^+ \,\&\, X_2^-, X_3^+ \oplus X_4^-$, while $[X_5^+ * X_6^-]\ X_1^+\&\ , X_3^+ \oplus X_4^-$ is one of its two &-resolutions. Notice the difference on cut pairs: they may be deleted in an additive resolution, but never in a &-resolution.

An **axiom link** – or simply a **link** – on $[\Sigma]\ \Gamma$ is an unordered pair of complementary leaves in $T([\Sigma]\ \Gamma)$ – one being X^+ and the other X^- for some unsigned atom X. A **linking** λ on $[\Sigma]\ \Gamma$ is a set of links on $[\Sigma]\ \Gamma$ such that the sets of the leaves of its links form a partition of the set of

[2]One may or may not add a pending arc with source the root of the tree and with undefined target. We choose here not to.

leaves of an additive resolution of $[\Sigma]\ \Gamma$, additive resolution which is denoted $[\Sigma]\ \Gamma \upharpoonright \lambda$. We extend this notion to a set of linkings Λ by $[\Sigma]\ \Gamma \upharpoonright \Lambda = \bigcup_{\lambda\in\Lambda}[\Sigma]\ \Gamma \upharpoonright \lambda$. We also call **conclusions** of the set of linkings the cut pairs and formulas of the cut sequent it is on. For instance, on the left-most graph of Figure 4.1, the red axiom links form a linking $\lambda_1 = \{(X_1^+, X_6^-); (X_4^-, X_5^+)\}$, whose additive resolution is $[X_5^+ * X_6^-]\ X_1^+ \,\&\, X_2^-, X_3^+ \oplus X_4^- \upharpoonright \lambda_1 = [X_5^+ * X_6^-]\ X_1^+ \&\ ,\ \oplus X_4^-$. Note the relation between slices in the sequent calculus and linkings in proof-nets: a slice "belongs" to an additive resolution, and a &-resolution "selects" a slice from a proof.

Given a $⅋$, &, $\otimes$ or $\oplus$-vertex, its **premises** are its arguments, in other words the two vertices with arcs towards it in the syntactic forest. The **conclusion** of a vertex is the arc towards its descendant in the syntactic forest – if it is the root of a syntactic tree, then it has no conclusion. A set of linkings Λ on $[\Sigma]\ \Gamma$ **toggles** a &-vertex w if both premises of w are in its additive resolution $[\Sigma]\ \Gamma \upharpoonright \Lambda$. We say a link a **depends** on a &-vertex w in a set of linkings Λ if there exist $\lambda, \lambda' \in \Lambda$ such that $a \in \lambda\backslash\lambda'$ and w is the only &-vertex toggled by $\{\lambda; \lambda'\}$. Looking at our running example, and taking $\lambda_1 = \{(X_1^+, X_6^-); (X_4^-, X_5^+)\}$ and $\lambda_2 = \{(X_2^-, X_3^+)\}$, the &-vertex is toggled by $\{\lambda_1; \lambda_2\}$. Furthermore, all links depend on this &-vertex for λ_1 and λ_2 are disjoint (*i.e.* they contain only different links).

Given a set of linkings Λ on a cut sequent $[\Sigma]\ \Gamma$, the graph $\mathcal{G}_\Lambda$ is defined as $[\Sigma]\ \Gamma \upharpoonright \Lambda$ with the arcs from $\bigcup \Lambda$[3] and enriched with jump arcs $l \to w$ for each leaf l and each &-vertex w such that there exists $a \in \lambda \in \Lambda$, between l and some l', with a depending on w in Λ. When $\Lambda = \{\lambda\}$ is composed of a single linking, we shall simply denote $\mathcal{G}_\lambda = \mathcal{G}_{\{\lambda\}}$, which is the graph $[\Sigma]\ \Gamma \upharpoonright \lambda$ with the arcs from λ and no jump arc. About our example, the graphs $\mathcal{G}_{\lambda_1}$, $\mathcal{G}_{\lambda_2}$ and $\mathcal{G}_{\{\lambda_1;\lambda_2\}}$ are illustrated on Figure 4.1.

In the text of this paper (but not on the graphs), we write $l \xrightarrow{j} w$ for a jump arc from a leaf l to a &-vertex w. We will often not draw any jump arc on graphs of proof-nets, because they may make the figures less readable and they can be computed from the linkings. When drawing proof-nets, we will denote membership of a linking by means of colors, with each linking being identified by its color. Sometimes, a link belongs to several linkings, in which case we put a multicolored edge to signify belonging to linkings identified by these colors, or draw several edges, one by color. When considering only two linkings, we will furthermore draw the links of one above the vertices and the links of the other below. See Figure 4.1 for these conventions.

When we write a $⅋\backslash\&$-vertex, we mean a $⅋$- or &-vertex – a **negative** vertex; similarly a $\otimes\backslash\oplus$-vertex is a $\otimes$- or $\oplus$-vertex – a **positive** vertex. A **switch arc** of a $⅋\backslash\&$-vertex n is an incoming arc of n, *i.e.* an arc to n from one of its premises or a jump to n. A **switching cycle** (resp. **switching path**) is a cycle (resp. a simple path) with at most one switch arc of each $⅋\backslash\&$-vertex. A **$⅋$-switching** of a linking λ is any sub-graph of $\mathcal{G}_\lambda$ obtained by deleting a switch arc of each $⅋$-vertex; denoting by ϕ this choice of edges, the sub-graph it yields is $\mathcal{G}_\phi$. For example, a cycle in a $⅋$-switching is a switching cycle, as in a graph $\mathcal{G}_\phi$ all $⅋\backslash\&$-vertices have one premise and there is no jump.

Definition 4.1 (Proof-net)**.** A $\mathsf{MALL}_{uf}^{0,2}$ **proof-net with atomic-axioms** θ on a cut sequent $[\Sigma]\ \Gamma$ is a set of linkings satisfying:

(P0) *Cut:* Every cut pair of Σ has a leaf in θ.

(P1) *Resolution:* Exactly one linking of θ is on any given &-resolution of $[\Sigma]\ \Gamma$.

[3] To be completely coherent with our definition of graph from Chapter 3, we have to choose an orientation for these arcs: we arbitrarily decide to direct them from positive atoms to negative atoms. Still, we will not represent these orientations on our illustrations.

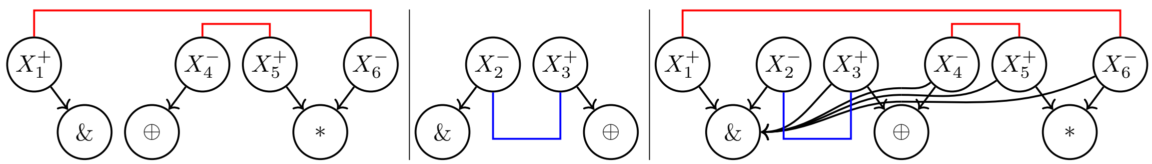


Figure 4.1: Example of a proof-net: from left to right $\mathcal{G}_{\lambda_1}$, $\mathcal{G}_{\lambda_2}$ and $\mathcal{G}_{\{\lambda_1;\lambda_2\}}$

(P2) *MLL:* For every linking $\lambda \in \theta$, $\mathcal{G}_\lambda$ has no switching cycle.

(P3) *Toggling:* Every set $\Lambda \subseteq \theta$ of two or more linkings toggles a &-vertex that is in no switching cycle of $\mathcal{G}_\Lambda$.

These conditions are called the **correctness criterion**. Condition (P0) is here to prevent unused $*$-vertices. A **cut-free** set of linkings is one without $*$-vertices, *i.e.* on a cut sequent $[\ ]\ \Gamma$; remark it respects (P0) trivially. Condition (P1) is a correctness criterion for ALL_{uf} proof-nets [HG05] and (P2) is the Danos-Regnier criterion for $\mathsf{MLL}_{uf}^{0,2}$ proof-nets [DR89]. However, (P1) and (P2) together are insufficient for cut-free $\mathsf{MALL}_{uf}^{0,2}$ proof-nets, hence the last condition (P3) taking into account interactions between the slices (see also [Di11] for a similar condition). Sets composed of a single linking λ are not considered in (P3), for by (P2) the graph $\mathcal{G}_\lambda$ has no switching cycle, and because there is no toggled &-vertex. We say that θ is a *proof-structure* if it satisfies (P0) and (P1). One can check that our example on Figure 4.1, $\{\lambda_1;\lambda_2\}$, is a proof-net.

Remark 4.2. Condition (P2) is equivalent to asking for every $⅋$-switching ϕ of every linking $\lambda \in \theta$, that $\mathcal{G}_\phi$ is acyclic. Indeed, a switching cycle in $\mathcal{G}_\lambda$ corresponds exactly to a cycle in some $\mathcal{G}_\phi$.

Without the mix_0- and mix_2-rules, *i.e.* in MALL_{uf}, the definition of proof-nets can be adapted with one more condition.

Definition 4.3 (Connected Proof-net)**.** A proof-net with atomic-axioms θ on a cut sequent $[\Sigma]\ \Gamma$ is **connected** if it satisfies:

($\mathrm{P2^c}$) *Connectivity:* For every $⅋$-switching ϕ of every linking $\lambda \in \theta$, $\mathcal{G}_\phi$ has exactly one connected component.

As a warning about terminology, a connected proof-net is not a proof-net θ whose graph $\mathcal{G}_\theta$ is connected. The condition ($\mathrm{P2^c}$) can be proved equivalent to the following one: for every linking $\lambda \in \theta$, for any two vertices v and u in $\mathcal{G}_\lambda$, there exists in $\mathcal{G}_\lambda$ a switching path from v to u. This is exploited for instance in [Ret03, Theorem 2]. Similarly, a definition can be given with only the mix_2-rule, *i.e.* for MALL_{uf}^2:

($\mathrm{P2^{c2}}$) For every $⅋$-switching ϕ of every linking $\lambda \in \theta$, $\mathcal{G}_\phi$ has at least one connected component.

We can do the same with only the mix_0-rule, for MALL_{uf}^0:

($\mathrm{P2^{c0}}$) For every $⅋$-switching ϕ of every linking $\lambda \in \theta$, $\mathcal{G}_\phi$ has at most one connected component.

Remark 4.4. Remark that an equivalent formulation of the conjunction of (P2) and ($\mathrm{P2^c}$), for MALL_{uf}, is the following: for every $⅋$-switching ϕ of every linking $\lambda \in \theta$, $\mathcal{G}_\phi$ is a tree. This definition was the one given in [HG05], among other equivalent formulations.

We thus have a hierarchy in four strata, where each level is included in the previous one:

1. set of linkings
2. proof-structure
3. proof-net
4. connected proof-net

While the correctness criterion may seem complex, the problem of checking whether a proof-structure is correct or not is NL-complete, at least for connected proof-nets [JM11]. In [HG05], a seemingly weaker property is conjectured equivalent to (P3).

Conjecture 4.5 (Single Switching Cycle Conjecture [HG05, Conjecture 4.22])**.** *The toggling condition* (P3) *is equivalent to:*

(P3$^-$) *For any set $\Lambda \subseteq \theta$ of two or more linkings and any switching cycle ω of $\mathcal{G}_\Lambda$, Λ toggles a &-vertex that is not in ω.*

We end this section by giving some more definitions we will need. A vertex is **terminal** if it is the root of a syntactic tree – it can be an $*$-vertex root of a cut pair. An additive $\oplus\backslash\&$-vertex is **unary** (resp. **binary**) in a set of linkings Λ if it has exactly one (resp. two) premises in $\mathcal{G}_\Lambda$. A **node** is a non-leaf vertex, and a proof-net is **node-free** if it contains only leaves.

When looking at a cut pair, or more generally at formulas A and $A^\perp$ both belonging to a cut sequent, one can define a notion of duality on leaves and connectives. For v a vertex in (the syntactic tree $T(A)$ of) A, we denote by $v^\perp$ the corresponding vertex in $A^\perp$. As expected, $v^{\perp\perp} = v$. This also respects orthogonality for formulas on leaves: given a leaf l of A, occurrence of a signed atom X, $l^\perp$ is an occurrence of $X^\perp$. We can also define a notion of duality on premises: given a premise of a vertex $v \in T(A)$, the dual premise of $v^\perp$ is the corresponding premise in $T(A^\perp)$. In other words, if in $l \to v \leftarrow r$ we consider the premise l then in $r^\perp \to v^\perp \leftarrow l^\perp$ its dual premise is $l^\perp$.

4.1.2 Cut-elimination in proof-nets

One can compose proof-nets using a cut pair, as well as eliminate a cut pair in a proof-net. Both operations preserve correctness, and contrary to what happens in sequent calculus cut-elimination is convergent.

Definition 4.6 (Composition)**.** For sets of linkings θ and ψ respectively on cut sequents $[\Sigma]\ \Gamma, A$ and $[\Xi]\ \Delta, A^\perp$, the **composition** over A of θ and ψ is the set of linkings $\theta \overset{A}{\bowtie} \psi = \{\lambda \cup \mu \mid \lambda \in \theta, \mu \in \psi\}$ on the cut sequent $[A * A^\perp, \Sigma, \Xi]\ \Gamma, \Delta$.

For example, see Figure 4.2 for a composition of two proof-nets. For a more complex example, see Figure 6.5 with a composition of the proof-nets on Figure 6.3, on Pages 224 and 225.

Lemma 4.7. *The composition of two proof-nets (resp. connected proof-nets, proof-structures) yields a proof-net (resp. a connected proof-net, a proof-structure).*

Proof. It suffices to check the correctness criterion is preserved by composition. Take θ and ψ respectively on $[\Sigma]\ \Gamma, A$ and $[\Xi]\ \Delta, A^\perp$.

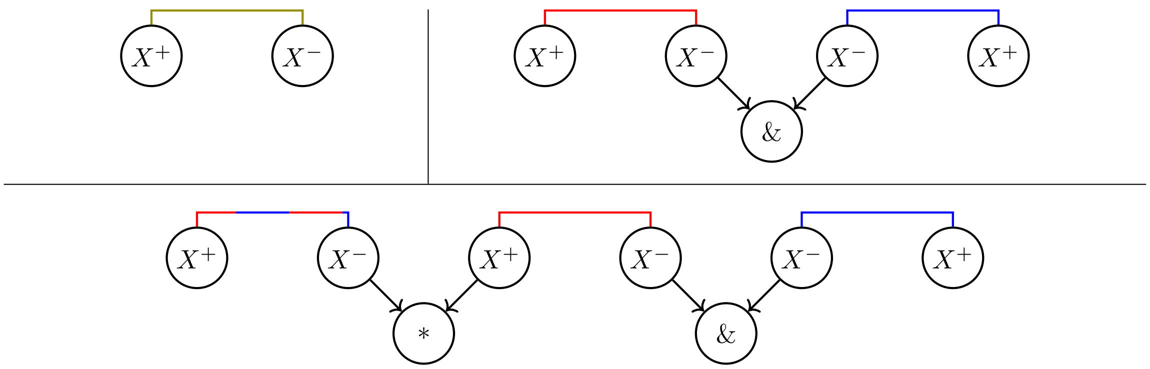


Figure 4.2: Example of composition by cut

If both θ and ψ respect (P1), then $\theta \overset{A}{\bowtie} \psi$ also respects it because a &-resolution of $[A * A^\perp, \Sigma, \Xi]\ \Gamma, \Delta$ corresponds exactly to one &-resolution of $[\Sigma]\ \Gamma, A$ and one of $[\Xi]\ \Delta, A^\perp$.

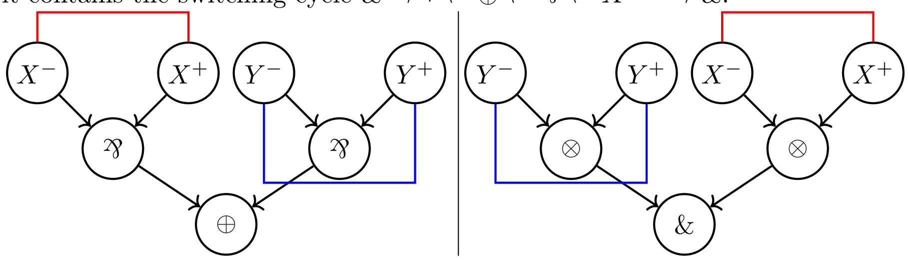
If θ and ψ are proof-structures, then $\theta \overset{A}{\bowtie} \psi$ respects (P0). Indeed, consider a cut pair in $A * A^\perp, \Sigma, \Xi$. Either this cut pair is in Σ, and we conclude using (P0) of θ, or it is in Ξ and we do similarly with (P0) of ψ, or the cut pair under consideration is $A * A^\perp$. In this last case, using (P1) of θ (resp. ψ), a leaf of A (resp. $A^\perp$) belongs to θ, as taking any &-resolution (there is at least one for we have a formula in the cut sequent) we get a linking on some additive resolution, and any additive resolution contains a leaf of A (resp. $A^\perp$). We thus find (P0) for $\theta \overset{A}{\bowtie} \psi$, by considering the union of the two previous linkings.

Preservation of (P2) is easy: a switching cycle in a linking $\lambda \cup \mu$ of $\theta \overset{A}{\bowtie} \psi$ cannot contain the $*$-vertex of $A * A^\perp$, which is the only path between the two syntactic forests – recall there is no jump arc in this case. Thus, a cycle can only be a cycle in $\mathcal{G}_\lambda$ or in $\mathcal{G}_\mu$, which cannot be switching. For the conservation of (P2$^\mathrm{c}$), by hypothesis one has a switching path in $\mathcal{G}_\lambda$ from a vertex v to the root of A, and a switching path in $\mathcal{G}_\mu$ from the root of $A^\perp$ to any vertex u. The path given by the concatenation of the first path with the two premises of the new $*$-vertex and then with the second path gives a switching path in $\mathcal{G}_{\lambda\cup\mu}$.

Finally, for (P3), consider a set of at least two linkings of $\Lambda \subseteq \theta \overset{A}{\bowtie} \psi$. Set Λ_θ those linkings in θ taken in Λ, *i.e.* $\Lambda_\theta = \{\lambda \mid \mu \in \Lambda \text{ and } \lambda \subset \mu\}$. Similarly, pose Λ_ψ those linkings in ψ taken in Λ. By definition, $\Lambda \subseteq \{\lambda \cup \mu \mid \lambda \in \Lambda_\theta, \mu \in \Lambda_\psi\}$, with Λ_θ or Λ_ψ of size at least two (for $\#|\Lambda| = \#|\Lambda_\theta| \times \#|\Lambda_\psi|$); *w.l.o.g.* say it is Λ_θ. Using (P3), one get a &-vertex w not in any switching cycle of Λ_θ. Remark there cannot be a jump arc between the vertices of $\mathcal{G}_{\Lambda_\theta}$ and $\mathcal{G}_{\Lambda_\psi}$ in $\mathcal{G}_{\theta \overset{A}{\bowtie} \psi}$. This is because if $\{\lambda \cup \mu; \lambda' \cup \mu'\}$ toggles a &, say from θ, then μ and μ' are on a same &-resolution of the cut sequent of ψ, so by (P1) $\mu = \mu'$ – in other words no axiom link coming from ψ depends on a &-vertex coming from θ, and symmetrically on ψ. Thus, cycles of $\mathcal{G}_\Lambda$ are either cycles of $\mathcal{G}_{\Lambda_\theta}$ or of $\mathcal{G}_{\Lambda_\psi}$, and therefore w is not in any switching cycle of $\mathcal{G}_\Lambda$. □

Remark 4.8. A proof of Lemma 4.7 can also be obtained from results not yet proven. Recognizing composition as the translation by desequentialization of the *cut*-rule, this is a corollary of Theorem 4.18, proven in [HG05]. Looking at the proof we give of Lemma 4.7, each part of the correctness criterion is preserved by the composition, except (P0) and (P3) which both need (P1). This hypothesis is needed for (P0): taking θ any set of linkings on Γ, A and the empty set on linkings $\emptyset$ on

$A^\perp$, both respects (P0) trivially, but $\theta \overset{A}{\bowtie} \psi = \emptyset$ on $[A * A^\perp]\ \Gamma$ does not. It is also needed for (P3): as an example, take on $(X^- \parr X^+) \oplus (Y^- \parr Y^+)$ the set $\theta = \{\{(X^-, X^+)\}; \{(Y^-, Y^+)\}\}$ and on $(Y^- \otimes Y^+) \mathbin{\&} (X^- \otimes X^+)$ the set $\psi = \{\{(X^-, X^+)\}; \{(Y^-, Y^+)\}\}$. The graph $\mathcal{G}_\theta$ is depicted below on the left, and $\mathcal{G}_\psi$ on the right. Then both sets of linkings respect (P3), but their composition does not: it contains the switching cycle $\& \to * \leftarrow \oplus \leftarrow \parr \leftarrow X^- \xrightarrow{j} \&$.



Definition 4.9 (Cut-elimination)**.** Let θ be a set of linkings on a cut sequent $[\Sigma]\ \Gamma$, and $A * A^\perp$ a cut pair in Σ. Define the *elimination* of $A * A^\perp$ (or of the cut $*$ between A and $A^\perp$) as the set of linkings obtained by:

(a) If A is an atom, delete $A * A^\perp$ from Σ and replace any pair of links (l, A), $(A^\perp, m)$ (l and m being other occurrences of $A^\perp$ and A respectively) with the link (l, m).

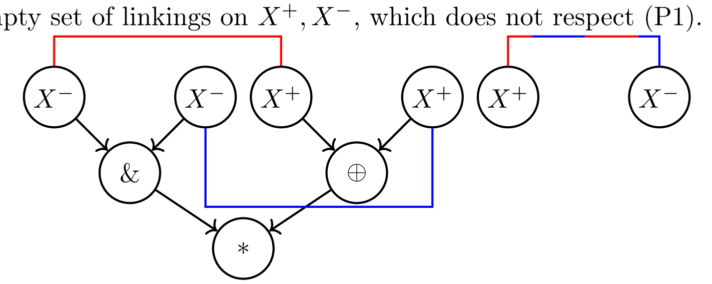
(b) If $A = A_1 \otimes A_2$ and $A^\perp = A_2^\perp \parr A_1^\perp$ (or vice-versa), replace $A * A^\perp$ with two cut pairs $A_1 * A_1^\perp$ and $A_2 * A_2^\perp$. Retain all original linkings.

(c) If $A = A_1 \mathbin{\&} A_2$ and $A^\perp = A_2^\perp \oplus A_1^\perp$ (or vice-versa), replace $A * A^\perp$ with two cut pairs $A_1 * A_1^\perp$ and $A_2 * A_2^\perp$. Delete all *inconsistent* linkings, namely those $\lambda \in \theta$ such that in $[\Sigma]\ \Gamma \upharpoonright \lambda$ the children $\&$ and $\oplus$ of the cut do not take dual premises. Finally, "garbage collect" by deleting any cut pair $B * B^\perp$ for which no leaf of $B * B^\perp$ is in any of the remaining linkings.

We use for proof-nets the same notations on cut-elimination as we did for the sequent calculus, namely $\xrightarrow{\beta}$ and $=_\beta$. An example of the (c) step of cut-elimination is illustrated on Figure 4.3. See Figure 6.6 for a more complex example, with a result of applying steps (b) and (c) to the proof-net of Figure 6.5.

Proposition 4.10 ([HG05, Proposition 5.4])**.** *Eliminating a cut in a proof-net (resp. connected proof-net) yields a proof-net (resp. connected proof-net).*

Remark 4.11. It is false that eliminating a cut in a proof-structure yields a proof-structure. As a counter-example, consider the proof-structure illustrated below: eliminating its only cut, with a (c) step, yields the empty set of linkings on X^+, X^-, which does not respect (P1).



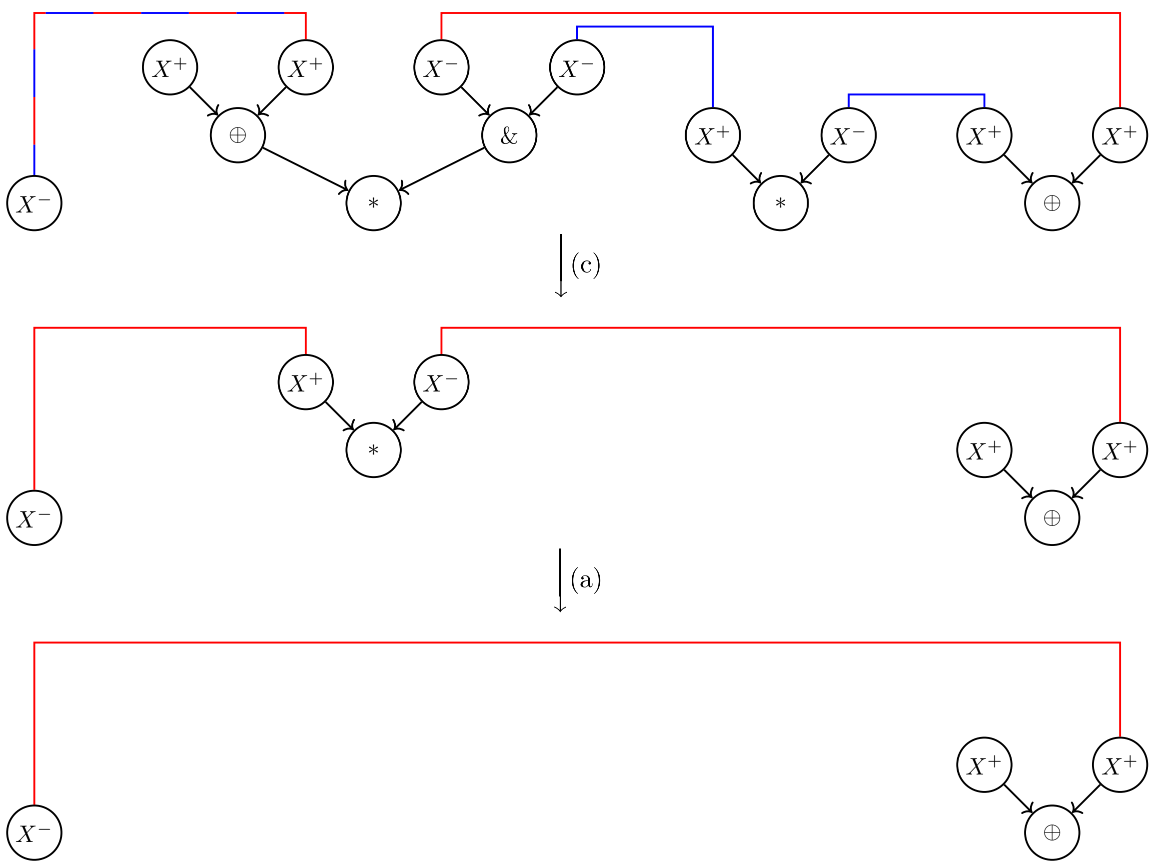


Figure 4.3: Example of cut-elimination steps

Theorem 4.12 ([HG05, Theorem 5.5]). *Cut-elimination of proof-nets is strongly normalizing and confluent.*

Remark 4.13. Other convergence properties can be proven as is proven Theorem 4.12 in [HG05], whose proof is basically that the only critical pair is two atom cases sharing a same link, with strong normalization being easy as a step remove vertices. We will use the two following results:

- eliminating all cuts using only a sub-set of the rules – for example not using the atomic case (a) – is confluent (and strongly normalizing as it is a sub-system of a strongly normalizing one);
- eliminating a given cut (and all cuts created this way) is confluent and strongly normalizing.

A linking λ on a cut sequent $[\Sigma]\ \Gamma$ **matches** if, for every cut pair $A * A^\perp$ in Σ, any given leaf l of A is in $[\Sigma]\ \Gamma \restriction \lambda$ if and only if $l^\perp$ of $A^\perp$ is in $[\Sigma]\ \Gamma \restriction \lambda$. A linking matches if and only if, when cut-elimination is carried out, the linking never becomes inconsistent, and thus is never deleted. Given linkings $\lambda \in \theta$ and $\mu \in \psi$, we say λ is **matching** for μ for a cut over B is $\lambda \cup \mu$ matches in the composition over B of θ and ψ. This allows defining **Turbo Cut-elimination** in [HG05], eliminating a cut in a single step by removing inconsistent linkings.

Definition 4.14 (Turbo Cut-elimination). Take θ a proof-net on a cut sequent $[\Sigma]\ \Gamma$. Delete all non-matching linkings of θ and, for a matching linking λ, replace every links $(l_0, l_1), ((l_i^\perp, l_{i+1}))_{i \in [\![1;n-1]\!]}$ by (l_0, l_n), where in these leaves only l_0 and l_n belong to Γ. The resulting proof-net on the cut sequent $[\ ]\ \Gamma$ is the normal form of θ for cut-elimination.

Remark 4.15. It is possible to modify the definition of Turbo Cut-elimination to the case where one eliminates only some cut pairs of Σ, which is a convergent rewriting system (Remark 4.13). It suffices to restrict matching for a set of cut pairs, and to adapt the definition in this setting.

4.2 From Sequent Calculus to Proof-nets

Here we establish a translation from proofs of sequent calculus to proof-nets. We first define *desequentialization*, a function from sequent calculus proofs to proof-nets (Section 4.2.1). We then show that cut-elimination in proof-nets simulates the one from sequent calculus (Section 4.2.2). Recall that all sequent calculus proofs we consider can be assumed atomic-axiom, thanks to Proposition 2.10.

4.2.1 Desequentialization

We desequentialize a $\mathsf{MALL}_{uf}^{0,2}$ atomic-axiom proof π of $\vdash \Gamma$ into a set of linkings by induction on π using the steps detailed on Figure 4.4, following [HG05] with the notation $\theta \triangleright [\Sigma]\ \Gamma$ for "θ is a set of linkings on the cut sequent $[\Sigma]\ \Gamma$". This is called the **desequentialization** $\mathfrak{P}$.

As identified in [HG05, Section 5.3.4], desequentializing with both *cut*- and &-rules is complex, for cut pairs can be shared (or not) when translating a &-rule: one has the choice to put a cut-formula appearing in both premises either in Σ, or in both Φ and Ψ. Therefore, the translation is non-deterministic; in other words, a proof desequentializes not to a set of linkings, but to a set $\mathfrak{P}(\pi)$ of those objects – *i.e.* $\mathfrak{P}(\pi)$ is a set of sets of linkings, which all are on different cut sequents, for some cut pairs can be duplicated in the cut sequent of some and not in the cut sequent of

$$\frac{}{\vdash \{\{(X^+, X^-\}\} \triangleright [\,]\ X^+, X^-}\ (ax)$$

$$\frac{\vdash \theta \triangleright [\Sigma]\ A, \Gamma \qquad \vdash \vartheta \triangleright [\Phi]\ A^\perp, \Pi}{\vdash \{\lambda \cup \mu \mid \lambda \in \theta, \mu \in \vartheta\} \triangleright [A * A^\perp, \Sigma, \Phi]\ \Gamma, \Pi}\ (cut)$$

$$\frac{\vdash \theta \triangleright [\Sigma]\ A, \Gamma \qquad \vdash \vartheta \triangleright [\Phi]\ B, \Pi}{\vdash \{\lambda \cup \mu \mid \lambda \in \theta, \mu \in \vartheta\} \triangleright [\Sigma, \Phi]\ A \otimes B, \Gamma, \Pi}\ (\otimes) \qquad \frac{\vdash \theta \triangleright [\Sigma]\ A, B, \Gamma}{\vdash \theta \triangleright [\Sigma]\ A \parr B, \Gamma}\ (\parr)$$

$$\frac{\vdash \theta \triangleright [\Sigma, \Phi]\ A, \Gamma \qquad \vdash \vartheta \triangleright [\Sigma, \Psi]\ B, \Gamma}{\vdash \theta \cup \vartheta \triangleright [\Sigma, \Phi, \Psi]\ A \,\&\, B, \Gamma}\ (\&)$$

$$\frac{\vdash \theta \triangleright [\Sigma]\ A, \Gamma}{\vdash \theta \triangleright [\Sigma]\ A \oplus B, \Gamma}\ (\oplus_1) \qquad \frac{\vdash \theta \triangleright [\Sigma]\ B, \Gamma}{\vdash \theta \triangleright [\Sigma]\ A \oplus B, \Gamma}\ (\oplus_2)$$

$$\frac{\vdash \theta \triangleright [\Sigma]\ \Gamma \qquad \vdash \vartheta \triangleright [\Phi]\ \Pi}{\vdash \{\lambda \cup \mu \mid \lambda \in \theta, \mu \in \vartheta\} \triangleright [\Sigma, \Phi]\ \Gamma, \Pi}\ (mix_2) \qquad \frac{}{\vdash \{\emptyset\} \triangleright [\,]}\ (mix_0)$$

We use the implicit tracking of formula occurrences downwards through the rules.

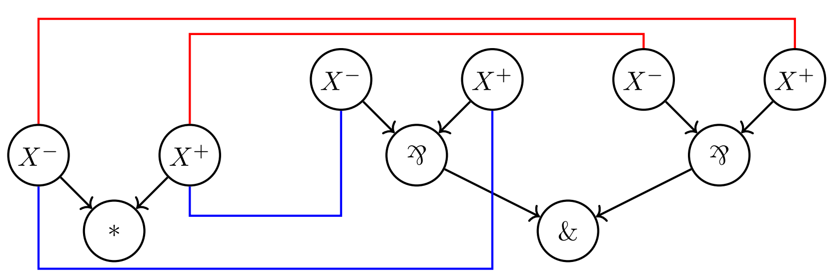
Figure 4.4: Inductive definition of the desequentialization $\mathfrak{P}$ of $\mathsf{MALL}^{0,2}_{uf}$ atomic-axiom proofs to sets of linkings

others. One can force $\Sigma = \emptyset$ to recover a function $\mathfrak{P}_f$, but then the $\& - cut$ rule commutation yields different proof-nets (contrary to the other commutations, see [HG16]).

$$\frac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \qquad \overset{\pi_2}{\vdash A, C, \Gamma}}{\vdash A, B \,\&\, C, \Gamma}\ (\&) \qquad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash B \,\&\, C, \Gamma, \Delta}\ (cut) \quad \overset{rc}{\mapsto} \quad \frac{\dfrac{\overset{\pi_1}{\vdash A, B, \Gamma} \qquad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash B, \Gamma, \Delta}\ (cut) \qquad \dfrac{\overset{\pi_2}{\vdash A, C, \Gamma} \qquad \overset{\pi_3}{\vdash A^\perp, \Delta}}{\vdash C, \Gamma, \Delta}\ (cut)}{\vdash B \,\&\, C, \Gamma, \Delta}\ (\&)$$

See Figure 4.5 illustrating the differences between the obtained sets of linkings.

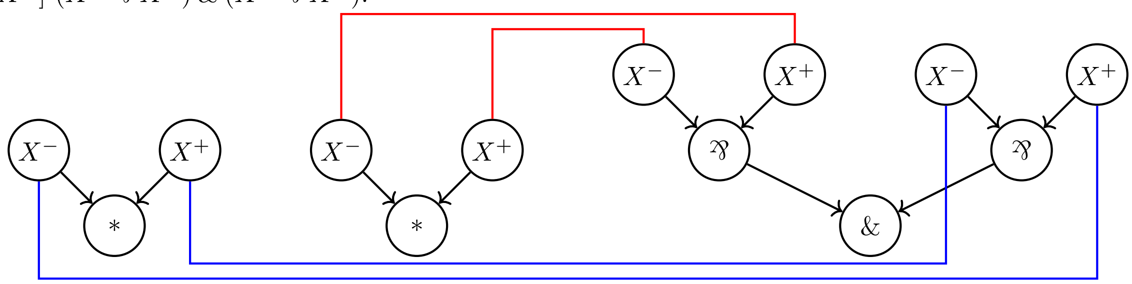
Another problem, not evoked explicitly in [HG05], is that given a proof-net θ, it belongs to $\mathfrak{P}(\pi)$ for some proof π, but there may not be any proof π such that $\mathfrak{P}_f(\pi) = \theta$. As a counter-example, consider the following proof-net θ, on the cut sequent $[X^- * X^+]\ (X^- \parr X^+) \,\&\, (X^- \parr X^+)$:



The only proof π such that $\theta \in \mathfrak{P}(\pi)$ is the following:

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax) \quad \dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, X^+}\,(cut)}{\vdash X^- \parr X^+}\,(\parr) \qquad \dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax) \quad \dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, X^+}\,(cut)}{\vdash X^- \parr X^+}\,(\parr)}{\vdash (X^- \parr X^+) \,\&\, (X^- \parr X^+)}\,(\&)$$

However, $\pi \neq \mathfrak{P}_f(\pi)$, as $\mathfrak{P}_f(\pi)$ is the other proof-net in $\mathfrak{P}(\pi)$, on the cut sequent $[X^- * X^+, X^- * X^+]\ (X^- \parr X^+) \,\&\, (X^- \parr X^+)$:



Nonetheless, given a proof π we often want any proof-net corresponding to it, as we will not care about canonicity of proof-nets with cut. We can thus arbitrary choose to consider the set of linkings $\mathfrak{P}_f(\pi)$ with the choice $\Sigma = \emptyset$ in the translation of all &-rules. Obviously, given a proof π without both &- and *cut*-rules, $\mathfrak{P}(\pi) = \{\mathfrak{P}_f(\pi)\}$.

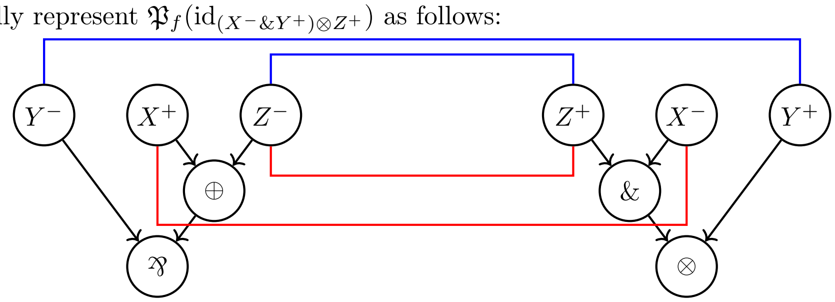
As a last example, consider the proof $\mathrm{id}_{(X^- \& Y^+) \otimes Z^+}$:

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{\vdash X^-, X^+}\,(ax)}{\vdash X^-, Y^- \oplus X^+}\,(\oplus_2) \quad \dfrac{\dfrac{}{\vdash Y^-, Y^+}\,(ax)}{\vdash Y^+, Y^- \oplus X^+}\,(\oplus_1)}{\vdash X^- \,\&\, Y^+, Y^- \oplus X^+}\,(\&) \quad \dfrac{}{\vdash Z^-, Z^+}\,(ax)}{\vdash (X^- \,\&\, Y^+) \otimes Z^+, Y^- \oplus X^+, Z^-}\,(\otimes)}{\vdash (X^- \,\&\, Y^+) \otimes Z^+, Z^- \parr (Y^- \oplus X^+)}\,(\parr)$$

Its desequentialization is unique and is the following set of two linkings:

$$\mathfrak{P}_f(\mathrm{id}_{(X^- \& Y^+) \otimes Z^+}) = \{\{(X^-, X^+); (Z^-, Z^+)\}; \{(Y^-, Y^+); (Z^-, Z^+)\}\}$$

We graphically represent $\mathfrak{P}_f(\mathrm{id}_{(X^- \& Y^+) \otimes Z^+})$ as follows:



Remark 4.16. An alternative definition of desequentialization in [HG05] consists in building a linking by slice. Given a cut-free proof π on a sequent Γ, each slice $s \in \mathcal{S}(\pi)$ of π gives a set of *ax*-rules defining a linking λ_s on Γ. We have $\mathfrak{P}_f(\pi) = \{\lambda_s \mid s \in \mathcal{S}(\pi)\}$. In this spirit, if a cut-free proof-net θ is obtained by desequentializing a proof π, there is a bijection between linkings in θ and slices of π.

In the presence of *cut*-rules, one has to compute first a cut sequent associated to π, with the very same problem in the &-rule about separating or superimposing cut pairs. Once this is done, the sets of linkings $\{\lambda_s \mid s \in \mathcal{S}(\pi)\}$, for all possible choices of cut sequents, yield $\mathfrak{P}(\pi)$.

Fact 4.17. *Consider a proof π and a proof-net θ such that $\theta \in \mathfrak{P}(\pi)$. Then π is cut-free if and only if θ is.*

One can easily prove that all elements of $\mathfrak{P}(\pi)$ are proof-nets. The reverse, that any proof-net belongs to $\mathfrak{P}(\pi)$ for some proof π, is harder and is the theorem with the longest proof in [HG05].

Theorem 4.18 (Sequentialization, [HG05, Theorem 5.9])**.** *A set of linkings on a cut sequent is a translation of an atomic-axiom $\mathsf{MALL}_{uf}^{0,2}$ proof if and only if it is a proof-net.*

A simpler proof of the hard direction of this theorem will be given in Section 4.3, using results from Chapter 3. The following theorem is a major result, proving that these proof-nets indeed quotient proofs as wished.

Theorem 4.19 ([HG16, Theorem 1])**.** *Two atomic-axiom cut-free MALL_{uf} proofs desequentialize to the same proof-net if and only if they can be converted into each other by a series of rule commutations.*

In the same paper, similar theorems are given with the commutations also involving the *cut*- and mix_2-rules (*i.e.* for $\overset{rc}{\longmapsto}$ in MALL_{uf}^{2}), except for the $\& - cut$ commutation. In presence of both mix_0- and mix_2-rules, namely in $\mathsf{MALL}_{uf}^{0,2}$, proof-nets quotient proofs not only by rule commutation $\overset{r^*}{\longmapsto}$, but also by the *mix*-Rétoré transformation $\overset{om}{\rightsquigarrow}$. Therefore, thanks to Theorems 2.50 and 4.19, a cut-free proof-net (resp. connected proof-net) is a canonical representation up to $=_{\beta\eta}$ of a proof in MALL_{uf}^{2} (resp. MALL_{uf}).

4.2.2 Simulation of cut-elimination

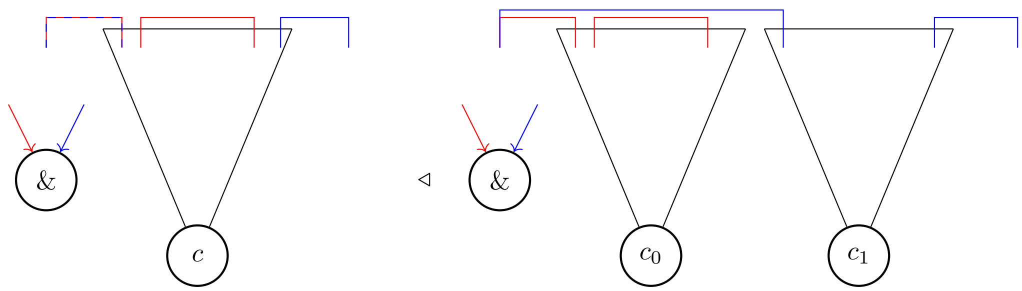
We show here that cut elimination in proof-nets mimics the one in sequent calculus, so that both concepts are strongly related. As written in Section 4.2.1, proof-nets have difficulties with the $\& - cut$ commutation, which corresponds to superimposing $*$-vertices.

Definition 4.20 ($\triangleleft$)**.** Let θ and ψ be sets of linkings. We denote $\theta \triangleleft \psi$ if there exists a $*$-vertex c in θ such that the syntactic forest of ψ is the syntactic forest of θ where the syntactic tree of c is duplicated into the syntactic trees of c_0 and c_1 (which are different occurrences of c), $\theta = \theta_0 \sqcup \theta_1$[4] and $\psi = \psi_0 \sqcup \psi_1$ with, for $i \in \{0;1\}$, $\psi_i = \theta_i$ up to assimilating c_i with c.

See Figure 4.5 for a graphical representation of this concept, as well as the link with the $\& - cut$ commutative case of cut-elimination.

Lemma 4.21 (Simulation - β)**.** *Let π and π' be $\mathsf{MALL}_{uf}^{0,2}$ proofs such that $\pi \xrightarrow{\beta} \pi'$. Then either $\mathfrak{P}_f(\pi) = \mathfrak{P}_f(\pi')$, $\mathfrak{P}_f(\pi) \triangleleft \mathfrak{P}_f(\pi')$ or $\mathfrak{P}_f(\pi) \xrightarrow{\beta} \mathfrak{P}_f(\pi')$.*

[4] The symbol $\sqcup$ means a union $\cup$ which happens to be between disjoint sets.



$$\dfrac{\dfrac{\vdash A, B, \Gamma \qquad \vdash A, D, \Gamma}{\vdash A, B \mathbin{\&} D, \Gamma}\,(\&) \qquad \vdash A^\perp, \Delta}{\vdash B \mathbin{\&} D, \Gamma, \Delta}\,c \quad \xrightarrow{\beta} \quad \dfrac{\dfrac{\vdash A, B, \Gamma \qquad \vdash A^\perp, \Delta}{\vdash B, \Gamma, \Delta}\,c_0 \qquad \dfrac{\vdash A, D, \Gamma \qquad \vdash A^\perp, \Delta}{\vdash D, \Gamma, \Delta}\,c_1}{\vdash B \mathbin{\&} D, \Gamma, \Delta}\,(\&)$$

Figure 4.5: Illustration of $\lhd$ and correspondence with the $\& - cut$ commutation

Proof. We reason by cases according to the step $\pi \xrightarrow{\beta} \pi'$. Recall that in $\mathfrak{P}_f$ we desequentialize by separating all cuts, and use the notations for steps from Definition 4.9. If $\pi \xrightarrow{\beta} \pi'$ is an ax (resp. $\parr - \otimes$, $\& - \oplus$) key case, then using a step (a) (resp. (b), (c)), we get $\mathfrak{P}_f(\pi) \xrightarrow{\beta} \mathfrak{P}_f(\pi')$. If it is a $\parr - cut$, $\otimes - cut - 1$, $\otimes - cut - 2$, $\oplus_1 - cut$, $\oplus_2 - cut$ or $mix_2 - cut$ commutative case, then $\mathfrak{P}_f(\pi) = \mathfrak{P}_f(\pi')$. Finally, in a $\& - cut$ commutative case, we duplicate the cut-rule: $\mathfrak{P}_f(\pi) \lhd \mathfrak{P}_f(\pi')$ (see Figure 4.5). $\square$

Nonetheless, the $\lhd$ relation is not really problematic since two proofs differing by a $\& - cut$ commutation yield proof-nets equal up to cut-elimination.

Lemma 4.22 ($\lhd \subseteq =_\beta$)**.** *Let θ and ψ be sets of linkings such that $\theta \lhd \psi$. Then $\theta =_\beta \psi$.*

Proof. By Definition 4.20 of $\lhd$, there exists a $*$-vertex c in θ, with $\theta = \theta_0 \sqcup \theta_1$, such that ψ is θ where the syntactic tree of c is duplicated into c_0 and c_1, and linkings in θ_0 (respectively θ_1) use c_0 (respectively c_1) as c.

We reason by induction on the size of the formula A of c (and also c_0 and c_1); *w.l.o.g.* A is positive. Applying a step of cut-elimination on c in θ yields a set of linkings θ'. On the other hand, in ψ a corresponding step of cut-elimination on c_0 then one on c_1 (if it has not been garbage collected by the previous cut-elimination step) yields ψ'.

If A is an atom, then we applied step (a), and we find $\theta' = \psi'$. Hence, $\theta \xrightarrow{\beta} \theta' = \psi' \xleftarrow{\beta+} \psi$.

If A is a $\otimes$-formula, *i.e.* $A = A_0 \otimes A_1$, then we applied step (b) and produced cuts $A_0 * A_0^\perp$ and $A_1 * A_1^\perp$ in θ', and (at most) two occurrences of these cuts in ψ'. Thus, $\theta' \lhd \zeta \lhd \psi'$ with ζ the set of linkings θ' where the cut on A_0 is duplicated (or $\theta' \lhd \psi'$ in case of garbage collection). By induction hypothesis, $\theta' =_\beta \zeta =_\beta \psi'$. It follows $\theta =_\beta \theta'$ as $\theta \xrightarrow{\beta} \theta' =_\beta \psi' \xleftarrow{\beta+} \psi$.

Finally, if A is a $\oplus$-formula with $A = A_0 \oplus A_1$, then we used step (c), producing cuts $A_0 * A_0^\perp$ and $A_1 * A_1^\perp$ in θ', and (at most) two occurrences of these cuts in ψ'. Remark that inconsistent linkings in ψ for these steps are exactly those of θ, and therefore the same cuts are garbage collected. Whence $\theta' \lhd \cdot \lhd \psi'$, $\theta' \lhd \psi'$ or $\theta' = \psi'$, according to the number of cuts garbage collected. In all cases, using the induction hypothesis we conclude $\theta =_\beta \psi$. $\square$

Remark 4.23. Another proof of Lemma 4.22, using the Turbo Cut-elimination procedure and no induction, is possible. We use the Turbo Cut-elimination procedure on c in θ, yielding a set of linkings θ'; we also use it in ψ on c_0 then on c_1, yielding ψ'. Whence, $\theta \xrightarrow{\beta^*} \theta'$ and $\psi' \xleftarrow{\beta^*} \psi$. It remains to prove that $\theta' = \psi'$. Remark that θ' and ψ' can only differ by their linkings, for they have the same syntactic forest. Notice that a linking in θ_i, $i \in \{0;1\}$, matches for c in θ if and only if it matches for c_i in ψ, because this linking uses c_i as c. Thence, the same linkings stay in θ' and ψ', and $\theta' = \psi'$ follows.

Theorem 4.24 (Simulation Theorem)**.** *Let π and π' be $\mathsf{MALL}^{0,2}_{uf}$ atomic-axiom proofs. If $\pi =_\beta \pi'$, then $\mathfrak{P}_f(\pi) =_\beta \mathfrak{P}_f(\pi')$.*

Proof. This is a corollary of Lemmas 4.21 and 4.22. □

We call $\mathcal{B}(\theta)$ the β-normal form of the proof-net θ, with all cuts eliminated (thanks to Proposition 4.10 and Theorem 4.12).

Lemma 4.25. *Let π, τ and ρ be $\mathsf{MALL}^{0,2}_{uf}$ atomic-axiom proofs of respective sequents $\vdash A, \Gamma$, $\vdash A^\perp, \Delta$ and $\vdash \Gamma, \Delta$. Assume $\pi \overset{A}{\bowtie} \tau =_\beta \rho$. Then $\mathcal{B}(\mathfrak{P}_f(\pi)) \overset{A}{\bowtie} \mathcal{B}(\mathfrak{P}_f(\tau))$ reduces, after fully eliminating the cut on A, to $\mathcal{B}(\mathfrak{P}_f(\rho))$.*

Proof. By the Simulation Theorem (Theorem 4.24), $\mathfrak{P}_f(\pi \overset{A}{\bowtie} \tau) =_\beta \mathfrak{P}_f(\rho)$. By definition of $\mathfrak{P}_f$ (Section 6.3.1), $\mathfrak{P}_f(\pi \overset{A}{\bowtie} \tau) = \mathfrak{P}_f(\pi) \overset{A}{\bowtie} \mathfrak{P}_f(\tau)$, so $\mathcal{B}(\mathfrak{P}_f(\pi) \overset{A}{\bowtie} \mathfrak{P}_f(\tau)) = \mathcal{B}(\mathfrak{P}_f(\rho))$. Moreover, by confluence of cut-elimination (Theorem 4.12), $\mathcal{B}(\mathfrak{P}_f(\pi) \overset{A}{\bowtie} \mathfrak{P}_f(\tau))$ can be obtained by taking the β-normal forms of $\mathfrak{P}_f(\pi)$ and $\mathfrak{P}_f(\tau)$, composing them over A and reducing this cut. In other words:

$$\mathcal{B}(\mathcal{B}(\mathfrak{P}_f(\pi)) \overset{A}{\bowtie} \mathcal{B}(\mathfrak{P}_f(\tau))) = \mathcal{B}(\mathfrak{P}_f(\pi) \overset{A}{\bowtie} \mathfrak{P}_f(\tau)) = \mathcal{B}(\mathfrak{P}_f(\rho))$$

The result then follows. □

4.3 Sequentialization

We give here a new proof of sequentialization for the proof-nets defined in Section 4.1.1. At the best of my knowledge, this is the only proof of sequentialization other than the original one from [HG05]. Our proof uses Theorem 3.19, as this setting is the one where some switching cycles are allowed, corresponding to bridge-free cycles in Chapter 3. If we were to work for $\mathsf{MLL}^{0,2}_{uf}$ proof-nets only, then we could have applied the simpler Theorem 3.29.

For the intuition, a switching cycle can be seen as a zone of dependency where the order of sequentialization may depend on slices. We have to find a jump arc from these cycles to a &-vertex which causes this dependency, across all slices. The parallel in Theorem 3.19 are the $\mathfrak{e}(\Omega)$, allowing to go out of a union of cycles Ω.

We choose to give a proof finding a terminal splitting vertex, as in [HG05] the proof proceeds by means of splitting $\parr\backslash\&$-vertex. Remark that, thanks to Theorem 3.19, we could also have done a proof by splitting $\parr\backslash\&$ – this is detailed in Section 4.3.6.

4.3.1 Key lemma

We use a basic result from [HG05], whose proof is reproduced here with some adaptations. It is the only use of the correctness criterion we make to prove sequentialization, and its role is to prove the hypotheses (H_G^0) and (H_G^1) for applying Theorem 3.19.

A sub-set Λ of a set of linkings θ is **saturated** if any strictly larger sub-set of θ toggles more &-vertices than Λ. Clearly, θ itself is saturated. For Λ a set of linkings and w a &-vertex, let Λ^w denote the set of all linkings in Λ whose additive resolution does not contain the right argument of w. In this definition – taken from [HG05] – we could have chosen left argument instead of right argument, meaning the asymmetricity is irrelevant. Write $\lambda \overset{w}{=} \lambda'$ if linkings $\lambda, \lambda' \in \theta$ are either equal or w is the only & toggled by $\{\lambda, \lambda'\}$. Saturated sets of linkings in a proof-structure have the following properties – those are given (without a proof) in [HG05].

Lemma 4.26. *In a set of linkings θ respecting* (P1)*, take $\Lambda \subseteq \theta$.*

(S1) *If Λ is saturated and toggles w, then Λ^w is saturated.*

(S2) *If Λ is saturated and toggles w and $\lambda \in \Lambda$, then $\lambda \overset{w}{=} \lambda_w$ for some $\lambda_w \in \Lambda^w$.*

(S3) *If Λ is saturated, toggles w and $\lambda \overset{x}{=} \lambda'$ for $\lambda, \lambda' \in \Lambda$, then for some $\lambda_w, \lambda'_w \in \Lambda^w$ we have:*

$$\begin{array}{ccc} \lambda & \overset{x}{=\!=} & \lambda' \\ w\| & & \|w \\ \lambda_w & \overset{x}{=\!=} & \lambda'_w \end{array}$$

Proof.

(S1) Assume Λ is saturated and toggles w. Remark Λ^w does not toggle w and is not empty, because w is toggled by Λ: set λ an arbitrary linking of Λ^w. Take Λ' a strictly larger subset of θ than Λ^w, and assume by contradiction it toggles no more &-vertices than Λ^w. Consider $\Lambda \cup \Lambda'$. This is a strictly bigger subset of θ than Λ. Indeed, if we had a linking $\lambda' \in \Lambda' \cap (\Lambda \backslash \Lambda^w)$, then $\{\lambda; \lambda'\} \subseteq \Lambda'$ would toggle w, contradicting that Λ' does not toggle more &s than Λ^w. By saturation, $\Lambda \cup \Lambda'$ toggles (at least) one more & than Λ; call x such a &. As Λ does not toggle x, say its additive resolution does not contain the left premise of x. There exists $\lambda' \in \Lambda'$ whose additive resolution has the left premise of x. But then $\{\lambda; \lambda'\} \subseteq \Lambda'$ toggles x, which is not toggled by Λ hence by Λ^w. Contradiction, hence Λ^w is saturated.

(S2) Assume Λ is saturated and toggles w, and take $\lambda \in \Lambda$. If $\lambda \in \Lambda^w$, then we are done with $\lambda_w = \lambda$. Otherwise, λ has the right premise of w in its additive resolution. Set R a &-resolution λ is on, build from the additive resolution of λ and keeping if possible for &s not inside a premise belonging to the additive resolution of Λ. Pose S this resolution where w instead keeps its left argument – giving for &-vertices introduced this way choices of premises which appear in the additive resolution of Λ if possible. By (P1), there is a linking λ_w on S. One have $\lambda \overset{w}{=} \lambda_w$. Indeed, either $\{\lambda; \lambda_w\}$ toggles only w by looking at &-resolutions there are on, or one of the two does not have w in its additive resolution, so is on both R and S, so that $\lambda = \lambda_w$ thanks to (P1). It remains to prove $\lambda_w \in \Lambda$. By contradiction, assume $\Lambda \cup \lambda_w$ is strictly bigger than Λ. By saturation, it must toggles a &-vertex x not toggled by Λ. Remark x belongs to the additive resolution of Λ. This x must be w, for otherwise its premise kept in S, so by λ_w, is already in the additive resolution of Λ. But w is toggled by Λ, reaching a contradiction.

(S3) Take a saturated Λ which toggles w and $\lambda \stackrel{x}{=} \lambda'$ for $\lambda, \lambda' \in \Lambda$. If $\lambda = \lambda'$, then we can trivially conclude through (S2), finding $\lambda_w = \lambda'_w \in \Lambda^w$ such that $\lambda \stackrel{w}{=} \lambda_w$. Thus, assume $\{\lambda; \lambda'\}$ toggles only x. Set R and R' two &-resolutions such that λ (resp. λ') is on R (resp. R') and R and R' differ only on x and its ancestors; this can be done by taking for R premises in the additive resolution of λ for those &s inside, otherwise taking the same premises as those in the additive resolution of λ' for those inside, and finally for &s appearing in neither to keep as premise one kept in the additive resolution of Λ if possible (and similarly for w'). Define S (resp. S') as R (resp. R') by choosing the other premise for w, and taking for &s introduced this way premises kept in the additive resolution of Λ if possible – and doing the same choices in both S and S'. By (P1), there are linkings λ_w and λ'_w respectively on S and S'. Mimicking the proof of (S2), we find $\lambda_w, \lambda'_w \in \Lambda$ as this set is saturated. We conclude as $\lambda \stackrel{w}{=} \lambda_w \stackrel{x}{=} \lambda'_w \stackrel{w}{=} \lambda'$, looking at the &-resolutions there are on – proceeding as in the proof of (S2) again. ☐

We aim to prove Corollary 4.29. As we will need a very similar result for the study of isomorphisms in Chapter 6, we prove a slight generalization.

Lemma 4.27. *Let Λ and Λ' be sub-sets of a set of linkings θ. Take ω a cycle such that $\omega \subseteq \mathcal{G}_\Lambda$ but $\omega \nsubseteq \mathcal{G}_{\Lambda'}$. Then there exists an arc $a \in \omega$ which is either an axiom link or a jump arc, such that $a \in \mathcal{G}_\Lambda \backslash \mathcal{G}_{\Lambda'}$.*

Proof. Since $\omega \nsubseteq \mathcal{G}_{\Lambda'}$, some arc a of ω is in $\mathcal{G}_\Lambda$ but not in $\mathcal{G}_{\Lambda'}$. We prove our result without loss of generality, by reducing step by step other possibilities to the wished one.

Without loss of generality, a is an arc from or to a leaf l. Indeed, for any other arc $u \to v$ in ω we have $l \to x_1 \to \cdots \to x_n \to u \to v$ in ω for some leaf l, because arcs not using leafs are in the syntactic forest of the cut sequent. Moreover $u \to v$ is in $\mathcal{G}_{\Lambda'}$ whenever $l \to x_1$ is in $\mathcal{G}_{\Lambda'}$.

Still without loss of generality, a is not an edge in the syntactic forest. Indeed, in such a case $a \notin \mathcal{G}_{\Lambda'}$ implies $l \notin \mathcal{G}_{\Lambda'}$. As l belongs to the cycle ω, let us look at the other arc in this cycle with endpoint l, say a'. As $l \notin \mathcal{G}_{\Lambda'}$, we also have $a' \notin \mathcal{G}_{\Lambda'}$. Remark that a' cannot be an arc in a syntactic tree, for only one such arc has for endpoint the leaf l, namely a. We thus consider a' instead of a. ☐

Lemma 4.28. *Given a proof-net θ, take a non-empty union Ω of switching cycles of $\mathcal{G}_\theta$. Set Λ a minimal saturated subset of θ with $\mathcal{G}_\Lambda$ containing Ω.*

There exists a &-vertex w toggled by Λ that is not in Ω and a leaf $l \in \Omega$, such that there is a jump $l \to w$ in $\mathcal{G}_\Lambda$, with l being in a link of Ω depending on w or being in a jump arc of Ω. Furthermore, $\Omega \nsubseteq \mathcal{G}_{\Lambda^w}$.

Proof. By (P2), singleton sub-sets of θ are acyclic, so Λ contains at least two linkings. Let w be a & toggled by Λ that is not in any switching cycle of $\mathcal{G}_\Lambda$, existing by (P3), so $w \notin \Omega$. Since Λ is minimal, $\Omega \nsubseteq \mathcal{G}_{\Lambda^w}$ using (S1), hence by Lemma 4.27 some axiom link or a jump arc e of Ω is in $\mathcal{G}_\Lambda$ but not in $\mathcal{G}_{\Lambda^w}$. Call l a leaf of this arc.

If e is an axiom link, take $\lambda \in \Lambda$ such that $e \in \lambda$. By (S2) $\lambda \stackrel{w}{=} \lambda_w$ for some $\lambda_w \in \Lambda^w$. Since $e \notin \lambda_w$ (for $e \notin \mathcal{G}_{\Lambda^w}$), the jump $l \to w$ is in $\mathcal{G}_\Lambda$.

If e is a jump to a &-vertex x, we have $\lambda, \lambda' \in \Lambda$ with $a \in \lambda$, $a \notin \lambda'$, l a leaf of a, and $\lambda \stackrel{x}{=} \lambda'$. By (S3), $\lambda \stackrel{w}{=} \lambda_w \stackrel{x}{=} \lambda'_w \stackrel{w}{=} \lambda'$ for $\lambda_w, \lambda'_w \in \Lambda^w$. One has $a \notin \lambda_w$ or $a \in \lambda'_w$, for otherwise $e \in \mathcal{G}_{\Lambda^w}$. Either way, the jump $l \to w$ is in $\mathcal{G}_\Lambda$. ☐

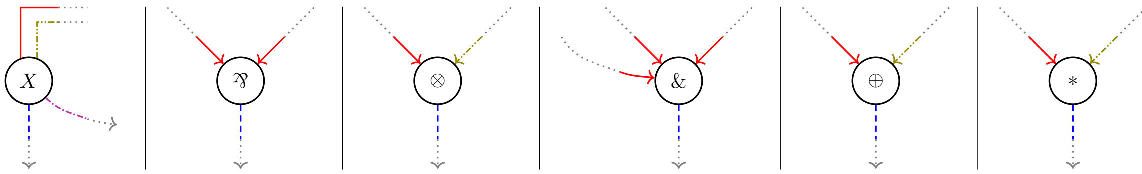


Figure 4.6: Local coloring of the graph of a set of linkings

Corollary 4.29 ([HG05, Lemma 4.32])**.** *Given a proof-net θ, every non-empty union Ω of switching cycles of $\mathcal{G}_\theta$ has a jump out of it: for some leaf $l \in \Omega$ and &-vertex $w \notin \Omega$, there is a jump $l \to w$ in $\mathcal{G}_\theta$.*

Proof. This is a corollary of Lemma 4.28. □

4.3.2 Finding a terminal splitting node

We now apply Theorem 3.19 to obtain the key ingredient in a proof of sequentialization: a terminal splitting node. We fix a local coloring c of a proof-structure as follows. All edges are colored differently except for switch arcs of ⅋\&-vertices which are given the same color when directed towards the ⅋\&-vertex – see Figure 4.6. In particular, there is no bridge of the shape (a^+, b^+) for a and b two arcs. Remark a bridge-free cycle for this coloring corresponds exactly to a switching cycle. We now consider sets of linkings endowed with such a local coloring, with as parameter of Theorem 3.19 the arcs whose target is a node taken forward (with direction $+$).

First, remark a maximal element for the ordering has a terminal target.

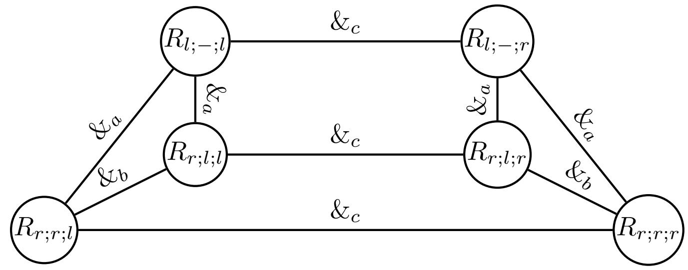
Lemma 4.30. *Let v be a non-terminal node in a proof-structure. Then v belongs to a switching cycle, or for all a of target v, there exists b such that $a^+ \lhd b^+$.*

Proof. Take v a non-terminal node and a an arc of target v. As v is not terminal, there exist an arc b and a node u such that b is a conclusion of v and a premise of u. Then $a^+ \overset{b^+}{\Rsh} b^+$ since (a^+, b^+) is not a bridge and b^+ is open. The only vertices of b^+ (seen as a path) are v and u. If $b^+ \overset{p}{\Rsh} v$ for some path p, then $b^+ \cdot p$ is a switching cycle containing v (by Lemma 3.4 and because b is a conclusion arc). Otherwise we have $a^+ \overset{b^+}{\lhd} b^+$. □

Lemma 4.31 (Splitting terminal)**.** *A proof-net is node-free or contains a splitting terminal node.*

Proof. Assume the proof-net has a node. Seen as a colored graph G, a proof-net respects (H^0_G), (H^1_G) and (H^2_G). Indeed, given a non-empty maximal connected union of bridge-free (*i.e.* switching) cycles, Corollary 4.29 gives an arc $\mathfrak{e}(\Omega)$, which is a jump arc from a leaf l in Ω to a &-vertex w outside of it. Its reverse is colored differently by c than all other edges of target l by definition of our coloring: all edges of target l have different colors as a leaf is not a ⅋\&-vertex. No two $\mathfrak{e}(\Omega)$ and $\mathfrak{e}(\Omega')$ can share a same support when $\Omega \neq \Omega'$, for they both are between a leaf and a &-vertex, so if $\mathfrak{e}(\Omega) = \mathfrak{e}(\Omega')$ then its source is a leaf l belonging to both Ω and Ω', thus $\Omega = \Omega'$ as they are maximal connected unions of bridge-free cycles.

Pose E the set of edges of the considered proof-net made of premises of ⅋-, ⊗-, & and ⊕-vertices, as well as jump arcs (all with direction $+$). In other words, E contains no edge of support an axiom



$R_{l;-;r}$ means we chose the left premise (l) of $\&_a$, $\&_b$ is absent ($-$) and the right premise (r) of $\&_c$

Figure 4.7: &-graph $\mathcal{W}_\Gamma$ of $\Gamma = X^+ \&_a (X^+ \&_b X^+), X^- \&_c X^-$

link, but has all other arcs taken forward.[5] This set E contains all bridge arches, as they are switch arcs of $⅋\backslash\&$-vertices taken forward, and all $\mathfrak{e}(\Omega)$, for those are jump arcs taken forward. It does not contain some $\overline{\mathfrak{e}(\Omega)}$ as this is an arc taken backward towards a leaf. Lastly, it is finite and non-empty as the proof-net has a node, hence it has a maximal element e for $\lhd$ (Lemma 3.16). Using Theorem 3.19, e has a splitting target, which must be terminal by Lemma 4.30 – because a splitting vertex is a pier in all cycles containing it, so cannot belong to a bridge-free (*i.e.* switching) cycle. $\square$

4.3.3 &-graph $\mathcal{W}_\Gamma$ of a sequent Γ

Contrary to MLL, in MALL, sequentialization is not an immediate consequence of Lemma 4.31. We need some results on proof-structures, notably that a terminal splitting $\oplus$-vertex is unary, so that it corresponds to a $\oplus_i$-rule, or that a splitting $*$-vertex is present in all additive resolutions, so that it corresponds to a final *cut*-rule. To obtain those, we use a tool called the *&-graph* of a cut sequent. This is a graphical representation of &-resolutions, allowing easier visualization of notions like toggling and dependency.

Definition 4.32 ($\mathcal{W}_\Gamma$)**.** The undirected total simple loop-free graph $\mathcal{W}_\Gamma$, called the *&-graph* of Γ, is defined as follows:

- $\mathcal{W}_\Gamma$ has for vertices the &-resolutions of Γ.
- for R and S two &-resolutions of Γ, there is an edge R — S in $\mathcal{W}_\Gamma$ if there is exactly one &-vertex w with different choices of premise in R and S; we label this edge with w, yielding $R \overset{w}{—} S$.

For instance, the graph $\mathcal{W}_\Gamma$ with $\Gamma = X^+ \&_a (X^+ \&_b X^+), X^- \&_c X^-$ is depicted on Figure 4.7. This graph represents the increase in complexity between MALL and MLL, because an $\mathsf{MLL}^{0,2}_{uf}$ proof-net is a single point on it.

Fact 4.33. *Given λ_0, λ_1 two linkings of a proof-structure and R_0, R_1 &-resolutions such that λ_0 (resp. λ_1) is on R_0 (resp. R_1), the &-vertices toggled by $\{\lambda_0; \lambda_1\}$ are included in the &-vertices with different choice of premises in R_0 and R_1.*

[5] In case one put pending arcs for the roots of the syntactic forest, we do not put such arcs in E.

Proof. Because a toggled & is in both additive resolutions $\Gamma \upharpoonright \lambda_0$ and $\Gamma \upharpoonright \lambda_1$, and $\Gamma \upharpoonright \lambda_i \subseteq R_i$ for R_i is a &-resolution on which λ_i is. ☐

Lemma 4.34. *Take linkings λ and λ' in a proof-structure, respectively on &-resolutions R and R', such that $R \overset{w}{—} R'$ is an edge in $\mathcal{W}_\Gamma$. Either w is the only & toggled by $\{\lambda; \lambda'\}$ or $\lambda = \lambda'$ (*i.e. $\lambda \overset{w}{=} \lambda'$*, notation from Section 4.3.1).*

Proof. A & toggled by $\{\lambda; \lambda'\}$ can only be w (Fact 4.33). If w is not toggled by $\{\lambda; \lambda'\}$, then λ' is also on R. This yields $\lambda = \lambda'$ through (P1). ☐

Useful paths to consider in $\mathcal{W}_\Gamma$ are those following toggled &-vertices.

Definition 4.35. A *toggling path* in $\mathcal{W}_\Gamma$ is a path between &-resolutions R and S such that the labels of its edges are exactly the &-vertices present in both R and S with different choices of premises, each taken exactly once.

Lemma 4.36. *The graph $\mathcal{W}_\Gamma$ is connected with respect to toggling paths: given two &-resolutions, there exists a toggling path between them in $\mathcal{W}_\Gamma$.*

Proof. Take two &-resolutions R and S, and denote by n the number of &-vertices with different choice of premises in R and S. We reason by induction on n.

If $n = 0$, then $R = S$ and the empty path is a toggling path. So assume $n \geq 1$, and take w one of those &-vertices. Consider R_w the &-resolution obtained from R by taking the other premise for w and giving to &-vertices introduced this way the same premise they have in S (these vertices are in S for S has not the same choice of premises for w as R). Remark that $R \overset{w}{—} R_w$ is an edge of $\mathcal{W}_\Gamma$.

By construction of R_w, there are $n-1$ &-vertices with different choice of premises in R_w and S: they are those with different choice of premises in R and S safe for w. By induction hypothesis, there is a toggling path μ between R_w and S. By Definition 4.35, μ has for labels on its edges the &-vertices with different choice of premises in R_w and S, each exactly once. It follows that $(R \overset{w}{—} R_w) \cdot \mu$ is a toggling path between R and S. ☐

Definition 4.37. For a proof-structure θ and a &-resolution R, we set $\boldsymbol{\lambda}(R)$ the unique linking of θ on R according to (P1).

The main use we will do of toggling paths is through the following lemma.

Lemma 4.38. *Consider two linkings μ and ν in a proof-structure θ. Take P a predicate on linkings of θ, and assume $P(\mu)$ is true while $P(\nu)$ is not. Then, there exist linkings $\mu', \nu' \in \theta$ such that $P(\mu')$ is true, $P(\nu')$ is not and $\{\mu'; \nu'\}$ toggles only one &-vertex, which is one of those toggled by $\{\mu; \nu\}$.*

Proof. Take R and S two &-resolutions respectively of μ and ν, chosen such that the &-vertices toggled by $\{\mu; \nu\}$ are exactly the &-vertices in R and S with different choice of premises. This can be done as follows, where we explain how to build R and a symmetric construction would yield S. For a given &-vertex, if it is in the additive resolution of μ than takes the same premise in R; otherwise, if it is in the additive resolution of ν, then keep the corresponding premise; otherwise, keep the left premise. Remark $\mu = \boldsymbol{\lambda}(R)$ and $\nu = \boldsymbol{\lambda}(S)$.

Take μ a toggling path between R and S (Lemma 4.36). As $P(\boldsymbol{\lambda}(R))$ is true while $P(\boldsymbol{\lambda}(S))$ is not, there is an edge $R' \overset{w}{—} S'$ in μ such that $P(\boldsymbol{\lambda}(R'))$ is true but $P(\boldsymbol{\lambda}(S'))$ is not – otherwise we could deduce $P(\boldsymbol{\lambda}(S))$ is true from $P(\boldsymbol{\lambda}(R))$, by following the edges of the path μ. As $\boldsymbol{\lambda}(R') \neq \boldsymbol{\lambda}(S')$, w is the only & toggled by $\{\boldsymbol{\lambda}(R'); \boldsymbol{\lambda}(S')\}$ (Lemma 4.34). By definition of a toggling path, and by our initial construction of R and S, w is one of the &-vertices toggled by $\{\mu; \nu\}$. $\square$

4.3.4 Results on proof-structures using &-graphs

Using &-graphs, one can more easily find jump arcs in a proof-net, which will entail the conditions we need to sequentialize thanks to splitting terminal vertices. The following characterization is the one we wish for the found vertices. A *terminal* node v in a proof-net θ is **sequentializing** if, depending on its kind:

- $\parr\backslash\&$-vertex: a terminal $\parr\backslash\&$-vertex is always sequentializing;
- $\otimes$-vertex: v belongs to no cycle in $\mathcal{G}_\theta$ – *i.e.* the removal of v in $\mathcal{G}_\theta$ breaks a connected component into two;
- $\oplus$-vertex: v is unary in θ;
- $*$-vertex: v belongs to the additive resolution of all linkings and is in no cycle in $\mathcal{G}_\theta$.

A terminal splitting $\parr\backslash\otimes\backslash\&$ corresponds to a sequentializing one by definition. We prove here that a terminal splitting $\oplus\backslash*$ is sequentializing – the reciprocal being easy.

Lemma 4.39. *Take a non-negative vertex v in a proof-structure θ such that there exist a left-ancestor l and a right-ancestor r of v in its formula tree, and a &-vertex w such that there are jump arcs $l \to w$ and $r \to w$ in $\mathcal{G}_\theta$. Then v is not splitting.*

Proof. First, remark w cannot be an ancestor of v, for it belongs to both the additive resolution of a linking containing l and to one containing r. We build a simple path p_l starting with the edge from v to its left premise and ending on w. Define q_l the simple open path from v to the leaf l using only conclusion arcs taken backward (with $-$ signs). The jump $l \to w$ also being a simple and open path, and w not being inside q_l, $p_l = q_l \cdot (l \to w)$ is a simple open path (Lemma 3.5) going from v to w – see Figure 4.8 for an illustration.

By applying the same reasoning on v and r, we have a simple open path p_r from v to w, sharing exactly its endpoints with p_l by construction – one is in the left sub-tree of v, the other in its right sub-tree. Wherefore, $p_l \cdot \overline{p_r}$ is a cycle starting from v, illustrated on Figure 4.8. As it does not contain v as a pier (for by our definition of coloring all edges with v as a target have different colors), v is not splitting. $\square$

Lemma 4.40 (Jumps of a terminal $\oplus$-vertex)**.** *Let v be a binary terminal $\oplus$-vertex of a proof-structure θ. There exist a left-ancestor l and a right-ancestor r of v in its formula tree, as well as a &-vertex w, such that there are jump arcs $l \xrightarrow{j} w$ and $r \xrightarrow{j} w$ in $\mathcal{G}_\theta$.*

Proof. As v is binary, there exist linkings $\lambda_{a'}, \lambda_{b'} \in \theta$ using respectively a leaf in the left and right formula tree of v. For v is terminal, any linking uses a leaf in the formula tree of v, either left or right (according to its additive resolution). Set $P(\lambda)$ the predicate "λ uses a leaf in the left formula tree of v". By Lemma 4.38 on this predicate, one gets linkings $\lambda_a, \lambda_b \in \theta$ such that λ_a uses the left sub-tree of v while λ_b uses the right one, and they toggle a unique &-vertex w. In particular, there is a link $a \in \lambda_a$ with a leaf l in the left formula tree of v, and a link $b \in \lambda_b$ with a leaf r in the right one. As a and b belong to different additive resolutions, we have more precisely $a \in \lambda_a \backslash \lambda_b$ and $b \in \lambda_b \backslash \lambda_a$. Thus, there are jump arcs $l \to w$ and $r \to w$ in $\mathcal{G}_\theta$ (see Figure 4.8). $\square$

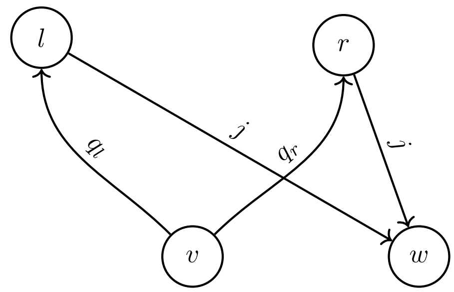


Figure 4.8: Illustration of a cycle containing a vertex having a left-ancestor and a right-ancestor with jump arcs to a same &-vertex

Corollary 4.41. *In a proof-structure θ, any binary terminal $\oplus$-vertex is not splitting.*

Proof. Corollary of Lemmas 4.39 and 4.40. $\square$

Lemma 4.42. *Let v be a $*$-vertex not in all additive resolutions of a proof-structure θ. There exist a left-ancestor l and a right-ancestor r of v in its formula tree, as well as a &-vertex w, such that there are jump arcs $l \xrightarrow{j} w$ and $r \xrightarrow{j} w$ in $\mathcal{G}_\theta$.*

Proof. By assumption, there exists a linking $\lambda'_{out} \in \theta$ such that v is not in its additive resolution. By (P0), v has a leaf in some λ'_{in}, thus v is in the additive resolution of this linking. Set $P(\lambda)$ the predicate "v is in the additive resolution of λ". By Lemma 4.38 on this predicate, one gets linkings $\lambda_{in}, \lambda_{out} \in \theta$ such that λ_{in} has v in its additive resolution while λ_b does not, and they toggle a unique &-vertex w. So there are axiom links $a, b \in \lambda_{in}$ using respectively a leaf l in the left and r in the right formula tree of v. Furthermore, $a, b \in \lambda_{in} \backslash \lambda_{out}$ as there are not in the additive resolution of λ_{out}. Thus, there are jump arcs $l \to w$ and $r \to w$ in $\mathcal{G}_\theta$ (see Figure 4.8). $\square$

Corollary 4.43. *In a proof-structure θ, a $*$-vertex not in all additive resolutions is not splitting.*

Proof. Corollary of Lemmas 4.39 and 4.42. $\square$

Lemma 4.44. *Let θ be a proof-structure, $\lambda, \lambda' \in \theta$ and $a \in \lambda \backslash \lambda'$ an axiom link. There exists a &-vertex w toggled by $\{\lambda; \lambda'\}$ such that a depends on w in θ.*

Proof. . Define a predicate P on linkings by $P(\mu) = a \in \mu$. By Lemma 4.38 on this predicate, one gets linkings $\mu, \mu' \in \theta$ such that $a \in \mu \backslash \mu'$ and $\{\mu; \mu'\}$ toggles a unique &-vertex w, that is also toggled by $\{\lambda; \lambda'\}$. Thus, a depends on w in θ. $\square$

4.3.5 Sequentialization

Thanks to the previous section, we now prove that terminal splitting vertices are easy to sequentialize. After proving it for each kind of vertex, we deduce the sequentialization theorem.

Lemma 4.45 (Sequentializing &)**.** *Let v be a terminal &-vertex in a proof-net θ on a cut sequent $[\Sigma]\ A_0 \,\&\, A_1, \Gamma$, with $A_0 \,\&\, A_1$ the formula associated to v. There exists θ_0, θ_1, Δ, Σ_0, Σ_1 such that $\theta = \theta_0 \sqcup \theta_1$, $\Sigma = \Delta, \Sigma_0, \Sigma_1$ and θ_i is a proof-net on $[\Delta, \Sigma_i]\ A_i, \Gamma$ for $i \in \{0; 1\}$.*

Proof. For $i \in \{0;1\}$, define θ_i as the linkings of θ using the premise A_i of $A_0 \,\&\, A_1$. Also set Δ the subset of Σ composed of the cut pairs having a leaf in both θ_0 and θ_1, and Σ_i the subset of Σ composed of the cut pairs having a leaf only in θ_i and not in θ_{1-i}.

We have θ the disjoint union of θ_0 and θ_1, for the additive resolution of any linking contains either (the root of) A_0 or A_1. That $\Sigma = \Delta, \Sigma_0, \Sigma_1$ follows: either a cut pair has a leaf in θ_0, or in θ_1 or in both, and it cannot be in none of them by (P0).

That θ_i is a set of linkings on $[\Delta, \Sigma_i]\ A_i, \Gamma$ follows from the fact that it is composed of linkings on $[\Sigma]\ A_0 \,\&\, A_1, \Gamma$ with A_i in its additive resolution. To obtain that θ_i is a proof-net on $[\Delta, \Sigma_i]\ A_i, \Gamma$, we have (P0), (P1), (P2) and (P3) to prove. Remark that, for all $\Lambda \subseteq \theta_i$, $\mathcal{G}_\Lambda$ in θ_i is $\mathcal{G}_\Lambda$ in θ where the unary vertex v is removed $(*)$.

The cut condition (P0) results immediately from the definition of Δ and Σ_i. The resolution condition (P1) follows directly from the one on θ, using that $\theta_i \subseteq \theta$. The MLL condition (P2) holds by $(*)$.

For the toggling condition (P3), take $\Lambda \subseteq \theta_i$ of size at least two. Then Λ is also a set of linkings of θ. By (P3) of θ applied to Λ, Λ toggles a &-vertex w not in any switching cycle of $\mathcal{G}_\Lambda$ of θ. This vertex w must be a vertex of $[\Delta, \Sigma_i]\ A_i, \Gamma$, for the other &-vertices are not toggled by $\Lambda \subseteq \theta_i$. Thus, w is in $\mathcal{G}_\Lambda$ of θ_i and it is in no switching cycle of it by $(*)$. ☐

Lemma 4.46 (Sequentializing $\parr$)**.** *Let v be a terminal $\parr$-vertex in a proof-net θ on a cut sequent $[\Sigma]\ A_0 \parr A_1, \Gamma$, with $A_0 \parr A_1$ the formula associated to v. Then θ is a proof-net on $[\Sigma]\ A_0, A_1, \Gamma$.*

Proof. The cut sequents $[\Sigma]\ A_0 \parr A_1, \Gamma$ and $[\Sigma]\ A_0, A_1, \Gamma$ have the same leaves and (aside from the presence/absence of v) the same $\&-$ and additive resolutions, so θ is also a set of linkings on $[\Sigma]\ A_0, A_1, \Gamma$ and respects (P0) and (P1) on it, for it respects them on $[\Sigma]\ A_0 \parr A_1, \Gamma$. For any $\Lambda \subseteq \theta$, $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_0, A_1, \Gamma$ is $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_0 \parr A_1, \Gamma$ where the vertex v is removed. Therefore, (P2) for θ on $[\Sigma]\ A_0, A_1, \Gamma$ follows from (P2) on $[\Sigma]\ A_0 \parr A_1, \Gamma$. Finally, any sub-set $\Lambda \subseteq \theta$ toggles the same &-vertices on $[\Sigma]\ A_0, A_1, \Gamma$ as it does on $[\Sigma]\ A_0 \parr A_1, \Gamma$, and $\mathcal{G}_\Lambda$ has the same switching cycles with respect to $[\Sigma]\ A_0, A_1, \Gamma$ as with respect to $[\Sigma]\ A_0 \parr A_1, \Gamma$, proving (P3). Therefore, θ is also a proof-net on $[\Sigma]\ A_0, A_1, \Gamma$. ☐

Lemma 4.47 (Sequentializing $\oplus$)**.** *Let v be a terminal splitting $\oplus$-vertex in a proof-net θ on a cut sequent $[\Sigma]\ A_0 \oplus A_1, \Gamma$, with $A_0 \oplus A_1$ the formula associated to v. There exists $i \in \{0;1\}$ such that θ is a proof-net on $[\Sigma]\ A_i, \Gamma$.*

Proof. By Corollary 4.41, v is unary: set $i \in \{0;1\}$ such that the formula tree of A_{1-i} has no axiom link on any of its leaves in $\mathcal{G}_\theta$.

For a linking $\lambda \in \theta$, we have by hypothesis that its additive resolution $[\Sigma]\ A_0 \oplus A_1, \Gamma \upharpoonright \lambda$ uses the premise of v corresponding to A_i (v belongs to it as it is terminal), thus $[\Sigma]\ A_i, \Gamma \upharpoonright \lambda$ is $[\Sigma]\ A_0 \oplus A_1, \Gamma \upharpoonright \lambda$ where v is removed. Therefore, λ is still a partitioning of the leaves of an additive resolution $[\Sigma]\ A_i, \Gamma \upharpoonright \lambda$ into links. Hence, θ is a set of linkings on $[\Sigma]\ A_i, \Gamma$. Remark that, for all $\Lambda \subseteq \theta$, $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_i, \Gamma$ is $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_0 \oplus A_1, \Gamma$ where we remove the unary v $(*)$. Let us prove that θ is a proof-net on $[\Sigma]\ A_i, \Gamma$.

Cut (P0) A cut pair in Σ has a leaf in θ on $[\Sigma]\ A_0 \oplus A_1, \Gamma$ by (P0), which is a leaf in θ on $[\Sigma]\ A_i, \Gamma$.

Resolution (P1) Let R' be a &-resolution of $[\Sigma]\ A_i, \Gamma$. We extend it into a &-resolution R of $[\Sigma]\ A_0 \oplus A_1, \Gamma$ by giving arbitrary choice of premises to the &-vertices of A_{1-i}. As all linkings of θ use the premise A_i of v, linkings on R' of $[\Sigma]\ A_i, \Gamma$ are exactly those on R of $[\Sigma]\ A_0 \oplus A_1, \Gamma$. Hence (P1) for θ on $[\Sigma]\ A_i, \Gamma$ from (P1) for θ on $[\Sigma]\ A_0 \oplus A_1, \Gamma$.

MLL (P2) By $(*)$, (P2) for θ on $[\Sigma]\ A_i, \Gamma$ follows from (P2) on $[\Sigma]\ A_0 \oplus A_1, \Gamma$.

Toggling (P3) Take $\Lambda \subseteq \theta$ of size at least 2. By (P3) applied to Λ on $[\Sigma]\ A_0 \oplus A_1, \Gamma$, Λ toggles a &-vertex w that is in no switching cycle of $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_0 \oplus A_1, \Gamma$. But w is in $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_i, \Gamma$ (for it is in the additive resolution of a linking, which takes A_i) and in no switching cycle of $\mathcal{G}_\Lambda$ on $[\Sigma]\ A_i, \Gamma$ by $(*)$. □

Lemma 4.48 (Sequentializing $\otimes\backslash *$). *Let v be a terminal splitting $\otimes$-vertex (resp. $*$-vertex) in a proof-net θ on a cut sequent $[\Sigma]\ A_0 \otimes A_1, \Gamma$ (resp. $[A_0 * A_1, \Sigma]\ \Gamma$), with $A_0 \otimes A_1$ (resp. $A_0 * A_1$) the formula associated to v. There exists θ_0, θ_1, Γ_0, Γ_1, Σ_0, Σ_1 such that $\theta = \{\lambda \cup \mu \mid \lambda \in \theta_0, \mu \in \theta_1\}$, $\Gamma = \Gamma_0, \Gamma_1$, $\Sigma = \Sigma_0, \Sigma_1$ and θ_i is a proof-net on $[\Sigma_i]\ A_i, \Gamma_i$ for $i \in \{0; 1\}$.*

Proof. Set $[\Sigma_0]\ \Gamma_0$ the elements of $[\Sigma]\ \Gamma$ whose roots in the removal of v in $\mathcal{G}_\theta$ are in the connected component of the root of A_0, and $[\Sigma_1]\ \Gamma_1$ the other elements (as v is splitting). We denote by Φ the cut sequent $[\Sigma]\ A_0 \otimes A_1, \Gamma$ (resp. $[A_0 * A_1, \Sigma]\ \Gamma$) and by Φ_i the cut sequent $[\Sigma_i]\ A_i, \Gamma_i$.

Pose, for $i \in \{0; 1\}$, $f_i(\lambda) = \{a \in \lambda \mid a \in \Phi_i\}$ for $\lambda \in \theta$, and $\theta_i = \{f_i(\lambda) \mid \lambda \in \theta\}$. We have $\Gamma = \Gamma_0, \Gamma_1$ and $\Sigma = \Sigma_0, \Sigma_1$ as we partitioned the formulas. Let us prove $\theta = \{\lambda \cup \mu \mid \lambda \in \theta_0, \mu \in \theta_1\}$, assuming (P1) for θ_i that we will prove later. By definition of the θ_i, $\theta \subseteq \{\lambda_0 \cup \lambda_1 \mid \lambda_0 \in \theta_0, \lambda_1 \in \theta_1\}$. Take $\lambda_0 \in \theta_0$ and $\lambda_1 \in \theta_1$, as well as R_0 and R_1 &-resolutions there are respectively on. Set R the &-resolution of θ taking the same choices of premises as R_0 and R_1. By (P1) of θ, there is a unique $\lambda \in \theta$ on it. But then for $i \in \{0; 1\}$, $f_i(\lambda)$ is on R_i. By (P1) of θ_i, this linking is λ_i, hence $\lambda = \lambda_0 \cup \lambda_1$. This proves $\theta = \{\lambda_0 \cup \lambda_1 \mid \lambda_0 \in \theta_0, \lambda_1 \in \theta_1\}$.

We now prove that θ_i is a proof-net on Φ_i. For $\lambda_i \in \theta_i$, we set $\lambda \in \theta$ a linking it origins from (*i.e.* such that $f_i(\lambda) = \lambda_i$).

θ_i is a set of linkings. Take $\lambda_i \in \theta_i$. We have $\lambda \in \theta$ a partition of the leaves of $\Phi \upharpoonright \lambda$. The terminal vertex v is inside this additive resolution (immediate if v is a $\otimes$-vertex and by Corollary 4.43 if it is a $*$-vertex). Thus, $\Phi \upharpoonright \lambda = \Phi \upharpoonright \lambda_0 \cup \Phi \upharpoonright \lambda_1$, and so λ_i is a partition of the leaves of $\Phi_i \upharpoonright \lambda_i$ for there is no axiom link between $\Phi \upharpoonright \lambda_0$ and $\Phi \upharpoonright \lambda_1$ by definition of $[\Sigma_0]\ \Gamma_0$.

Cut (P0) A cut pair in θ_i is one of θ, so has a link on it in θ. This link belongs to θ_i by definition of θ_i.

Resolution (P1) Let R_i be a &-resolution of Φ_i. We can extend it into an arbitrary &-resolution R of Φ. There exists a unique $\lambda \in \theta$ on R by (P1). Then λ_i is on R_i for there is no axiom link between Φ_0 and Φ_1. For unicity, assume we have $\lambda_i^1, \lambda_i^2 \in \theta_i$ both on R_i, with $\lambda_i^1 \neq \lambda_i^2$. There exists *w.l.o.g.* $a \in \lambda_i^1 \backslash \lambda_i^2 \subseteq \lambda^1 \backslash \lambda^2$. By Lemma 4.44, there exists a &-vertex w on which a depends (so there are jump arcs from the leaves of a to w) and toggled by $\{\lambda^1; \lambda^2\}$. It follows that w is in θ_i as there are no jump arcs from θ_i to θ_{1-i}. But then $\{\lambda_i^1; \lambda_i^2\}$ toggles w, a contradiction as they are on the same &-resolution R_i.

MLL (P2) For any $\lambda \in \theta$, $\mathcal{G}_{\lambda_i}$ is a sub-graph of $\mathcal{G}_\lambda$. Hence the preservation of (P2).

Toggling (P3) Take $\Lambda_i = \{\lambda_i^j \mid j \in [\![1; n]\!]\}$ a set of linkings of θ_i of size $n \geq 2$. Each λ_i^j, for $j \in [\![1; n]\!]$, has a &-resolution R_i^j it is on in θ_i. We choose an arbitrary &-resolution on θ_{1-i} and use it to extend all R_i^j to &-resolutions R^j on θ. By (P1) of θ, there exists a unique linking μ^j associated to each R^j.

We have for all $j \in [\![1; n]\!]$, $f_i(\mu^j) = \lambda_i^j$. Indeed, the &-resolution R^j of μ^j coincides with R_i^j on θ_i, and we conclude by (P1) of θ_i. We pose $\Lambda = \{\mu^j \mid j \in [\![1; n]\!]\}$, of size $n \geq 2$ (they are distinct for their projections by f_i are distinct).

By (P3), there exists a &-vertex w toggled by Λ and not in any switching cycle of $\mathcal{G}_\Lambda$. This vertex w cannot be in θ_{1-i}, for all linkings of Λ have the same &-resolution on θ_{1-i} by definition: thus it belongs to θ_i. As there are no jump arcs from θ_{1-i} to θ_i, the axiom links depending on w are in λ_i, and thus belong to $f_i(\mu^j) = \lambda_i^j$. Therefore Λ_i toggles w. Finally, a switching cycle of $\mathcal{G}_{\Lambda_i}$ is a switching cycle of $\mathcal{G}_\Lambda$, as $\mathcal{G}_{\Lambda_i}$ is an induced sub-graph of $\mathcal{G}_\Lambda$ (thanks to $f_i(\mu^j) = \lambda_i^j$ for $j \in [\![1;n]\!]$). It follows that w is in no switching cycle of $\mathcal{G}_{\Lambda_i}$, for it is in no switching cycle of $\mathcal{G}_\Lambda$. $\square$

We have now everything needed to prove the sequentialization theorem.

Theorem 4.49 (Sequentialization)**.** *Given a proof-net θ on a cut sequent $[\Sigma]\ \Gamma$, there exists a proof π of $\vdash \Gamma$ in $\mathsf{MALL}^{0,2}_{uf}$ such that $\theta \in \mathfrak{P}(\pi)$.*

Proof. We reason by induction on the number of nodes of $\mathcal{G}_\theta$. Assume first that $\mathcal{G}_\theta$ contains a non-leaf vertex. By Lemma 4.31, there is a terminal splitting node v in $\mathcal{G}_\theta$. We distinguish cases according to the kind of this node.

If v is a $⅋$-vertex. In this case, $[\Sigma]\ \Gamma = [\Sigma]\ A_0 ⅋ A_1, \Delta$ with $A_0 ⅋ A_1$ the formula associated to v. Using Lemma 4.46, θ is a proof-net on $[\Sigma]\ A_0, A_1, \Delta$. By induction hypothesis, one gets a proof π_0 of $\vdash A_0, A_1, \Delta$ such that $\theta \in \mathfrak{P}(\pi_0)$. We add a $⅋$-rule to π_0, obtaining a proof π of $\vdash A_0 ⅋ A_1, \Delta$ satisfying $\theta \in \mathfrak{P}(\pi)$.

If v is a $\&$-vertex. In this case, $[\Sigma]\ \Gamma = [\Sigma]\ A_0 \& A_1, \Delta$ with $A_0 \& A_1$ the formula associated to v. Using Lemma 4.45, $\Sigma = \Phi, \Sigma_0, \Sigma_1$ and $\theta = \theta_0 \cup \theta_1$ with θ_i a proof-net on $[\Phi, \Sigma_i]\ A_i, \Delta$ for $i \in \{0; 1\}$. By induction hypothesis, one gets proofs π_i of $\vdash A_i, \Delta$ such that $\theta_i \in \mathfrak{P}(\pi_i)$. We add a $\&$-rule to π_0 and π_1, obtaining a proof π of $\vdash A_0 \& A_1, \Delta$ satisfying $\theta \in \mathfrak{P}(\pi)$.

If v is a $\oplus$-vertex. In this case, $[\Sigma]\ \Gamma = [\Sigma]\ A_0 \oplus A_1, \Delta$ with $A_0 \oplus A_1$ the formula associated to v. Using Lemma 4.47, there exits $i \in \{0; 1\}$ such that θ is a proof-net on $[\Sigma]\ A_i, \Delta$. By induction hypothesis, one gets a proof π_0 of $\vdash A_i, \Delta$ such that $\theta \in \mathfrak{P}(\pi_0)$. We add a $\oplus_i$-rule to π_0, obtaining a proof π of $\vdash A_0 \oplus A_1, \Delta$ satisfying $\theta \in \mathfrak{P}(\pi)$.

If v is a $\otimes$-vertex. In this case, $[\Sigma]\ \Gamma = [\Sigma]\ A_0 \otimes A_1, \Delta$ with $A_0 \otimes A_1$ the formula associated to v. Using Lemma 4.48, $\Delta = \Delta_0, \Delta_1$, $\Sigma = \Sigma_0, \Sigma_1$ and $\theta = \{\lambda \cup \mu \mid \lambda \in \theta_0, \mu \in \theta_1\}$ with θ_i a proof-net on $[\Sigma_i]\ A_i, \Delta_i$ for $i \in \{0; 1\}$. By induction hypothesis, one gets proofs π_i of $\vdash A_i, \Delta_i$ such that $\theta_i \in \mathfrak{P}(\pi_i)$. We add a $\otimes$-rule to π_0 and π_1, obtaining a proof π of $\vdash A_0 \otimes A_1, \Delta$ satisfying $\theta \in \mathfrak{P}(\pi)$.

If v is a $*$-vertex. In this case, $[\Sigma]\ \Gamma = [A_0 * A_1, \Delta]\ \Gamma$ with $A_0 * A_1$ the formula associated to v. Using Lemma 4.48, $\Gamma = \Gamma_0, \Gamma_1$, $\Delta = \Delta_0, \Delta_1$ and $\theta = \{\lambda \cup \mu \mid \lambda \in \theta_0, \mu \in \theta_1\}$ with θ_i a proof-net on $[\Delta_i]\ A_i, \Gamma_i$ for $i \in \{0; 1\}$. By induction hypothesis, one gets proofs π_i of $\vdash \Gamma_i$ such that $\theta_i \in \mathfrak{P}(\pi_i)$. We add a *cut*-rule to π_0 and π_1, obtaining a proof π of $\vdash \Gamma$ satisfying $\theta \in \mathfrak{P}(\pi)$.

Now assume that $\mathcal{G}_\theta$ contains no node. This implies $\Sigma = \emptyset$ and Γ is composed only of atoms. In particular, $\theta = \{\lambda\}$ using (P1). We reason now by induction on the size of λ. If λ is not the empty set, take a link a inside, between leaves X^+ and X^-. Define λ' and Γ' such that $\lambda = \lambda' \cup a$ and $\Gamma = X^-, X^+, \Gamma'$ (up to permutation). One can easily check that $\{\lambda'\}$ is a proof-net on $[\ ]\ \Gamma'$. By induction hypothesis, one gets a proof π' of $\vdash \Gamma'$ such that $\{\lambda'\} \in \mathfrak{P}(\pi')$. We build:

$$\pi = \dfrac{\dfrac{}{\vdash X^-, X^+}\ {\scriptstyle (ax)} \qquad \begin{array}{c}\pi' \\ \vdash \Gamma'\end{array}}{\vdash X^-, X^+, \Gamma'}\ {\scriptstyle (mix_2)}$$

which satisfies $\{\lambda\} \in \mathfrak{P}(\pi)$.

If $\lambda = \emptyset$, then it must be on the empty cut sequent $[\ ]$. We have $\{\emptyset\}$ the sole element of $\mathfrak{P}\left(\dfrac{}{\vdash}\ {\scriptstyle (mix_0)} \right)$. $\square$

The proof π built here always contains a mix_0-rule and contains a mix_2-rule as soon as θ is not empty. It is possible to optimize this by considering separately the case of a proof-net reduced to a single link between a pair of leaves, which is the desequentialization of $\overline{\vdash X^+, X^-}\ (ax)$. In this way, we always get a $\overset{om}{\rightsquigarrow}$-normal form.

Remark 4.50. With this proof of sequentialization, it is easy to check that there is some freedom in the definitions of proof-nets. For instance, one could replace axiom links with *ax*-vertices having two out-going arcs instead, and put jump arcs from these *ax*-vertices instead of from leaves of the links, keeping the same correctness criterion. Or one can put jump arcs not from both leaves of a link, but only from the negative one. The proof of sequentialization in [HG05] may also allow such modifications, for the only use of (P3) inside is to prove Corollary 4.29 too; nonetheless it is less clear that the rest of the proof is mostly independent from the definition of a jump arc.

While at first sight this last part (Sections 4.3.3 to 4.3.5) of the proof of sequentialization is longer than the corresponding one in [HG05], with in particular the use of the &-graphs, this is because we prove in details arguments that were simply claimed in their proof. As an example, the following result is used in [HG05] without an explicit proof.

Lemma 4.51 (Claim in the proof of sequentialization of [HG05]). *Let v be a splitting $⅋\backslash\&$-vertex in a proof-net θ on $[\Sigma]\ \Gamma$. For all $\lambda \in \theta$, $v \in [\Sigma]\ \Gamma \upharpoonright \lambda$.*

Proof. Notice at least one linking contains v in its additive resolution, otherwise v is not in $\mathcal{G}_\theta$. We prove the contrapositive: given a vertex $v \in \mathcal{G}_\theta$ such that there exist linkings $\lambda, \lambda' \in \theta$ with v in $[\Sigma]\ \Gamma \upharpoonright \lambda$ but not in $[\Sigma]\ \Gamma \upharpoonright \lambda'$, there exists a cycle of $\mathcal{G}_\theta$ containing the conclusion of v. Thus, there is an additive vertex u such that one of its premises is kept by λ while λ' keeps the other, with v in the syntactic tree of u.

Without any loss of generality, these linkings toggle only one &-vertex, w. Indeed, set P a predicate on linkings by $P(\mu) = v \in [\Sigma]\ \Gamma \upharpoonright \mu$. By Lemma 4.38 on this predicate, one gets linkings $\mu, \mu' \in \theta$ such that v in $[\Sigma]\ \Gamma \upharpoonright \mu$ but not in $[\Sigma]\ \Gamma \upharpoonright \mu'$ and $\{\mu; \mu'\}$ toggles a unique &-vertex w.

Set l a leaf in $\lambda\backslash\lambda'$ found by going up in the sub-tree of v, following premises kept in the additive resolution of λ. Similarly, there is a leaf $l' \in \lambda'\backslash\lambda$ by going up in the sub-tree of u, following premises kept in the additive resolution of λ'. The cycle $v \dashrightarrow u \dashleftarrow l' \rightarrow w \leftarrow l \dashrightarrow v$, where $\dashrightarrow$ is a path using as edges the arcs of a syntactic tree in the positive direction, is a cycle containing the conclusion arc of v (w is not an ancestor of u for it belongs to both additive resolutions). $\square$

4.3.6 Other proofs of sequentialization

The work on graph theory factorizes several proofs of sequentialization. Using the parameter E of Theorem 3.29, one can recover the proof by splitting $⅋\backslash\&$ from [HG05], as well as by other kinds of splitting vertices. Here are possible such sets (sorted by increasing size):

- if E is the set of all switch arcs of $⅋\backslash\&$-vertices taken with direction $+$, one gets a splitting $⅋\backslash\&$-vertex;
- if E is the set of all edges of the shape a^+ with a not an axiom link, one gets a terminal splitting node (Lemma 4.31);
- if E contains all edges except jump arcs taken backward, or all edges except those of support an axiom link and jump arcs taken backward, one gets an arbitrary splitting vertex (possibly a leaf).

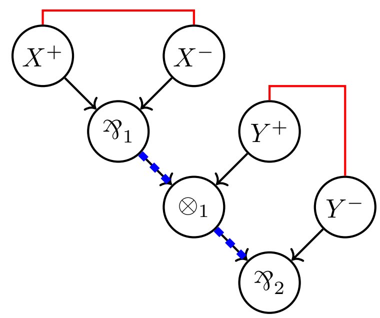


Figure 4.9: Connected proof-net where $\parr_1$ is a splitting $\parr$ whose two premises are not maximal for $\lhd$

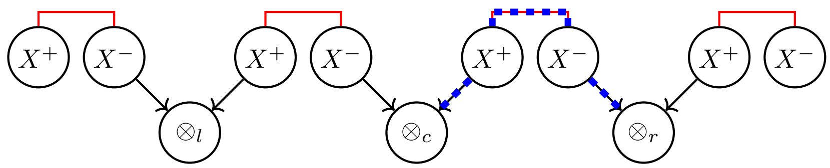


Figure 4.10: Connected proof-net with a terminal splitting vertex $\otimes_c$ whose both premises are not maximal for $\lhd$

For the cases other than the one we chose, sequentialization is possible by considering some kind of proof-nets with hypotheses (shared among all slices), meaning we cut some sub-trees of the syntactic forest – this is needed because the splitting vertex found may not be terminal. The concept of such proof-nets is at best complex, and does not seem that useful outside of sequentialization, hence we will not give any more details. Another solution, which is the one chosen in [HG05], consists in introducing a new leaf and axiom link to recover a proof-net without hypothesis.

4.3.7 Splitting but not maximal

Using Theorem 3.19 in the context in proof-nets in Sections 4.3.2 and 4.3.6, we saw that edges maximal for $\lhd$ lead to splitting vertices. However, the converse does not necessarily hold as some splitting vertices may not be the target of a maximal edge, even in the simpler setting of MLL_{uf} proof-nets.

If one is looking at splitting $\parr$-vertices, with E made of premises of $\parr$-vertices, a counter-example is given on Figure 4.9. This connected proof-net has two splitting vertices, $\parr_1$ and $\parr_2$. However, both premises of $\parr_1$ are smaller than the left premise of $\parr_2$, using the path $\parr_1 \to \otimes_1 \to \parr_2$ (dashed on Figure 4.9).

Similarly, when searching for a terminal splitting vertex, with E made of premises of all $\parr$- and $\otimes$-vertices, Figure 4.10 illustrates a counter-example. All three $\otimes$-vertices of this connected proof-net are terminal and splitting, but no premise of $\otimes_c$ is maximal: the left one is smaller than the left premise of $\otimes_r$ while the right one is lesser than the right premise of $\otimes_l$, using respectively the paths $\otimes_c \leftarrow X^+ — X^- \to \otimes_r$ (dashed on Figure 4.10) and $\otimes_c \leftarrow X^- — X^+ \to \otimes_l$.

Finally, with E made of all edges except jump arcs taken backward (and possibly with no edges

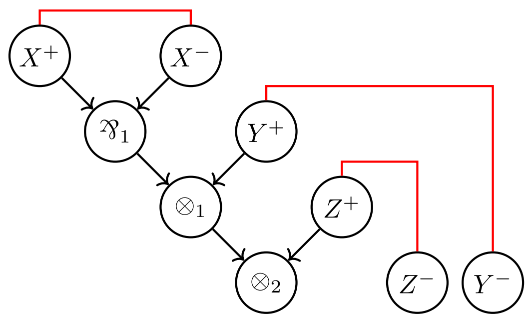


Figure 4.11: Connected proof-net with a splitting vertex $\otimes_1$ whose in-edges are not maximal for $\lhd$

of support a link), Figure 4.11 gives a counter-example. The splitting vertex $\otimes_1$ in this connected proof-net has none of its in-edges maximal for $\lhd$: both of its premises are smaller than the left premise of $\otimes_2$, with the path $\otimes_1 \to \otimes_2$, and its conclusion arc taken backwards (with $-$) is smaller than the conclusion arc of $⅋_1$ taken backwards, using $\otimes_1 \leftarrow ⅋_1$.

Looking only at connected proof-nets in MLL_{uf}, there is a notion of *kingdom* [BV95], where the kingdom $k(v)$ of v is the smallest sub-graph of the proof-net where v is a terminal vertex and which is a connected proof-net. This implies a natural order on vertices: u is smaller than v if $u \in k(v)$. It is possible to prove the following: in a MLL_{uf} connected proof-net, if a is an arc of $k(v)$ then, for any arc b with source v, $a^+ \lhd b^+$. In other words, our order $\lhd$ contains the order of kingdoms. The converse is false, as shown on Figure 4.10 where the kingdom of each $\otimes$-vertex is its set of premises. We will not give any more details about this as kingdoms work badly in $\mathsf{MLL}_{uf}^{0,2}$ and in MALL_{uf}, not to speak about $\mathsf{MALL}_{uf}^{0,2}$.

4.3.8 Variations

Now that we have sequentialization for full $\mathsf{MALL}_{uf}^{0,2}$, we can consider some restrictions to specific sub-systems, and characterize sub-systems of the sequent calculus by means of properties of their image in proof-structures. The key argument is that Lemma 4.31 holds for proof-nets, so in particular for connected proof-nets respecting (P2$^{\mathrm{c}}$), but also for proof-nets respecting (P2$^{\mathrm{c}0}$) or (P2$^{\mathrm{c}2}$).

Given a proof π, consider one of its slice s, and set $\#mix_2$ (resp. $\#mix_0$) the number of mix_2-rules (resp. mix_0-rules) inside. Looking at the associated linking λ_s (recall Remark 4.16), each of its $⅋$-switching ϕ is such that $\mathcal{G}_\phi$ has $1 + \#mix_2 - \#mix_0$ connected components. In particular, a proof without mix_2- and mix_0-rules (resp. without mix_0-rules, without mix_2-rules) desequentializes into a proof-net respecting (P2$^{\mathrm{c}}$) (resp. (P2$^{\mathrm{c}2}$), (P2$^{\mathrm{c}0}$)).

Conversely, one can adapt the proof of Theorem 4.49 to proof-nets respecting (P2$^{\mathrm{c}}$) (resp. (P2$^{\mathrm{c}2}$), (P2$^{\mathrm{c}0}$)) to recover a proof without mix_2- and mix_0-rules (resp. without mix_0-rules, without mix_2-rules) – because each of these properties is preserved by removal of a sequentializing vertex. This yields variations of Theorem 4.18 in these three cases.

4.4 General Results on Proof-Nets

We give here some general results about proof-nets, that will be of use later.

4.4.1 Connectivity and number of connected components

Remember that, given a linking λ in a proof-net θ and one of its $\parr$-switchings ϕ, we defined a graph $\mathcal{G}_\phi$, which is asked to be acyclic. Also remember we may consider its number of connected components, in particular when working without the mix_2-rule.

Lemma 4.52. *Pose θ a set of linkings respecting* (P2)*. Take a linking $\lambda \in \theta$ and a $\parr$-switching ϕ. The number of connected components of $\mathcal{G}_\phi$ is equal to $|V| + |\parr| - |E|$ where $|V|$ is the number of vertices of $\mathcal{G}_\lambda$, $|\parr|$ its number of $\parr$-vertices and $|E|$ its number of edges.*

Proof. As $\mathcal{G}_\phi$ is acyclic by (P2) (recall Remark 4.2), using Corollary 3.48 its number of connected components is equal to its number of vertices minus is number of edges. As $\mathcal{G}_\phi$ is obtained from $\mathcal{G}_\lambda$ by deleting one edge by $\parr$-vertex, the conclusion follows. □

Thanks to this lemma, one can define **the number $\#cc$ of connected components of a linking** λ as $|V| + |\parr| - |E|$ (with the conventions of the above lemma). Then, a proof-net is connected if and only if for any linking λ, its number of connected components is equal to 1.

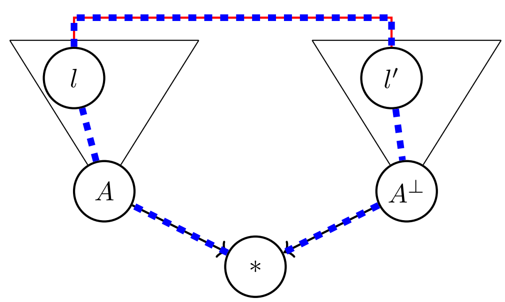
In particular, for a $\mathsf{MLL}_{uf}^{0,2}$ proof-net, made of only one linking by (P1), one can define its number of connected components, which is not to be confound with the standard number of connected components from graph theory of the graph associated to this proof-net.

4.4.2 Forbidden Sub-graphs in Proof-Nets

We give in this section some forbidden sub-graphs that a (connected) proof-net cannot have.

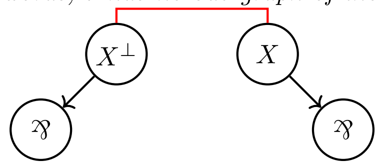
Lemma 4.53. *Consider a proof-net θ on the cut sequent $[A * A^\perp, \Sigma]\ \Gamma$. No axiom link of θ is between a leaf of A and one of $A^\perp$.*

Proof. Towards a contradiction, assume there is an axiom link between a leaf l of A and a leaf l' of $A^\perp$, in a linking λ. Then there is a switching cycle in $\mathcal{G}_\lambda$, a contradiction with (P2). This cycle, depicted below, uses the incriminated link l — l', the path of the syntactic tree from l' to the root of $A^\perp$, goes through the $*$-vertex, and then takes the path of the syntactic tree from the root A to l.



□

Lemma 4.54. *A connected proof-net θ does not contain a link $⅋X^\perp$ — $X⅋$, where these two $⅋$-vertices are distinct. In other words, θ has no sub-graph of the following shape:*

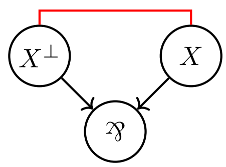


Similarly, θ cannot contain a link $⅋X^\perp$ — X (resp. $X^\perp$ — $X⅋$) where X (resp. $X^\perp$) is a formula (occurrence) of the cut sequent of θ.

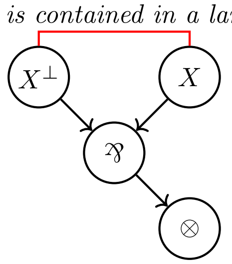
Proof. By contradiction, consider such a link a. It belongs to a linking $\lambda \in \theta$. Consider a $⅋$-switching ϕ of λ such that the $⅋$-vertex below the $X^\perp$ (resp. X) in a does not keep its premise leading to this $X^\perp$ (resp. X). By (P2) and (P2^c), $\mathcal{G}_\phi$ is a tree (see Remark 4.4). Contradiction, as the sub-graph made of $X^\perp$ — X is a connected component of $\mathcal{G}_\phi$, not connected to the $⅋$-vertices we saw.

The other cases, with less $⅋$-vertices, are similar. ☐

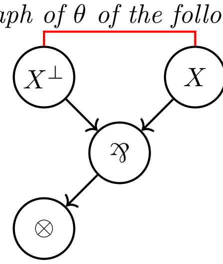
Lemma 4.55. *Consider a cut-free connected $\mathsf{MLL}^{0,2}_{uf}$ proof-net θ containing a proper sub-graph of the following shape:*



Then this sub-graph is contained in a larger sub-graph of θ of the following form:



or



Proof. As it is a proper sub-graph, and by (P2^c), there must be a connective below the $⅋$-vertex under consideration – otherwise this sub-graph would be a connected component of the proof-net. This connective cannot be a $⅋$-vertex, for this would yield an unconnected $\mathcal{G}_\phi$ with ϕ a $⅋$-switching where we choose not to keep the premise of this $⅋$ leading to the sub-graph. So, this connective must be a $\otimes$. ☐

Remark 4.56. Lemma 4.55 does not hold in $\mathsf{MALL}^{0,2}_{uf}$, as below could be a $\oplus$-vertex for instance.

4.5 Perspectives

After recalling the definition of proof-nets for $\mathsf{MALL}^{0,2}_{uf}$ from [HG05], we give a proof of the sequentialization theorem thanks to our generalization of Yeo's theorem. This last theorem could be instrumental in proving this theorem for other notions of proof-nets. It should be at least useful for non-atomic-axiom proof-nets for $\mathsf{MALL}^{0,2}_{uf}$. To define those, it should suffice to have an axiom link not between leaves, but generally between edges of dual labels. Then everything should work the same, provided a linking with a &-vertex above one of its links is considered as having both premises of this & in its additive resolution. Note however that, from a complexity point of view, our proof herited from graph does not seem optimal. Indeed, while our method is efficient for proving the existence of a splitting vertex, searching for such a vertex thanks to our ordering may need many steps. A linear algorithm for sequentializing MLL proof-nets is known [Gue11], and it could be interesting to consider this problem for MALL.

While the interactions between additive and exponential connectives of linear logic are notoriously complicated, we presume this in mainly due to the comonoidal structure imposed on exponentials. In a setting where exponentials have none of these properties – $?_w$ is not neutral for $?_c$, and $?_c$ is neither associative nor commutative – a proof-net syntax looks possible. Indeed, in such a framework one can see the $?_c$-rule (read from bottom to top) as taking an occurrence of $?A$ and creating another occurrence of it. It is then possible to follow occurrences of ?-formulas, with each $?_c$-vertex having a main parent and a secondary one; it is in particular possible to do so across slices, just by "superimposing" exponential trees, along the main (original) occurrence! Doing the naive proof-nets in this setting by merging $\mathsf{MALL}^{0,2}_{uf}$ and $\mathsf{MELL}^{0,2}$ proof-nets seems to do the trick, up to adding jump arcs from $?_c$-vertices to &-vertices such that toggling them makes the $?_c$-vertex disappear – *i.e.* when there is a $?_c$-rule on this formula in the, say, left sub-tree of the &-rule, that is not in its right sub-tree. Proving sequentialization for these new proof-nets should be as simple as using Theorem 3.19 in the $\mathsf{MALL}^{0,2}_{uf}$ case.

Nonetheless, even if such proof-nets could be defined for unit-free linear logic, with these restraints on properties of exponentials, an extension to units seems hardly doable. The main result from [HH16a] rules out "simple" *canonical* proof-nets with multiplicative units: determining if two MLL proofs (with units) are equal up to rule commutation is PSPACE-complete. If one forgets the canonical requirement for the units, some nice definitions for MLL exist, *e.g.* [Hug13]. Still, one has to consider in such proof-nets the analog of rule commutation for unit, called *rewiring* [Blu+96; Hug12]. Considering additive units, the situation is worse as a $\top - \otimes$ commutation can create or erase a sub-proof. Here is a problematic example, courtesy of Willem Heijltjes. As soon as $\vdash A$ and $\vdash B$ are provable, then all the following proofs are equivalent up to rule commutations, whatever the sub-proofs π and ϕ are.

$$\dfrac{\overset{\pi}{\vdash A} \qquad \dfrac{}{\vdash \top, \top \otimes B}\,(\top)}{\vdash A \otimes \top, \top \otimes B}\,(\otimes) \qquad\qquad \dfrac{\overset{\pi}{\vdash A} \qquad \dfrac{\dfrac{}{\vdash \top, \top}\,(\top) \qquad \overset{\phi}{\vdash B}}{\vdash \top, \top \otimes B}\,(\otimes)}{\vdash A \otimes \top, \top \otimes B}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A} \qquad \dfrac{}{\vdash \top, \top}\,(\top)}{\vdash A \otimes \top, \top}\,(\otimes) \qquad \overset{\phi}{\vdash B}}{\vdash A \otimes \top, \top \otimes B}\,(\otimes) \qquad\qquad \dfrac{\dfrac{}{\vdash A \otimes \top, \top}\,(\top) \qquad \overset{\phi}{\vdash B}}{\vdash A \otimes \top, \top \otimes B}\,(\otimes)$$

However, if only one of the two, say $\vdash A$, is provable then we generally cannot equalize all proofs of the form $\dfrac{\overset{\pi}{\vdash A} \qquad \dfrac{}{\vdash \top, \top \otimes B}\,(\top)}{\vdash A \otimes \top, \top \otimes B}\,(\otimes)$. It seems really hard to have a unique canonical object corresponding to all of these proofs, because here what matters is exactly provability. As a complement, other solutions than proof-nets can be considered to study canonical proofs in the sequent calculus linear logic, see *e.g.* [CMS08] for an approach based on focusing.

Chapter 5

Formalization of Multiplicative Proof-Nets in Coq

Proof assistants are used to formalize mathematical reasonings and thus to certify formal proofs. One of these proof assistants is *Coq* [Coq], in which the *Yalla* [Lau17] library is developed. This library, developed by my PhD's advisor Olivier Laurent, focuses on linear logic and the formalization of results in this theory. While many key results have already been proved in this library, for instance admissibility of the *cut*-rule, there is nothing about proof-nets. More generally, while there have been several implementations of (sub-systems of) linear logic in various proof assistants (*e.g.* in Coq [Lau17; PW99; Xav+18; CL; PC; Sad03], Abella [CLR19; CLR17] and Isabelle [KP95; Gro95]), this has always been done for the sequent calculus syntax, and not for the proof-net syntax. As proof-nets are one of the most important discovery in linear logic, and their theory is well developed on paper, this absence is quite problematic. Such a state can be explained by the main constraint when formalizing proof-nets: while a proof of sequent calculus is an inductive object, easily defined in most proof assistants, a proof-net is a (multi)graph, which is harder formally to define and manipulate.

There are relatively few results about graphs in proof assistants, as compared to results on paper. An explanation is that many reasonings involves "geometric" or "graphical" arguments, with drawings to explain the proof, as well as silent "evident" arguments – for instance when writing "$p \cdot q$ is a simple path", one does not often precise why this holds nor gives explicitly all hypotheses making the concatenation simple (see Lemma 3.4). As another simple example, consider a simple path p in a graph G, and remove a vertex v not in this path. Then, on paper, one affirms that p is a simple path in $G - v$. On computer, there are a lot more to do: firstly vertices of $G - v$ are not syntactically vertices of G, so that p is not a path of $G - v$ and one has to define a function f sending paths from G to paths of $G - v$. Then, one has to explicitly prove that $f(p)$ is simple because p is. More generally, another explanation for the difficulty of considering graphs in proof assistants is that transformations of graphs – such as adding or removing vertices and arcs – are done very frequently in the literature but are complex formally: vertices after the transformation are not those before, and whereas on paper one usually and implicitly consider graphs up to isomorphisms, on computer one has to provide these isomorphisms (for instance when removing a vertex and adding it back, we do not get exactly the original graph). Nonetheless, graph theory can benefit from the use of proof assistants, as shown by the four colors theorem proved in Coq by Georges Gonthier [Gon23].

In proofs like this one with many cases, a computer check that every case has been handled brings more trust compared to pages upon pages of proofs with a very large case study (*c.f.* the full proof in appendix of Proposition 2.46).

Since a few years, a graph library has been implemented in Coq by *Damien Pous and Christian Doczkal*, called *Graph Theory* [DP18], built upon the *Mathematical Components* library [MC] with its *SSReflect* proof language. Thanks to this library, some non-easy results from graph theory have been proved, involving for instance minor or treewidth [DP20]. Hence, this library seems robust enough and adequate to the formalization of proof-nets. In particular, this library contains the total (directed multi)graphs defined in Section 3.1.2, with labeling functions on both vertices and arcs, as well as many operations we need: adding and removing vertices and arcs, making the union of two graphs, isomorphisms of graphs, properties of the standard operations up to isomorphism (*e.g.* that disjoint union is associative and commutative), sub-graphs, ...

One of the goals of this PhD was to formalize connected proof-nets for MLL_{uf} in Coq. Even if there is not much doubt that the key results (sequentialization, cut-elimination, ...) about these proof-nets hold, for they have been proved in many different ways, this is not the case for all proof-nets; for instance, the proof of sequentialization of $\mathsf{MALL}_{uf}^{0,2}$ proof-nets of [HG05] was at the best of our knowledge the only one before the other developed in this thesis. As this proof, and generally proofs of sequentialization for proof-nets more complex than MLL_{uf} ones, are usually at least a little involved, a formalization can bring some more confidence on them. This is also the case for demonstrations that proof-nets identify proofs exactly up to rule commutation (with cut), that usually involve plenty of case studies. Still, before formalizing these complex objects, starting with the simpler and most successful MLL_{uf} proof-nets is a first step where most difficulties arising from the formalization are present: does the graph libraries used have all needed results? can they be used easily? It is also a sufficient setting to try several formalizations of various equivalent definitions so as to see which ones are simpler to manipulate on computer, as even these basic objects have an abundance of definitions – and the definition of proof-nets we formalized does not correspond to the one defined in Chapter 4, because we want proof-nets with possibly non expanded axioms. Proof-nets for MLL_{uf} are also good objects to see how much the proofs from the theory of proof-nets on paper are expanded when encoded in a proof assistant, so as to estimate the amount of work needed for more complex proof-nets. To sum up, starting with MLL_{uf} proof-nets is a key step before formalizing more involved proof-nets, in a field that would really benefit from a formalization. It is also an instrumental step towards more complex results, such as proving that two proofs desequentialize to the same proof-net if and only if there are equal up to rule commutation.

In more details, here are the key results one wants for MLL_{uf} proof-nets in a proof assistant:

- a (formal) definition of what is a proof-structure, a proof-net, and a connected proof-net.
- a definition of the desequentialization function, and a proof of the sequentialization theorem;
- a definition of cut-elimination, and a proof it is convergent;
- all links with sequent calculus: cut-elimination in proof-nets mimics the one of sequent calculus, similarly for axiom-expansion, proof-nets are the quotient of proofs by rule commutation (with cut), ...

The first three preceding results are the core of the theory, and already there we have non-trivial choices to make for which of the equivalent definitions to take of proof-structure, of the correctness criterion, which proof of sequentialization to formalize, ... Furthermore, this led us to develop parts of the graph library which were lacking according to our needs, such as the theory of undirected simple paths. As it is complicated to guess which detailed definition is the most adapted to a formalization, we had to try and modify several choices all along the implementation.

What has been implemented exactly are the followings:

- a definition of connected proof-nets (and thus of the previous strata);
- a definition of the desequentialization function from proofs in sequent calculus to graphs;
- a proof that a graph obtained by desequentialization is a connected proof-net;
- a proof of the sequentialization theorem: given a connected proof-net, there exists a proof desequentializing to it, or more precisely to a graph isomorphic to this proof-net;
- a definition of cut-elimination in a connected proof-net;
- a proof that applying a cut-elimination step on a connected proof-net step yields a connected proof-net, and that one can rewrite to a cut-free graph;
- a proof of (an equivalent formulation of) the parameterized local Yeo Theorem (Theorem 3.29), used in the proof of sequentialization – in fact we even prove a version with a little more freedom on the parameter so as not to deal with the case of an arc-free graph;
- a proof that the number of connected components of an acyclic multigraph is given by its number of vertices and arcs, used for showing preservation of connectivity (a variation of Corollary 3.48).

The source code is public and can be found here: `https://github.com/RemiDiG/proofnet_mll`.

In this chapter are given details about some of these results, how they have been formalized and why they have been formalized this way – as a large part of the work was to try formalization choices to discover which ones were best suited to be formalized on a computer.

The author would like to thank *Damien Pous* for advice about how to use his graph library GraphTheory and the Mathematical Components library that GraphTheory itself uses, as well as for discussions about the implementation of proof-nets.

Outline We begin by giving the chosen formal definition of proof-net on paper, as well as parts of its implementation in Coq with some discussion about rejected solutions (Section 5.1). Then are explained the main difficulties encountered during the formalization, that should be solved before formalizing more involved notions of proof-net (Section 5.2). We end this chapter with a quick overview of the code structure (Section 5.3).

5.1 Formalization of proof-nets

We present here the cornerstone of our formalization, where choices mattered the most: the formal definition of connected proof-nets, and all depending strata (proof-structures, ...). Choosing these definitions was the most complex part of the formalization because they directly influence all other choices of formalization, *e.g.* the definition of the desequentialization function. Many different, but equivalent, definitions of multiplicative proof-nets exist in the literature: the one we gave in

Chapter 4, proof-nets defined as hypergraphs [GM01], ... We choose here one of the most usual presentations, with a proof-net seen as a directed multigraph made of *ax*-, $⅋$-, $\otimes$- and *cut*-vertices, and arcs between them. In addition to being the most standard definition of MLL_{uf} proof-nets, it seems simple enough (contrary to our definition in Chapter 4 that is more adapted to the presence of additives). Even once this definition chosen, there are still subtle variations possible, among which:

- does one put conclusion vertices, or have pending arcs, and thus a partial graph?
- does one put *ax*-vertices or just axiom links?
- which correctness criterion to differentiate proof-nets among proof-structures?
- $⅋$- and $\otimes$-vertices have ordered premises (*i.e.* syntactically $A ⅋ B \neq B ⅋ A$); how should it be taken into account?

Our goal was to choose a formalization making the definitions and proofs we are interested in easier to write (desequentialization, sequentialization and cut-elimination mostly). As it is hard to guess beforehand which choice on proof-nets leads to a simpler implementation on the whole, we successively tried several definitions. Furthermore, for taking advantage of the definition of proofs (from sequent calculus) already implemented in the Yalla library, some notions of formulas and sequents were already fixed in view of integrating this work into Yalla. In particular, the choices made in the current version of Yalla is that a sequent is a *list* of formulas and there is an exchange rule, which is quite not the setting used in this thesis (recall the discussion in Section 1.1.3).

When defining proof-nets, there are concretely two main difficulties. On one hand, we need to add some supplementary data on graphs: an order to recover a sequent, how to identify a left and a right premise of a $⅋$- or $\otimes$-vertex. On the other hand, there are all the constraints on proof-structures: in- and out-degrees of vertices, relations respected by formulas of arcs. It is natural to separate these two parts, with first a definition of graphs with some more data, and then asking properties on these new objects, so as to get proof-structures and then connected proof-nets. The part on properties coming from the definitions of the supplementary data, it is no surprise the first part was the most difficult to formalize.

In this section, we start by giving on paper a precise definition of connected proof-nets, corresponding to the one we formalized. Then we describe some non-trivial implementation choices we made in the end, explaining why alternatives we thought of are worse, and thus justifying the previous choices for our precise definitions.

5.1.1 Definition of proof-nets on paper

We give here another equivalent definition of connected proof-nets for MLL_{uf} than the one obtained from the proof-nets for $\mathsf{MALL}_{uf}^{0,2}$ of Section 4.1. The present definition is more in the format of graph theory, so that a proof-net is a graph respecting some properties, and not a linking on a sequent such that a computable graph respects some properties. We choose to have conclusion vertices and *ax*-vertices, so as to have total graphs, from which the proved sequent can be recovered, instead of some notion of "graphs on a given sequent", as is done in Chapter 4.

We start with the kind of underlying graphs to consider.

Definition 5.1. A MLL_{uf} **graph** is a finite total directed multigraph equipped with:

- a labeling function $\mathcal{R}$ from vertices to rules of MLL_{uf} – ax, $\otimes$, $\parr$ or cut – or to a special symbol c for conclusion vertices;
- a labeling function $\mathcal{F}$ from arcs to formulas;
- a labeling function $\mathcal{L}$ from arcs to booleans to identify left premises.

Vertices are named according to their label: ax**-vertices**, cut**-vertices**, $\otimes$**-vertices**, $\parr$**-vertices** and c**-vertices** respectively. Given a vertex v, arcs with target v are the **premises** of v and arcs with source v are the **conclusions** of v. A c-vertex is a **conclusion** of the graph.

We need to identify which arcs are left and right ones for $\otimes$- and $\parr$-vertices. The standard trick on paper is to always write the left premise on the left of the vertex and the right on its right. Obviously, formally we need something else, which is here the $\mathcal{L}$ function. Remark we do not have all needed information here: when considering the sequent associated to such a graph, we want it to be a *list* for compatibility with the Yalla library. Thus, we have to order vertices labeled c, as for instance $\vdash A, B$ and $\vdash B, A$ are different sequents in this formalism. This leads to the following definition.

Definition 5.2. A MLL_{uf} **pre-proof-structure** is a MLL_{uf} graph along with a list of arcs of this graph, called its **order**. The **sequent** associated to a formal pre-proof-structure is the list of formulas obtained by applying $\mathcal{F}$ on this list.

From these graphs, which contain all the data we need, proof-structures can be defined, with on one hand some geometrical constraint on in- and out-degrees, and one the other hand considerations on labels of arcs. All these constraints are easily defined because there are local. We also have to relate the list of arcs order to the c-vertices.

Definition 5.3. A MLL_{uf} **proof-structure** is a MLL_{uf} pre-proof-structure such that:

- any vertex v such that $\mathcal{R}(v) = \parr$ has exactly two in-coming arcs a and b and exactly one out-going arc c; moreover, $\mathcal{L}(a)$ is true, $\mathcal{L}(b)$ is false and $\mathcal{F}(c) = \mathcal{F}(a) \parr \mathcal{F}(b)$;
- any vertex v such that $\mathcal{R}(v) = \otimes$ has exactly two in-coming arcs a and b and exactly one out-going arc c; moreover, $\mathcal{L}(a)$ is true, $\mathcal{L}(b)$ is false and $\mathcal{F}(c) = \mathcal{F}(a) \otimes \mathcal{F}(b)$;
- any vertex v such that $\mathcal{R}(v) = ax$ has no in-coming arc and exactly two out-going arcs a and b; moreover, $\mathcal{F}(b) = \mathcal{F}(a)^{\perp}$;
- any vertex v such that $\mathcal{R}(v) = cut$ has exactly two in-coming arcs a and b and no out-going arc; moreover, $\mathcal{F}(b) = \mathcal{F}(a)^{\perp}$;
- any vertex v such that $\mathcal{R}(v) = c$ has exactly one in-coming arc and no out-going arc;
- the order of the pre-proof-structure contains exactly the arcs of target a c-vertex, each exactly once;
- any arc a not of target a $\parr$- or $\otimes$-vertex respects $\mathcal{L}(a)$ is true.

The last condition in the previous definition is there for the sake of canonicity, and is a convention (choosing $\mathcal{L}(a)$ is false would also be adequate). Such a convention is needed because $\mathcal{L}$ is defined on all arcs. Having it defined only on in-arcs of $\parr$- or $\otimes$-vertices is enough, but formally in Coq this would be a function whose domain is a dependent type, which is complex to manipulate. Having a simpler definition, with some easy conditions to check, was overall simpler in the implementation.

We choose (a variation of) the Danos-Regnier correctness criterion [DR89], as it is one of the most used and for it speaks about concepts from the theory of graphs only, that will anyway be needed at some point. An important remark is that we define correctness directly on the level of MLL_{uf} graphs, *i.e.* on the simpler objects we manipulate. Thus are defined switching paths.

Definition 5.4. A **switching path** is an undirected path which is arc-simple, and not containing several premises of a $⅋$-vertex. A **switching cycle** is a closed undirected path which is arc-simple, and not containing several premises of a $⅋$-vertex. A **left-switching path** is an undirected path which is arc-simple, and not containing a premise of a $⅋$-vertex whose image by $\mathcal{L}$ is false.

We choose arc-simple paths and not vertex-simple ones, as they are easier to concatenate (see Fact 3.2 and Lemma 3.4).

Definition 5.5. A MLL_{uf} **proof-net** is a MLL_{uf} proof-structure which is **correct**, meaning we have both:

acyclicity any switching cycle is an empty path;

connectivity between any pair of vertices there is a left-switching path, and the graph is non-empty; in other words, the number of connected components for left-switching paths (corresponding to the connected components of the sub-graph where right premises of $⅋$-vertices are removed) is equal to one.

One can check this definition is equivalent to the standard one. The key arguments are the following:

1. there is an arc-simple switching cycle if and only if there is a vertex-simple one;

2. all acyclic correctness graphs have the same number of connected components.

That Item 1 holds is because in a proof-structure any arc-simple cycle is a vertex-simple one, due to the fact that vertices are of degree at most three. Indeed, take such a cycle and assume a given vertex has more than two incident arcs inside. Then it must have at least four of those, as it is a cycle: contradiction. On the other hand, Item 2 is a consequence of Corollary 3.48, as all correctness graphs have the same number of vertices and arcs – one arc by $⅋$-vertex is removed. Hence, they all have exactly one connected component if and only if any of them has so, and in particular we can consider the correctness graph where we remove all right premises.

The main reason to consider these switching and left-switching paths is to stay in the underlying graph of the proof-net. If one were to directly consider correctness graphs, then one would need to prove results of the following kind: if a correctness graph is acyclic, and a vertex in the proof-net is removed, then the correctness graph stays acyclic. These results, not proved on paper because they are considered trivial, are tedious formally because one needs to transport paths from one graph to another, which is not hard but is a long and boring chore. Considering only one graph prevents the need for such lemmas, at the lighter cost of considering paths with restrictions on their arcs – restriction we need in some shape anyway, to prevent the repetition of arcs. Moreover, our definitions of switching and left-switching paths, which may seem ad hoc, are simply the instantiations of an alternating path for a particular arc-coloring in both cases.

Remark 5.6. We define here only MLL_{uf} proof-nets and not $\mathsf{MLL}_{uf}^{0,2}$ ones, because when starting this implementation one of the simplest proofs of sequentialization known was applicable only in this setting, and for a first formalization of proof-nets adapting a simple proof was an obvious concern. Since then, we found a simpler proof of sequentialization even in the presence of the optional multiplicative rules – corresponding to an use of Theorem 3.29, that itself admits a simple proof. This proof of sequentialization is the one formalized, but the implementation was already too far gone at that point to easily adapt it to this extended setting.

Once proof-nets are defined, other definitions follow more or less straightly. Let us consider the case of desequentialization in details.

Definition 5.7 (MLL_{uf} desequentialization)**.** We define by induction on a proof π of $\vdash \Gamma$ its **desequentialization** $\mathfrak{P}_f(\pi)$ which is a MLL_{uf} pre-proof-structure whose sequent is $\vdash \Gamma$.

- If π is reduced to an *ax*-rule with conclusion $\vdash A^\perp, A$, then $\mathfrak{P}_f(\pi)$ is the pre-proof-structure with one *ax*-node v, two *c*-vertices u_1 and u_2 and two arcs a_1 and a_2. Both arcs have source v, and a_1 (resp. a_2) has target u_1 (resp. u_2) and label $A^\perp$ (resp. A). The order of the pre-proof-structure is $[a_1; a_2]$.

- If the last rule of π is a $⅋$-rule applied to a proof π_1 then $\mathfrak{P}_f(\pi)$ is obtained from $\mathfrak{P}_f(\pi_1)$ by removing the two first arcs in the order of $\mathfrak{P}_f(\pi_1)$ as well as their targets, and adding a new $⅋$-node v and a new *c*-vertex u. We also add two new arcs with the same sources and labels A and B as the removed arcs, but with as target v, and another arc labeled $A ⅋ B$ with source v and target u. The order of $\mathfrak{P}_f(\pi)$ is first u, then the remaining arcs coming from the order of $\mathfrak{P}_f(\pi_1)$ (keeping their ordering).

- If the last rule of π is a $\otimes$-rule applied to two proofs π_1 and π_2 then $\mathfrak{P}_f(\pi)$ is obtained from the disjoint union of $\mathfrak{P}_f(\pi_1)$ and $\mathfrak{P}_f(\pi_2)$ by removing the first arc in the order of $\mathfrak{P}_f(\pi_1)$, as well as its target, and the one of $\mathfrak{P}_f(\pi_2)$ and adding a new $\otimes$-node v and a new *c*-vertex u. We also add two new arcs with the same sources and labels A and B as the removed arcs, but with as target v, and another arc labeled $A \otimes B$ with source v and target u. The order of $\mathfrak{P}_f(\pi)$ is first the added arc of label $A \otimes B$, then the remaining arcs in the order of $\mathfrak{P}_f(\pi_2)$, then those of $\mathfrak{P}_f(\pi_1)$ – this corresponds to a rule $\dfrac{\vdash A, \Gamma \quad \vdash B, \Delta}{\vdash A \otimes B, \Delta, \Gamma}\ (\otimes)$, which is the one chosen in Yalla so as to consider cyclic linear logic.

- If the last rule of π is a *cut*-rule applied to two proofs π_1 and π_2 then $\mathfrak{P}_f(\pi)$ is obtained from the disjoint union of $\mathfrak{P}_f(\pi_1)$ and $\mathfrak{P}_f(\pi_2)$ by removing the first arc in the order of $\mathfrak{P}_f(\pi_1)$, as well as its target, and the one of $\mathfrak{P}_f(\pi_2)$ and adding a new *cut*-node v. We also add two new arcs with the same sources and labels A and $A^\perp$ as the removed arcs, but with as target v. The order of $\mathfrak{P}_f(\pi)$ is the remaining arcs in the order of $\mathfrak{P}_f(\pi_2)$ followed by those of $\mathfrak{P}_f(\pi_1)$.

- If the last rule of π is a *ex*-rule applied to a proof π_1 then $\mathfrak{P}_f(\pi)$ is the same graph as $\mathfrak{P}_f(\pi_1)$ with as order the one of $\mathfrak{P}_f(\pi_1)$ to which is applied the permutation of the *ex*-rule.

5.1.2 Formalization of graphs and pre-proof-structures

We start from the definition of graphs from the library GraphTheory, which is the following:

```
Record graph (Lv Le : Type) : Type :=
```

```
  Graph {
      vertex:> finType;
      edge: finType;
      endpoint: bool → edge → vertex;
      vlabel: vertex → Lv;
      elabel: edge → Le }.
Notation source := (endpoint false).
Notation target := (endpoint true).
```

Said differently, a graph is the data of a finite set *vertex* of vertices, *edge* of arcs, a function *endpoint* generalizing the source and target functions, and labeling functions *vlabel* and *elabel* on vertices and arcs respectively to arbitrary types *Lv* and *Le*. Remark that for more confusion an edge for the library is what we call an arc. For consistency with the rest of this thesis, we will still call it an arc, even if it is named an edge in the Coq code snippets.

Looking at this starting definition from the library, one better understands our choices in Definition 5.1, choosing definitions adapted to those of the graph library. The implementation of MLL_{uf} graphs from this definition is the following:[1]

```
Notation base_graph := (graph (flat rule) (flat (formula × bool))).
Definition flabel {G : base_graph} (e : edge G) : formula := fst (elabel e).
Definition llabel {G : base_graph} (e : edge G) : bool := snd (elabel e).
```

This means a *base_graph* is a graph with vertices labeled by *rule* (a type whose elements are $\{ax; cut; ⅋; \otimes; c\}$), and arcs by *formula* (taken from Yalla) together with a *bool* to identify left and right arcs. As we need two labeling functions on arcs, one giving formulas and the other stating if an arc is a left one, we consider graphs with the product of these labeling functions, *i.e.* one from arcs to the cartesian product, from which we can recover the two labeling functions *flabel* and *llabel* respectively corresponding to $\mathcal{F}$ and $\mathcal{L}$. This definition was easy to write, because it corresponds to an instance of the graphs defined in the library GraphTheory, which is one of the reasons we took on paper this definition of MLL_{uf} graph.

Going to pre-proof-structure, in Definition 5.2, is done as follows:

```
Record graph_data : Type :=
  Graph_data {
    graph_of :> base_graph;
    order : seq (edge graph_of);
  }.
```

Later, for a proof-structure, the list of arcs *order* will be asked to contain exactly once each arc of target a c-vertex. This list allows to define the sequent associated to such a graph:

```
Definition sequent (G : graph_data) : seq formula :=
  [seq flabel e | e ← order G].
```

We choose to separate the list *order* from the other data on the graph, because these are quite disjoint concepts, and we rarely need the sequent of a graph. In particular, many concepts can be defined independently of this list, for instance switching paths. Meanwhile, the sequent is only used to prove it is preserved by a transformation (as cut-elimination) or in desequentialization and sequentialization.

There have been several notable previous choices of implementation we tried for defining MLL_{uf}

[1] The *flat* function is the identity, and is used by the graph library for technical reasons about isomorphisms. It can be safely ignored.

graphs and pre-proof-structures, corresponding to other definitions than the ones given in Section 5.1.1.

Usually, on paper, left arcs are not identified this way, and it may seem strange to decide this soon that an arc is a left one. One could wonder if it would not be simpler to report such considerations to the time where proof-structures are defined. A short answer is that it is always easier to have one's data on a simple object, than a more accurate but partial data on complex structures. A main reason is that, when modifying a graph, one also has to adapt proofs that it is a proof-structure, on which will depend the definition of a left arc in this case. This leads to convoluted proofs and definitions of left arcs, whereas having left from the very beginning, with possibly some nonsensical value for, say, the premise of a c-vertex, makes defining objects easier. And the easier our objects are, the easier are the proofs on them.

Still, other solutions than labeling functions can be considered. Doing a formalization as near as what happens usually on paper led us first to define functions *left* and *right*, generalized as a unique function *direction*, defined on ⅋- and ⊗-vertices, and an *order* function, associating to each c-vertex its order in the sequent proved by the graph, *i.e.* an integer bound by the number of c-vertices. This yields the following definition in Coq, where '*I_n* is the set of natural numbers less than *n*.

```
Record graph_data : Type :=
  Graph_data {
      graph_of :> graph rule formula;
      order : [finType of { v : graph_of | vlabel v == c}] →
        'I_#|[finType of { v : graph_of | vlabel v == c}]|;
      order_inj : injective order;
      direction : bool → [finType of { v : graph_of | vlabel v == ⊗ || vlabel v == ⅋ }]
        → edge graph_of;
  }.
Notation left := (direction false).
Notation right := (direction true).
```

The *order* function takes as argument a c-vertex and returns an integer less than the number of c-vertices. It is supposed injective, which by cardinality argument implies bijectivity. Alas, this definition is terrible to manipulate, in particular because of *order*: using cardinals makes it mandatory to compute these cardinals as soon as one defines such a graph, or worse when defining a graph from another. As the generic formal way of computing cardinals is giving explicitly a bijection, this becomes immediately unbelievably tedious. Moreover, checking injectivity to define a new graph was quite heavy, as we need to do so for instance each time we remove or add a vertex. Even improving this last part, for instance by saying *order* is a permutation on the c-vertices, the definition remains complex to manipulate. It gives the following:

```
      order : { perm [finType of { v : graph_of | vlabel v == c}]}
```

Furthermore, the use of dependent types, whether for *order* or *direction*, complexifies the usage of this definition, which is a shame as it is our basic block and therefore is present everywhere! As an example, to do a basic operation such as the disjoint union of two graphs, one needs to use proofs that a vertex labeled ⊗ before keeps this label after, and so on, just to define this union.

In order not to use dependent types, and thus having easily manipulable definitions, we define the functions we need in the most general possible way, even if only some of the arguments are to be considered, and then ask these functions to respect some properties. This way, the basic definitions are simple to write and easy to handle, and then we prove properties on them, that we later can use when needed. If one further defines only *left* instead of both *left* and *right* with *direction* – as *right* can be deduced from *left* – we get an even simpler definition, starting to look like our final one.

```
Record graph_data : Type :=
  Graph_data {
    graph_of :> graph rule formula;
    left : graph_of → edge graph_of;
    order : seq graph_of;
  }.
```

In other words, we only have a function from vertices to arcs, and a list of vertices. Still, the function *left* has bogus values for *ax*-, *cut*- and *c*-vertices, as it really gives left premises of ⅋- and ⊗-vertices. The problem here was that giving a valid bogus value – for instance an arc for a *c*-vertex – was sometimes non-trivial when modifying a proof-net, which complexified artificially some proofs. The need to have some existing arc to give as a bogus value means we have to know more about our graph, as there exists graphs without arcs. This in turn prevents us from making general definitions valid on all graphs, as they would involve some proof our graph has an arc!

Adding an option type on the output, with *left* : *graph_of* → *option* (*edge graph_of*) on the graph giving for each ⅋- or ⊗-vertex its left arc, and for other vertices *None*, gives a bogus value for free and solves our previous difficulty. Nonetheless, manipulating an option type for such a basic concept becomes heavy quickly, and we would then need some property that *left* applied to a vertex v gives *None* exactly when it is not a ⅋- or ⊗-vertex.

Finally, taking for *order* a list of vertices, one can ask for a proof-structure that this list contains exactly once each *c*-vertex. This solution is simple enough, except for defining the sequent of a proof-structure: we need to use the proof that each *c*-vertex is of degree one to get its arc, and then take the label of this arc. This was a little more complex than the current solution, with a list of arcs directly. A variation could have been to define $c(A)$-vertices, with A a formula which should be the one of the arc towards this vertex, so as to define a sequent directly from the list of vertices. Still, it gives a more complex type for the label of vertices, in particular there is no more a finite number of those. Plus, enforcing that a $c(A)$-vertex has for in-arc one labeled A would have been one more property to enforce at the level of proof-structure.

This is why the implementation given at the beginning was chosen, as its simplicity helps keeping following definitions and proofs easy to handle. Still, the left premise of a ⅋- or ⊗-vertex is undefined at this stratum, and will only be at the level of proof-structures. This is not really a problem, for we really need this concept only for upper levels. This solution is nonetheless not without fault, for we need to fix a value for *llabel* on irrelevant arcs, that we choose arbitrary to be *true*. Also, conceptually, *c*-vertices are kind of useless, and are there only for their in-arcs and the fact that the graph is total. This is still better than the previous difficulties encountered with the other definitions.

5.1.3 Formalization of proof-structures

The instantiation we choose of Definition 5.3 is to consider a pre-proof-structure along proofs this graph respects the properties in the definition:

```
Record proof_structure : Type :=
  Proof_structure {
    graph_data_of :> graph_data;
```

```
  p_deg : proper_degree graph_data_of;
  p_ax_cut : proper_ax_cut graph_data_of;
  p_tens_parr : proper_tens_parr graph_data_of;
  p_noleft : proper_noleft graph_data_of;
  p_order_full : proper_order_full graph_data_of;
  p_order_uniq : proper_order_uniq graph_data_of;
}.
```

Each of these local properties correspond to a part of Definition 5.3:

p_deg each vertex has the in- and out-degrees corresponding to its label;

p_ax_cut for each *ax*-vertex (resp. *cut*-vertex), there exists two arcs a and b of source (resp. target) this vertex, whose labels are dual formulas;

p_tens_parr for each $⅋$-vertex (resp. $\otimes$-vertex), there exists three arcs a, b and c with the first two of source this vertex, the last one of target this vertex, a is a left arc but not b and the label of c is the $⅋$ (resp. the $\otimes$) of the labels of a and b;

p_noleft each arc of target an *ax*-, *cut*- or *c*-vertex is a left arc (which is an arbitrary choice, but one has to be done for canonicity);

p_order_full arcs in *order* are exactly arcs of target a *c*-vertex;

p_order_uniq the list *order* has no duplicate.

For instance, the code for *proper_tens_parr* is the following:

```
Definition proper_tens_parr (G : base_graph) :=
  ∀ (b : bool) (v : G), vlabel v = (if b then ⅋ else ⊗) →
  ∃ el er ec,
  el \in edges_at_in v ∧ llabel el ∧
  er \in edges_at_in v ∧ ¬llabel er ∧
  ec \in edges_at_out v ∧ flabel ec = (if b then parr else tens) (flabel el) (flabel er).
```

An important remark is that here the exact choices of the definition of these properties are quite irrelevant to the complexity of the code, with the meaning that when modifying them one mostly just has then to modify their proofs and uses. This is quite different from the previous explained choices, as modifying the definition of *base_graph* for instance would lead to modifying almost all following definitions and proofs.

This is from this point only that the functions *left*, *right*, ... can be defined, as before there was no guarantee for instance that a $\otimes$-vertex has exactly one in-arc with label left. We chose not to use *left* to define these properties instead, as this would need to define some pseudo-left first at the level of *base_graph*, and to prove this function has good properties, which complexifies the code.

We could have separated the properties about *order* from the other ones, having on one side *p_deg*, *p_ax_cut*, *p_tens_parr* and *p_nolef*, corresponding to properties on *base_graph*, and on the other side *p_order_full* and *p_order_uniq* which consider the list *order*. We choose not to do so as this would give one or two more stratum in our definitions, and we would still need to have both when studying sequentialization, which is the peak of our implementation. Hence we kept these properties on the same level.

5.1.4 Formalization of the correctness criterion

Once proof-structures have been implemented, one can finally define (connected) proof-nets, using a correctness criterion. Recall our particular definition of the Danos-Regnier correctness criterion in Definition 5.5. An important reason to choose this solution is that removing vertices in a graph is done by GraphTheory through a dependent type, and as explained previously it is preferable not to use them when possible.

Having the same general notion of paths, so that the two concepts of switching and left-switching paths are instantiations of it, allows us to factorize most of the work, hence it is the solution we used. On one hand, we need to be able to identify different arcs (premises of a same ⅋-vertex for switching paths) and to forbid some arcs (right premises of ⅋-vertices) for left-switching paths. Thus are defined f-paths, given as parameter a function f from arcs to some option type, which can be seen as an arc-coloring with a special color *None*. A path is an **f-path** if its image by f has no duplicate (meaning for two arcs a and b in this walk, which are not the same occurrence in the walk seen as a list, $f(a) \neq f(b)$) and does not contain *None*. Remark this is more constrained than what happens in an alternating path: we have here a global criterion that implies in particular arc-simplicity.

Switching paths are f-paths for f the identity, except for premises of ⅋-vertices which are sent to the same element – we choose here the ⅋-vertex they point to. Left-switching paths are f-paths for f the identity, except for right (*i.e.* non-left) premises of ⅋-vertices which are sent to *None*.

About the implementation in Coq, undirected paths were not in GraphTheory, there were only directed ones. We defined them this way, by mimicking the definition of directed paths:

```
Definition upath {Lv Le : Type} {G : graph Lv Le} := seq ((edge G) × bool).
Notation usource e := (endpoint (~~e.2) e.1).
Notation utarget e := (endpoint e.2 e.1).
Fixpoint uwalk (x y : G) (p : upath) :=
  if p is e :: p' then (usource e == x) && uwalk (utarget e) y p' else x == y.
```

Then we define f-paths. Notice with need decidable equality in the image of f, so as to be able to compare images of arcs.

```
Definition supath {Lv Le : Type} {I : eqType} {G : graph Lv Le} (f : edge G → option I) (s t
: G) (p : upath) :=
  (uwalk s t p) && uniq [seq f e.1 | e ← p] && (None \notin [seq f e.1 | e ← p]).
```

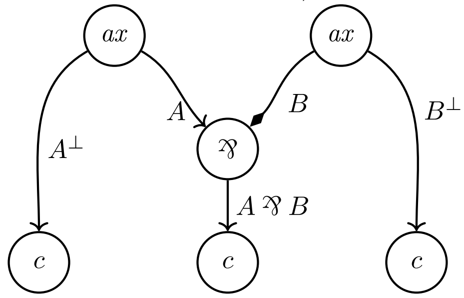
Remark in particular that with $f : e \longmapsto Some\ e$, an f-path is exactly an arc-simple path. We can then define our switching and left-switching paths:

```
Definition switching {G : base_graph} : edge G → option ((edge G) + G) :=
  fun e ⇒ Some (if vlabel (target e) == ⅋ then inr (target e) else inl e).
Definition switching_left {G : base_graph} : edge G → option (edge G) :=
  fun e ⇒ if (vlabel (target e) == ⅋) && (~~llabel e) then None else Some e.
```

There is no problem for defining acyclicity for this kind of paths. We only need to take care of the empty path, that is a cycle in our formalization – which was simpler as we do not want to check a list is empty too often (when concatenating, ...).

```
Definition uacyclic {Lv Le : Type} {I : eqType} {G : graph Lv Le} (f : edge G → option I) :=
  ∀ (x : G) (p : Supath f x x), p = supath_nil f x.
```

Nonetheless, using such a general notion of f-paths is not trivial when considering connectivity. While operations such as taking a sub-path or inverting a path work as well as for arc-simple paths, the notion of connected component is awkward because the relation "being linked by an f-path" is not transitive. In particular, being linked by a switching path is not, looking at this graph (where a diamond arrowhead is used to indicate a non-left arc):



Both ax-vertices are linked to the ⅋-vertex by means of switching paths, but there is no switching path from one to the other. Still, this relation is transitive if f is injective for arcs not of image *None*, as it is the case for left-switching. In such a setting, we prove a generalization of Corollary 3.48.

5.2 Limits of the implementation

We explain here the main difficulties we faced during the implementation, which should ideally be addressed before considering more complex proof-nets, such as the ones we consider in the rest of this thesis (defined in Section 4.1.1). There were three of them: manipulating graphs, using isomorphisms and the computation time of Coq itself.

5.2.1 Graph manipulation & Desequentialization

To give a glimpse into the complexity of formal graph manipulation, let us take a look at the implementation of the desequentialization function. As one can observe in Definition 5.7, the cases of ⅋-, ⊗- and *cut*-vertices are very similar. It is then natural to factorize the definition by handling these three cases with one definition: *add_node*. What we want is to add a ⅋-, ⊗- or *cut*-vertex to a MLL_{uf} graph or pre-proof-structure (in case of a ⊗ or *cut*, we previously do a disjoint union), with a new c-vertex (excepted in the case of a *cut*) and removing the first two arcs in the list *order*.

We define a set with 3 elements to make the cases.

```
Inductive trilean :=
  | tens_t
  | parr_t
  | cut_t.
```

We can then define an *add_node* function to this end. It takes as input a *base_graph* named *G* and two of its arcs, *e0* and *e1*, and returns a new *base_graph*. The idea is that *e0* and *e1* are the premises of the c-vertices to replace. Thus, what is returned is the original graph where have been added a sub-graph *graph_node* made of two vertices linked by an edge, or a single vertex in the *cut* case, as well as two new edges from the sources of *e0* and *e1* to *target_node*, the latter being the new ⅋-, ⊗- or *cut*-vertex.

```
Definition add_node_graph_1 (t : trilean) {G : base_graph} (e0 e1 : edge G) :=
  let graph_node (t' : trilean) := match t' with
  | tens_t ⇒ edge_graph (⊗) (tens (flabel e0) (flabel e1), true) c
  | parr_t ⇒ edge_graph (⅋) (parr (flabel e0) (flabel e1), true) c
```

```
  | cut_t ⇒ unit_graph cut
  end in
  let G1 (t' : trilean) := G ⊎ graph_node t' in
  let target_node := match t return G1 t with
  | tens_t ⇒ inr (inl tt)
  | parr_t ⇒ inr (inl tt)
  | cut_t ⇒ inr tt
  end in
  G1 t +̇ [inl (source e0), (flabel e0, true), target_node]
        +̇ [inl (source e1), (flabel e1, match t with | cut_t ⇒ true | _ ⇒ false end), target_node].
```

In the above code, $\uplus$ is the disjoint union of graphs and $G \dot{+} [v, l, u]$ adds to G an arc labeled l from v to u.

One then has to remove the two (c-)vertices targeted by *e0* and *e1*, which can be done through taking an induced sub-graph by *induced* applied to some set of vertices.

```
Definition add_node_graph (t : trilean) {G : base_graph} (e0 e1 : edge G) :=
  induced ([set: add_node_graph_1 t e0 e1] :\ inl (target e0) :\ inl (target e1)).
```

This defined the wished MLL_{uf} graph. As can be seen from the code snippets, there are already some complications because vertices of G_1 are not exactly vertices of $G_1 \uplus G_2$: one uses *inl* to see a vertex of the first as one of the second. The same has to be done for arcs, and thus for paths. Nonetheless, these definitions are easy thanks to the simplicity of the definition of *base_graph*.

Defining this transformation between MLL_{uf} pre-proof-structures is a little harder, because we have to prove that the new arc to a c-vertex (if any) is still in the induced sub-graph. We thus need some intermediate steps. First, we remove from the list *order* the arcs removed when taking the induced sub-graph, that in a proof-structure are exactly *e0* and *e1*.

```
Definition add_node_order_1 {G : graph_data} (e0 e1 : edge G) :=
  [seq x ← order G | (target x != target e0) && (source x != target e0) &&
                     (target x != target e1) && (source x != target e1)].
```

Then, we see these arcs as arcs of the union graph, before removing some vertices in the *induced* step.

```
Definition add_node_type_order (t : trilean) {G : graph_data} (e0 e1 : edge G) (l : seq (edge G)) :
  seq (edge (add_node_graph_1 t e0 e1)) := [seq Some (Some (inl e)) | e ← l].
```

Afterwards, we add the arc targeting the new c-vertex at the head of this list, excepted in the case of a *cut*-vertex.

```
Definition add_node_order_2 (t : trilean) {G : graph_data} (e0 e1 : edge G) :=
  match t return seq (edge (add_node_graph_1 t e0 e1)) with
  | cut_t ⇒ [::] | _ ⇒ [:: Some (Some (inr None))] end
  ++ add_node_type_order t e0 e1 (add_node_order_1 e0 e1).
```

Now, we need to see these arcs as ones of the final *base_graph*. To this end, we have to prove they are in the kept induced sub-graph, meaning we prove the following (which is easy).

```
Lemma add_node_consistent_order (t : trilean) {G : graph_data} (e0 e1 : edge G) :
  let S := [set: add_node_graph_1 t e0 e1] :\ inl (target e0) :\ inl (target e1) in
  all (pred_of_set (edge_set S)) (add_node_order_2 t e0 e1).
```

Remark that if in the definition of *add_node_graph* we had first removed vertices and arcs before adding new ones (instead of the converse), then we would have need a proof that the sources and targets of the arcs to add belong to the induced sub-graph.

Thus, we obtain the wanted list of arcs of the graph *add_node_graph t e0 e1*.

```
Definition add_node_order (t : trilean) {G : graph_data} (e0 e1 : edge G) :
  seq (edge (add_node_graph t e0 e1)) :=
  sval (all_sigP (add_node_consistent_order t e0 e1)).
```

The pre-proof-structure with its list of arcs is finally:

```
Definition add_node_graph_data (t : trilean) {G : graph_data} (e0 e1 : edge G) :
  graph_data := {|
  graph_of := add_node_graph t e0 e1;
  order := add_node_order _ _ _;
  |}.
```

Then, one has to prove that if the original graph was a proof-structure, then the one obtained this way also is one, and so on.

As should be visible on this example, preserving structures can become tedious easily. This would be worst with a left function defined exactly on $\parr$- and $\otimes$-vertices, because we would need proofs that such a vertex is still labeled in the same way afterwards, and so on.

The above implementation is a more general definition than strictly needed for desequentialization, with some useless cases to take into account, but is easier to write. Generalizing adding a $\otimes$, $\parr$ or *cut*-vertex was simpler than transporting proofs everywhere, and proving thrice that this operation preserves being a proof-structure for a $\parr$, a $\otimes$ and a *cut*.

5.2.2 Isomorphisms

When proving the sequentialization theorem, the hard part on paper is finding a terminal splitting vertex, *i.e.* proving the equivalent of Lemma 4.31. Then, proving that such a vertex can be removed to get one or two proof-nets – results equivalent to Lemmas 4.46 and 4.48 – is trivial enough to be implicit, and is often not even mentioned on paper. On computer, it is the inverse: the proof there is a terminal splitting vertex is not much harder than on paper, the true difficulty for sequentialization was proving the implicit part, which translates on computer to a most tedious task!

As an example, consider a splitting terminal $\otimes$-vertex v. As it belongs to no cycle, removing it results in two connected components, which correspond to two proof-nets. Formally, one has to define those, for instance by looking at all vertices having a path from them to the source of the left premise of v, not having v as an internal vertex. Then one has to use this complex definition to prove that adding a c-vertex in place of v in this graph indeed yields a proof-net, with a proof of each property in the definition of a proof-structure.

Furthermore, once this tedious part is done on both graphs, one still has to give an isomorphism between the original graph and the two proof-nets to which one adds v back! In other words, one must prove that taking the two proof-nets constructed this way, and adding between them a $\otimes$-vertex, as done in the definition of desequentialization, gives the same graph as the starting one, up to isomorphism. To do so, one has to explicitly give a bijection between vertices, one between arcs, and prove they are bijections.

Coding this procedure for a $\otimes$-, a $\parr$- and a *cut*-vertex was the longest part of the proof of sequentialization, and doing so for just one kind of vertex is longer than proving the existence of a terminal splitting vertex! See the organization of the code in Section 5.3, with a single file for proving our version of Yeo's theorem, and a file per kind of vertex to prove a terminal splitting one is sequentializing.

5.2.3 Computation time of Coq

Some problems encountered during this formalization where due, not to the formalization on its own, but to the internal mechanisms of Coq itself. They mainly came from the manipulation of complex graphs. We detail here such an example, that can be found in the file bug_report_1_bis.v of the code, which is reproduced in Appendix C.

Let us consider a simple operation: adding a vertex "in the middle of an arc", namely transform an arc $v \to u$ into $v \to x \to u$ with x a new vertex. Doing this as we would on paper is doable with the graph library: one removes the involved arc, adds a vertex and two new arcs with the right sources and targets.

```
Variables (Lv Le : Type) (cut : Lv).

Definition extend_edge_graph (G : graph Lv Le) (e : edge G) (R : Lv) (As At : Le) :
  graph Lv Le :=
  remove_edges [set e : edge G] ∔ R ∔ [inl (source e), As, inr tt] ∔ [inr tt, At, inl (target e)].
```

Repeating twice this basic operation is easy and can be used for studying the ax-key case of cut-elimination in this framework of MLL_{uf} proof-nets. Alas, the graph obtained this way is hardly manipulable in Coq!

```
Definition new_graph (G : graph Lv Le) (e : edge G) :=
  (@extend_edge_graph
    (@extend_edge_graph G e cut (elabel e) (elabel e))
    None cut (elabel e) (elabel e)).
```

The problem comes from the fact that deleting an arc or a vertex introduces a dependent type: arcs of G where we remove an arc a are given by the dependent type corresponding to $\{b \in \mathcal{E} \mid b \neq a\}$. Repeating the operation yields a dependent type whose main type is itself a dependent type, as in $\{b \in \{b \in \mathcal{E} \mid b \neq a\} \mid b \neq (c, \pi)\}$ with π a proof that $c \neq a$. In such a graph, if we consider a particular arc, it will be an object of the following form:

```
Some (Some (inl (Sub (Some (Some (inl (Sub a p1)))) p2)))
```

where *a* is an arc of the original graph, *p1* a proof this arc is not the removed *e* and *p2* one it is not the added then removed intermediate edge *None*. Not only is this not immediate to read and understand, Coq also does not manage to manipulate it in a reasonable time! Consider for instance the following function, used to consider an arc of the starting graph as one of the new one:

```
Fail Time Definition transport_to_new (G : graph Lv Le) (e : edge G) :
  edge G → edge (new_graph e) :=
  fun a ⇒
    if @boolP (a \notin [set e]) is AltTrue p1 then
      if @boolP (Some (Some (inl (Sub a p1))) \notin [set None]) is AltTrue p2 then
        Some (Some (inl (Sub (Some (Some (inl (Sub a p1)))) p2)))
      else None
    else None.
```

After more than ten minutes on my machine, Coq still did not manage to check it is well typed! This is clearly not quick enough to be used in an interactive proof assistant, and this execution time simply does not allow to manipulate this graph. A bandage solution is to give the explicit dependent types at some points when giving such an arc, as in the following:

```
Definition transport_to_new (G : graph Lv Le) (e : edge G) :
  edge G → edge (new_graph e) :=
  fun a ⇒
```

```
if @boolP (a \notin [set e]) is AltTrue p1 then
  if @boolP ((Some (Some (inl (Sub a p1 : edge (remove_edges [set e : edge G])))) : edge
(@extend_edge_graph G e cut (elabel e) (elabel e))) \notin [set None]) is AltTrue p2 then Some
(Some (inl (Sub (Some (Some (inl (Sub a p1 : edge (remove_edges [set e : edge G])))) : edge
(@extend_edge_graph G e cut (elabel e) (elabel e))) p2 : edge (remove_edges [set None : edge
(@extend_edge_graph G e cut (elabel e) (elabel e))])))))
  else None
else None.
```

This is incredibly both tedious and fragile. The source of this problem seems to be on the unification side, with the time needed for Coq to check that an object indeed is an arc of the graph being in the order of the tens of minutes, or even hours in more complex graphs!

On a more general note, doing non-trivial manipulations of graphs with successive additions and deletions quickly becomes a nightmare to write, read and understand. As we add dependent types, option types or union types for each operation, one gets edges like those in the above code snippets or like *Some* (*Some* (*inl* (*inl* (*Some* (*inl* (*exist* *e* *E*)))))). This is an incredibly confusing way to see an edge from the original graph as the corresponding one in the resulting graph after the transformations. This difficulty seems hard to solve, because on paper one can by abuse consider a same vertex or arc before and after a transformation, but formally these are distinct objects.

5.3 Organization of the code

The code developed during this thesis has not been yet incorporated into the Yalla library, as there are still some improvements to be made. It can be find here: `https://github.com/RemiDiG/proofnet_mll`. To give an overview of the size of this project, here is detailed the structure of the files, at the time this thesis is written, ordered by dependency.

mll_prelim.v general results, with no relation to linear logic;

graph_more.v some general lemmas about GraphTheory;

upath.v development of part of GraphTheory, with undirected paths in a multigraph;

supath.v development of part of GraphTheory, with undirected arc-simple paths in a multigraph;

simple_upath.v development of part of GraphTheory, with undirected vertex-simple paths in a multigraph;

graph_uconnected_nb.v proof of Corollary 3.48, that an acyclic graph has for number of connected components its number of vertices minus its number of arcs, generalized to our notion of f-paths;

mgraph_dag.v development of part of GraphTheory, about directed acyclic multigraph;

mgraph_tree.v proof that in an acyclic graph, there is at most one simple path between two vertices;

yeo.v proof of one of our generalizations of Yeo theorem (Theorem 3.29);

mll_def.v definition of proof-nets, with its intermediate definitions, along basic results and reformulations;

mll_basic.v general results following from the definition of a proof-net;

mll_correct.v defines some basic graph operations and proves they respect the correctness criterion;

mll_seq_to_pn.v definition of the desequentialization function, proof it produces proof-nets, and that each step respects isomorphisms (for instance, taking two isomorphic graphs and adding a $\parr$-vertex between isomorphic arcs yield isomorphic graphs) – the latter being used for sequentialization;

mll_pn_to_seq_def.v definition of a sequentializing vertex as the inverse of desequentializing;

mll_pn_to_seq.v proof that a proof-net contains a terminal splitting vertex, using Yeo theorem;

mll_pn_to_seq_ax.v a terminal *ax*-vertex is sequentializing;

mll_pn_to_seq_cut.v a splitting *cut*-vertex is sequentializing;

mll_pn_to_seq_parr.v a terminal $\parr$-vertex is sequentializing;

mll_pn_to_seq_tens.v a terminal splitting $\otimes$-vertex is sequentializing;

mll_pn_to_seq_th.v proof of the sequentialization theorem from the previous results;

mll_cut.v definition of cut-elimination, proof it respects correction, and that it terminates;

mll_ax_exp.v work in progress about axiom-expansion in proof-nets;

unused.v general results that became useless during the development;

bug_report_x.v files with examples where Coq compiles for a very long time.

5.4 Perspectives

We implemented in Coq a definition of proof-nets for MLL_{uf} with two main results of this theory: a proof of the sequentialization theorem, and a normalization of cut-elimination. The current implementation could be simplified in several ways, considering for instance only vertex-simple paths instead of switching and left-switching paths. It was not done this way from the beginning because the proof of sequentialization used was not found yet when starting this project! This change of the sequentialization proof during the development induces some fragility and a revision of the code for homogenization purpose is needed before a proper integration in the Yalla library. Likewise, it remains to simplify the results on undirected paths for them to be integrated in GraphTheory.

Many concepts and results could be added to this formalization of proof-nets. To cite only a few: axiom-expansion, links with sequent calculus regarding cut-elimination and rule commutation, adding the mix_2- and mix_0-rules, ... Nonetheless, this project shows that formalizing proof-nets can be done in a not so tedious fashion. Extending this work to $\mathsf{MELL}^{0,2}$ or $\mathsf{MALL}^{0,2}_{uf}$ proof-nets seems doable, because the main difficulties should be on the manipulation of graphs, meaning the same as for MLL_{uf} proof-nets, which were solved here.

Part III

Isomorphisms & Retractions

In this part, we use the previous concepts and results (proof-nets, cut-elimination is Church-Rosser modulo rule commutation, etc) in order to study *isomorphisms* and *retractions* in some sub-systems of linear logic. These relations can be defined in a very generic way in category theory, using only basic notions (see for instance [Mac, Chapter 1, Section 5]).

The concept of isomorphism is central in category theory, because it corresponds to indistinguishable objects. For instance, the unicity of a limit holds only up to isomorphism. A natural question is then what is the quotient implicitly done by all the results with an "up to isomorphism". As this is not easy using the definition (in particular proving that two objects are not isomorphic is hard), a good result characterizing isomorphisms is an *equational theory* corresponding exactly to them. Such a theory makes it easier to check whether two objects are isomorphic or not, and its equations are the "basic" isomorphisms generating all others.

Meanwhile, retractions give a natural notion of sub-typing, for the underlying couple of morphisms allows to encode an element of the sub-type as one of the super-type thanks to the first morphism, and to decode it back using the second morphism. Here again, one can wish for a characterization by means of an inequational theory.

Our two main results are the following. We give in Chapter 6 a characterization of isomorphisms in MALL^2 and show they are the same as in MALL, extending a result for MLL by Balat and Di Cosmo [BD99]. This implies a characterization for isomorphisms in $\star$-autonomous categories with finite products. The obtained characterization is the expected one, which is a much richer equational theory than in MLL, involving not only associativity, commutativity and unitality laws but also distributivity and cancellation laws. The unit-free case is obtained by relying on the proof-net syntax of Hughes & Van Glabbeek (Definition 4.1), while units are handled in sequent calculus thanks to confluence of cut-elimination up to rule commutation (Theorem 2.49). The idea behind this methodology is that using the canonical representation of proofs that are proof-nets is preferable when proofs are considered up to cut-elimination and axiom-expansion, *i.e.* up to rule commutation: the proof-net syntax with one graph per proof is more suitable that the usual sequent calculus representation, where one has to consider equivalence classes for rule commutation, which muddles the reasoning. Unfortunately, as we know no proof-net syntax for MALL with units, this is only possible in the unit-free case. Our second contribution is stated in Chapter 7, with some findings about retractions in MLL, and in particular a characterization of retractions to an atom. The first problem is a simpler one (isomorphism) in a more complex setting (MALL), while the second problem is a harder one (retraction) in a simpler framework (MLL).

Chapter 6

Isomorphisms for Multiplicative-Additive Linear Logic

The question of type isomorphisms consists in trying to understand when two types in a type system, or two formulas in a logic, are "the same". The general question can be described in category theory: two objects A and B are isomorphic, denoted $A \simeq B$, if there exist morphisms $A \xrightarrow{f} B$ and $B \xrightarrow{g} A$ such that $f \circ g = \mathrm{id}_B$ and $g \circ f = \mathrm{id}_A$; *i.e.* if the following diagram commutes:

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \arrow[r, bend left, "f"] & B \arrow[l, bend left, "g"] \arrow[loop right, "\mathrm{id}_B"] \end{tikzcd}$$

The arrows f and g are the underlying isomorphisms. Given a (class of) category, the question is then to find equations characterizing when two objects A and B are isomorphic (in all instances of the class). The focus here is on pairs of isomorphic objects rather than on the isomorphisms themselves. For example, in the class of cartesian categories, one finds the following isomorphic objects: $A \times B \simeq B \times A$, $(A \times B) \times C \simeq A \times (B \times C)$ and $A \times \top \simeq A$. Regarding type systems and logics, one can instantiate the categorical notion. For instance in typed λ-calculi: two types A and B are isomorphic if there exist two λ-terms $M : A \to B$ and $N : B \to A$ such that $\lambda x : B.(M\ (N\ x)) =_{\beta\eta} \lambda x : B.x$ and $\lambda x : A.(N\ (M\ x)) =_{\beta\eta} \lambda x : A.x$ where $=_{\beta\eta}$ is $\beta\eta$-equality. This corresponds to isomorphic objects in the syntactic category generated by terms up to $=_{\beta\eta}$. Similarly, type isomorphisms can also be considered in logic, following what happens in the λ-calculus through the Curry-Howard correspondence: simply replace λ-terms with proofs, types with formulas, β-reduction with cut-elimination and η-expansion with axiom-expansion. In this way, type isomorphisms are studied in a wide range of theories, such as category theory [Sol83; DP97], λ-calculus [Di95] and proof theory [BD99]. They have been used to develop practical tools, such as search in a library of a functional programming language [Rit91; ARG21].

Following the definition, it is usually easy to prove that the type isomorphism relation is a congruence. It is then natural to look for an equational theory generating this congruence. Testing whether or not two types are isomorphic is then much easier. An equational theory $\mathcal{T}$ is called **sound** with respect to type isomorphisms if types equal up to $\mathcal{T}$ are isomorphic. It is said **complete** if it equates any pair of isomorphic types. Given a (class of) category, a type system or a logic, our goal is to find an associated sound and complete equational theory for type isomorphisms. Such a characterization is not always possible as the induced theory may not be finitely axiomatizable (see for instance [FDB02]). As an example, according to [Sol83], the following equational theory is sound and complete for isomorphisms of cartesian closed categories:

Commutativity	Associativity	Curryfication	Distributivity
$A \times B = B \times A$	$A \times (B \times C) = (A \times B) \times C$	$(A \times B) \to C = A \to (B \to C)$	$A \to (B \times C) = (A \to B) \times (A \to C)$
	$A \times 1 = A$	$1 \to A = A$	$A \to 1 = 1$

Soundness is usually the easy direction as it is sufficient to exhibit pairs of terms corresponding to each equation. The completeness part is often harder, and there are in the literature two main approaches to solve this problem. The first one is a semantic method, relying on the fact that if two types are isomorphic then they are isomorphic in all (denotational) models. One thus looks for a model in which isomorphisms can be computed (more easily than in the syntactic model) and are all included in the equational theory under consideration – this is the approach used in [Sol83; Lau05] for example. Finding such a model simple enough for its isomorphisms to be computed, but still complex enough not to contain isomorphisms absent in the syntax is the difficulty. The second method is the syntactic one, which consists in studying isomorphisms directly in the syntax. The analysis of pairs of terms composing to the identity should provide information on their structure and then on their type so as to deduce the completeness of the equational theory – see for example [Di95; BD99]. The more easily the equality ($=_{\beta\eta}$ for example) between proof objects can be computed, the easier the analysis of isomorphisms will be.

As we work in the framework of linear logic, the underlying question becomes "is there an equational theory corresponding to the isomorphisms between formulas in this logic?". In MLL, which corresponds to $\star$-autonomous categories, the question of type isomorphisms was answered positively using a syntactic method based on proof-nets by Balat and Di Cosmo [BD99]: isomorphisms emerge from associativity and commutativity of the multiplicative connectives $\otimes$ and $\parr$, as well as unitality of the multiplicative units 1 and $\perp$. The question was also solved for the polarized fragment of LL by my PhD advisor using game semantics [Lau05]. It is conjectured that isomorphisms in full linear logic (with the $\overset{oe}{\mapsto}$ transformation as well as identification of proofs up to $! - ?_c$ and $! - ?_w$ commutations, see Remark 1.15) correspond to those in its polarized fragment, constituted of the equations on Table 6.1 together with the four exponential isomorphisms $!(A \,\&\, B) \simeq {!A} \otimes {!B}$, $?(A \oplus B) \simeq {?A} \parr {?B}$, $!\top \simeq 1$ and $?0 \simeq \perp$. As a step towards solving this conjecture, we prove the type isomorphisms in MALL^2 are generated by the equational theory of Table 6.1. This applies at the same time to the class of $\star$-autonomous categories with finite products (whose corresponding equational theory in its usual syntax is on Table 6.2 on Page 240). This situation is much richer than in MLL since isomorphisms include not only associativity, commutativity and unitality, but also the distributivity of the multiplicative connective $\otimes$ (resp. $\parr$) over the additive $\oplus$ (resp. $\&$) as well as the associated cancellation laws for the additive unit 0 (resp. $\top$) over the multiplicative connective $\otimes$ (resp. $\parr$).

Using a semantic approach for completeness looks difficult here. In particular, most of the known "concrete" models of MALL (with the meaning that isomorphisms are more easily computed inside) immediately come with unwanted isomorphisms not valid in the syntax. For example, one can check that $\top \otimes A \simeq \top$ in coherent spaces [Gir87], while it is plainly obvious that $\top \otimes A \not\simeq \top$ in the syntax as the second formula is provable but generally the first one is not. An idea would then be to consider less "concrete" models of MALL or of $\star$-autonomous categories with finite products, typically categories built in a natural way over MLL or $\star$-autonomous categories. For instance,

$\mathcal{L}^\dagger$				
Commutativity	Associativity	Distributivity	Unitality	Cancellation
$A \otimes B = B \otimes A$	$A \otimes (B \otimes C) = (A \otimes B) \otimes C$	$A \otimes (B \oplus C) = (A \otimes B) \oplus (A \otimes C)$	$A \otimes 1 = A$	$A \otimes 0 = 0$
$A \parr B = B \parr A$	$A \parr (B \parr C) = (A \parr B) \parr C$	$A \parr (B \,\&\, C) = (A \parr B) \,\&\, (A \parr C)$	$A \parr \bot = A$	$A \parr \top = \top$
$A \oplus B = B \oplus A$	$A \oplus (B \oplus C) = (A \oplus B) \oplus C$		$A \oplus 0 = A$	
$A \,\&\, B = B \,\&\, A$	$A \,\&\, (B \,\&\, C) = (A \,\&\, B) \,\&\, C$		$A \,\&\, \top = A$	
$\mathcal{L}$				

Table 6.1: Type isomorphisms in multiplicative-additive linear logic

one could consider the free completion by products and coproducts of $\star$-autonomous categories, hoping the resulting category corresponds to MALL – in the same spirit as what has been done for bicompletion of $\star$-autonomous categories [Joy95a; Joy95b], except MALL should not be the free bicompletion of MLL as it is not expected to have all limits. Or one could consider models based on coherence spaces such as [HJ97; HJ99; BHS05]. We see two main difficulties with this approach. The first is that isomorphisms may not be that easy to compute in these categories, and we lack results giving isomorphisms of a completion from isomorphisms of the base category. Especially in our case, where we have distributivity isomorphisms involving both multiplicative and additive connectives, meaning isomorphisms of MALL are not directly deducible from those of MLL and ALL. The second obstacle is that these categories may not correspond exactly to MALL, and can have unwanted isomorphisms. In particular, it is possible that the models built following coherence spaces contain the same unwanted $\top \otimes A \simeq \top$.

For this reason we prefer to use a syntactic method, and follow the approach from Balat and Di Cosmo [BD99] based on proof-nets. Hence, our main tool will be the proof-nets defined in Section 4.1. Indeed, the canonicity of proof-nets makes it a very good syntax for linear logic where studying composition of proofs by cut, cut-elimination and identity of proofs is very natural, highly simplifying the problem by going from $=_{\beta\eta}$ to plain equality of graphs. However, already in [BD99] some trick had to be used to deal with units as proof-nets are working perfectly only in MLL_{uf}. Handling the units thus has to be done in sequent calculus, and in particular involves studying proofs up to rule commutation.

Reasoning Our proof of completeness can be sketched as follows.

(1) A simple but key idea is to use the distributivity, unitality and cancellation equations of Table 6.1 from left to right so as to rewrite formulas into a *distributed* form, where none of these equations can be applied anymore. Between such distributed formulas, the only isomorphisms left should be commutativity and associativity ones. Furthermore, units should not play any special role. All the problem now is to prove this is the case.

(2) Working in sequent calculus, we prove that in any isomorphism between distributed formulas one can indeed replace units by fresh atoms and the resulting formulas are still isomorphic. The main difficulty here is that this only holds for distributed formulas, as generally units do not behave like atoms at all. This leads us to identify the following *patterns* in proofs of

isomorphisms:

$$\frac{}{\vdash \top, 0}\ (\top) \qquad \frac{\frac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$$

Once proven that unit rules are constrained to these patterns, it is easy to replace each of their uses by an *ax*-rule. A problem here is that we have to consider proofs up to cut-elimination, hence up to rule commutation (using results from Chapter 2). In particular, instead of the second pattern above, one must consider the following more general one, with a sequence of $\oplus_1$- and $\oplus_2$-rules between the 1- and $\bot$-rules, that turns the 1-formula into a more complex one F:

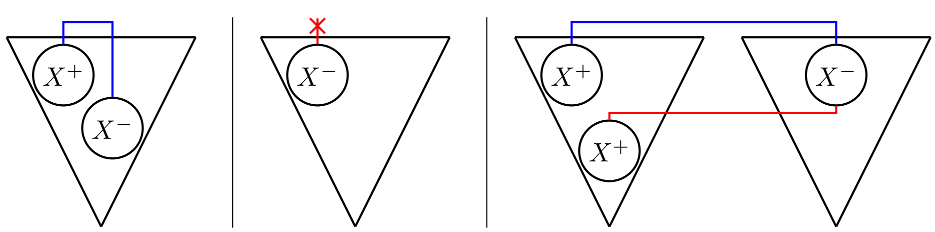
$$\frac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash F}\ \oplus_i}{\vdash \bot, F}\ (\bot)$$

Moreover, we need two main steps to get these patterns.

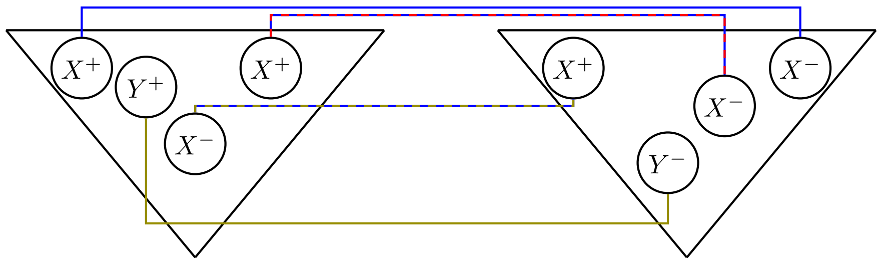
(2a) First, we prove these patterns are in the identity proofs and are preserved by all rule commutations, so as to know the equivalence class of the identity has them. This is done by giving properties preserved by rule commutations, with the tedious work of checking each rule commutation.

(2b) Then, we transpose these patterns backward through cut-elimination to get them in the proofs of isomorphisms. This needs a detailed study to prove that cutting proofs of an isomorphism cannot "completely erase" a unit rule, *i.e.* that the behavior of the $\& - \oplus_i$ key cases and of the $\top - cut$ commutative cases are quite constrained. We do so by a thorough consideration of the evolution of *slices* when eliminating a *cut*-rule, as well as a meticulous monitoring of $\top$-rules during the reduction.

(3) Once the problem reduced to the unit-free case, we can transpose isomorphisms to proof-nets, so as to not have to bother with rule commutations and equivalence classes anymore. There, we prove that proof-nets associated to isomorphisms have very particular shapes, with local constraints on their axiom links. More precisely, we prove that the three following configurations are forbidden: an axiom link between atoms of a same formula, an atom with no axiom link on it and an atom with axiom links to several distinct atoms.



This means that proof-nets associated to an isomorphism $A \simeq B$ have the following general shape, with each atom of A linked to a unique atom of B by means of one or several axiom links:



This is the main challenge, because having such a shape is again only true for isomorphisms between distributed formulas, so we have to use this global property on distributed formulas in order to deduce a local property on the shape of axiom links. We employ here the correctness criterion (P3) in a direct way. The idea is to prove that a forbidden configuration, typically the third one, implies having in one formula a & being an ancestor of a ⅋. The distributivity hypothesis then gives a $\otimes\backslash\oplus$-vertex between these two, thanks to which a cycle contradicting the correctness criterion can be built. This crucial part of the proof seems very hard to transpose in sequent calculus as it hinges on a geometric reasoning, and even if it were possible we would expect some heavy work on rule commutations, turning this already complicated proof into an utterly bewildering one.

(4) Knowing that proof-nets have very specific shapes, we recognize they correspond only to a rearrangement of formulas because axiom links bind an atom of a formula with one of the isomorphic formula in a bijective fashion. This is enough to finally conclude that the only isomorphisms are commutativity and associativity ones.

Outline Once the definition of isomorphism given and soundness of the equational theory proven (Section 6.1), our proof of the completeness of the equational theory of Table 6.1 goes in two main steps. First, we work in the sequent calculus to simplify the problem and reduce it to the unit-free sub-system, by lack of a good-enough notion of proof-nets for $\mathsf{MALL}^{0,2}$ in the presence of additive units. Using results from Chapter 2, we analyze the behavior of units inside isomorphisms to conclude that they can be replaced with fresh atoms, once formulas are simplified appropriately (Section 6.2, points (1) and (2) of the sketch). Secondly, being in a setting admitting proof-nets, we adapt the proof of Balat & Di Cosmo [BD99]: we transpose the definition of isomorphisms to this new syntax, then prove completeness (Section 6.3, points (3) and (4) of the sketch). This last step is the core of the work and requires a precise analysis of the structure of proof-nets because of the richer structure induced by the presence of the additive connectives. The situation is much more complex than in the multiplicative setting since for example sub-formulas can be duplicated through distributivity equations, breaking a linearity property crucial in [BD99]. Finally, seeing MALL as a category, we extend our result to conclude that Table 6.1 (or more precisely its adaptation to the language of categories, Table 6.2 on Page 240) provides the equational theory of isomorphisms valid in all $\star$-autonomous categories with finite products (Section 6.4). We discuss the situation of products in symmetric monoidal closed categories as well. As an aside, we observe what happens when identifying proofs not up to $\beta\eta$-equality but only β-equality, and show the corresponding notion of isomorphism is trivial (Section 6.5).

6.1 Linear Isomorphisms

Definition 6.1 (Isomorphism)**.** Consider (a sub-system of) linear logic with $\overset{o}{\vdash\!\dashv}$ some (possibly none, or all) of the Rétoré transformations. Two formulas A and B are **isomorphic** (in this sub-system), denoted $A \simeq B$, if there exist proofs π of $\vdash A^\perp, B$ and π' of $\vdash B^\perp, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A$ and $\pi' \overset{A}{\bowtie} \pi =_{\beta\eta o} ax_B$:

$$\pi \overset{B}{\bowtie} \pi' \;=\; \dfrac{\overset{\pi}{\vdash A^\perp, B} \qquad \overset{\pi'}{\vdash B^\perp, A}}{\vdash A^\perp, A}\,(cut) \;=_{\beta\eta o}\; \dfrac{}{\vdash A^\perp, A}\,(ax) \;=\; ax_A$$

and

$$\pi' \overset{A}{\bowtie} \pi \;=\; \dfrac{\overset{\pi'}{\vdash B^\perp, A} \qquad \overset{\pi}{\vdash A^\perp, B}}{\vdash B^\perp, B}\,(cut) \;=_{\beta\eta o}\; \dfrac{}{\vdash B^\perp, B}\,(ax) \;=\; ax_B$$

One can check that isomorphisms in linear logic form a congruence (Appendix B), so that it makes sense to search for a corresponding equational theory. We aim to prove that two MALL^2 (resp. MALL^2_{uf}) formulas are isomorphic if and only if they are equal in the equational theory $\mathcal{L}$ (resp. $\mathcal{L}^\dagger$) defined as follows.

Definition 6.2 (Equational theories $\mathcal{L}$ and $\mathcal{L}^\dagger$)**.** We denote by $\mathcal{L}$ the equational theory given in Table 6.1 on Page 199, while $\mathcal{L}^\dagger$ denotes the part not involving units, *i.e.* with commutativity, associativity and distributivity equations only.

Given an equational theory $\mathcal{T}$, the notation $A =_{\mathcal{T}} B$ means that formulas A and B are equal in the theory $\mathcal{T}$. As often, the soundness part is easy (but tedious) to prove.

Theorem 6.3 (Isomorphisms soundness, see [Lau05, Lemma 3])**.** *If $A =_{\mathcal{L}} B$ then $A \simeq B$.*

Proof. It suffices to give the proofs for each equation, then check their compositions can be reduced by cut-elimination to an axiom-expansion of an ax-rule. For instance, looking at the commutativity of $\parr$, *i.e.* $A \parr B \simeq B \parr A$, we set π and π' the following proofs:

$$\pi = \dfrac{\dfrac{\dfrac{}{\vdash B^\perp, B}\,(ax) \qquad \dfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash B^\perp \otimes A^\perp, B, A}\,(\otimes)}{\vdash B^\perp \otimes A^\perp, B \parr A}\,(\parr) \quad \textit{and} \quad \pi' = \dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax) \qquad \dfrac{}{\vdash B^\perp, B}\,(ax)}{\vdash A^\perp \otimes B^\perp, A, B}\,(\otimes)}{\vdash A^\perp \otimes B^\perp, A \parr B}\,(\parr)$$

One can check

$$\dfrac{\overset{\pi}{\vdash B^{\perp} \otimes A^{\perp}, B \parr A} \qquad \overset{\pi'}{\vdash A^{\perp} \otimes B^{\perp}, A \parr B}}{\vdash B^{\perp} \otimes A^{\perp}, A \parr B}\,(cut) \xrightarrow{\beta *} \dfrac{\dfrac{\dfrac{}{\vdash B^{\perp}, B}\,(ax) \qquad \dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash B^{\perp} \otimes A^{\perp}, A, B}\,(\otimes)}{\vdash B^{\perp} \otimes A^{\perp}, A \parr B}\,(\parr) \xleftarrow{\eta} \dfrac{}{\vdash B^{\perp} \otimes A^{\perp}, A \parr B}\,(ax)$$

and

$$\dfrac{\overset{\pi'}{\vdash A^{\perp} \otimes B^{\perp}, A \parr B} \qquad \overset{\pi}{\vdash B^{\perp} \otimes A^{\perp}, B \parr A}}{\vdash A^{\perp} \otimes B^{\perp}, B \parr A}\,(cut) \xrightarrow{\beta *} \dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax) \qquad \dfrac{}{\vdash B^{\perp}, B}\,(ax)}{\vdash A^{\perp} \otimes B^{\perp}, B, A}\,(\otimes)}{\vdash A^{\perp} \otimes B^{\perp}, B \parr A}\,(\parr) \xleftarrow{\eta} \dfrac{}{\vdash A^{\perp} \otimes B^{\perp}, B \parr A}\,(ax)$$

□

All the difficulty lies in the proof of the other implication, completeness, on which the rest of this chapter focuses.

6.2 Reduction to unit-free distributed formulas

As written at the beginning of this chapter, our plan is to use proof-nets so as to benefit from the canonicity of this syntax. Unfortunately, proof-nets currently exist only for atomic-axiom proofs in $\mathsf{MALL}^{0,2}_{uf}$. Using previous results about axiom-expansion from Section 2.2, one easily reduces the problem to atomic-axiom proofs (Section 6.2.1). In parenthesis, *conservativity* then easily follows: isomorphisms of MALL between MLL or ALL formulas are isomorphisms of MLL or ALL (Section 6.2.2). Now, it remains to take care of the units. The main idea is that the multiplicative and additive units can be replaced by fresh atoms for the study of isomorphisms. However, this is not true in general, for instance we have $(1 \oplus A) \otimes B \simeq B \oplus (A \otimes B)$ (using the soundness theorem, Theorem 6.3), and this isomorphism uses that 1 is unital for $\otimes$. Hence, we begin by reducing the problem to so-called *distributed* formulas, using the distributivity, unitality and cancellation equations of Table 6.1 on Page 199 (Section 6.2.3). Then, for theses special formulas, we identify patterns containing units in proofs equal to an identity up to rule commutation, patterns which can be lifted to proofs of isomorphisms (Section 6.2.4). Finally, we prove that, in an isomorphism between distributed formulas, replacing units by fresh atoms preserves being an isomorphism, reducing the study of isomorphisms to distributed formulas in the unit-free sub-system (Section 6.2.5). This section is fully about sequent calculus, and will then allow us to use proof-nets.

6.2.1 Reduction to atomic-axiom proofs

One can simplify a little the problem, considering only normal forms for proofs and forgetting about axiom-expansion. This can be done in full propositional logic, *i.e.* with all optional rules and any of the Rétoré transformations, but without the quantifier rules.

Definition 6.4. We write $A \overset{\pi,\pi'}{\simeq} B$ when the *atomic-axiom, cut-free,* $\overset{o}{\rightsquigarrow}$*-normal* proofs π of $\vdash A^\perp, B$ and π' of $\vdash B^\perp, A$ respects $\pi \overset{B}{\bowtie} \pi' =_{\beta o} \mathrm{id}_A$ and $\pi \overset{A}{\bowtie} \pi' =_{\beta o} \mathrm{id}_B$.

Lemma 6.5. *Set π and π' proofs respectively of $\vdash A^\perp, B$ and $\vdash B^\perp, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A$. Then, letting ϕ (resp. ϕ') be a result of expanding all axioms, eliminating all cuts and applying all possible $\overset{o}{\rightsquigarrow}$ in π (resp. π'), we have $\phi \overset{B}{\bowtie} \phi' =_{\beta o} \mathrm{id}_A$.*

Proof. We have $\phi \overset{B}{\bowtie} \phi' =_{\beta\eta o} \pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A \overset{\eta^*}{\longleftarrow} \mathrm{id}_A$. In particular, $\phi \overset{B}{\bowtie} \phi' =_{\beta\eta o} \mathrm{id}_A$. Using Proposition 2.10, we get $\eta(\phi \overset{B}{\bowtie} \phi') =_{\beta o} \eta(\mathrm{id}_A)$. However, ϕ, ϕ' and id_A are atomic-axiom, hence $\eta(\phi \overset{B}{\bowtie} \phi') = \phi \overset{B}{\bowtie} \phi'$ and $\eta(\mathrm{id}_A) = \mathrm{id}_A$. Thus, $\phi \overset{B}{\bowtie} \phi' =_{\beta o} \mathrm{id}_A$ follows. □

Lemma 6.6. *Given two formulas A and B, $A \simeq B$ if and only if there exist proofs π and π' such that $A \overset{\pi,\pi'}{\simeq} B$.*

Proof. The converse way follows by definition of an isomorphism. For the direct way, take proofs π and π' given by the definition of an isomorphism. One can expand all axioms, eliminate all cuts and apply all possible $\overset{o}{\rightsquigarrow}$ in π and π' (Proposition 2.8 and Theorem 2.15) to obtain respectively ϕ and ϕ'. Using Lemma 6.5 twice, $\phi \overset{B}{\bowtie} \phi' =_{\beta o} \mathrm{id}_A$ and $\phi \overset{A}{\bowtie} \phi' =_{\beta o} \mathrm{id}_B$. Hence, $A \overset{\phi,\phi'}{\simeq} B$. □

6.2.2 Conservativity

Our study of cut-elimination from Chapter 2 is enough to obtain **conservativity** of isomorphisms of MALL over MLL and ALL: if $A \simeq B$ is an isomorphism in MALL with formulas A and B in MLL (resp. ALL), then $A \simeq B$ is also an isomorphism in MLL (resp. ALL). Notice the reciprocal follows immediately from definitions. While this result will not be used to characterize all isomorphisms of MALL, it is interesting by itself as its proof is easy and generalizes to many systems of linear logic, *e.g.* could be used to prove conservativity of isomorphisms of LL over MALL (admitting our Church-Rosser result from Chapter 2 generalizes to this framework).

Lemma 6.7 (Conservativity)**.** *Isomorphisms of* MALL *are conservative over* MLL *and* ALL.

Proof. We prove it only for MLL, the same proof holding for ALL by only replacing these acronyms. Consider $A \simeq B$ in MALL with A and B also formulas of MLL. By Lemma 6.6, one gets MALL proofs π and π' such that $A \overset{\pi,\pi'}{\simeq} B$ in MALL. By the sub-formula property (Fact 1.7), π and π' are also MLL proofs. The only difficulty left is that cut-elimination in MALL may equalize more MLL proofs than the simpler cut-elimination in MLL, so that $\pi \overset{B}{\bowtie} \pi' =_\beta \mathrm{id}_A$ and $\pi \overset{A}{\bowtie} \pi' =_\beta \mathrm{id}_B$ in MALL but not in MLL. We prove it is not the case, and more generally that when two MLL proofs ϕ and ϕ' are β-equal in MALL, they also are β-equal in MLL, which concludes the proof. First, one can *w.l.o.g.* assume ϕ and ϕ' cut-free: otherwise, just reduce all cuts in these proofs, these reductions involving only cut-elimination steps of MLL. By Theorem 2.47, $\phi \overset{r^*}{\longmapsto} \phi'$ using rule commutation in

MALL. But rule commutation preserves that a proof in MLL stays in MLL, because these are the cut-free rule commutations, so any rule they may introduce is on a connective of the conclusion sequent, thus is a rule of MLL. Thence, the sequence $\phi \overset{r^*}{\vdash\!\dashv} \phi'$ involves only rule commutations and proofs from MLL. We conclude $\phi =_\beta \phi'$ in MLL thanks to Proposition 2.46 (or more exactly its analog for MLL, whose proof is contained in the one we did for MALL). ☐

6.2.3 Reduction to distributed formulas

Definition 6.8 (Distributed formula)**.** A formula is **distributed** if it does not have any sub-formula of the form

$$A \otimes (B \oplus C) \quad (A \oplus B) \otimes C \quad A \otimes 1 \quad 1 \otimes A \quad A \oplus 0 \quad 0 \oplus A \quad A \otimes 0 \quad 0 \otimes A$$

or their duals

$$(C \,\&\, B) \parr A \quad C \parr (B \,\&\, A) \quad \bot \parr A \quad A \parr \bot \quad \top \,\&\, A \quad A \,\&\, \top \quad \top \parr A \quad A \parr \top$$

(where A, B and C are any formulas).

Remark 6.9. This notion is stable under duality: if A is distributed, so is $A^\perp$.

Definition 6.10. We call $\mathfrak{D}$ the following abstract rewriting system, on the set of MALL formulas.

$$\begin{array}{ccc|ccc}
A \otimes (B \oplus C) & \to & (A \otimes B) \oplus (A \otimes C) & (C \,\&\, B) \parr A & \to & (C \parr A) \,\&\, (B \parr A) \\
(A \oplus B) \otimes C & \to & (A \otimes C) \oplus (B \otimes C) & C \parr (B \,\&\, A) & \to & (C \parr B) \,\&\, (C \parr A) \\
\hline
\end{array}$$

$$\begin{array}{ccc|ccc|ccc|ccc}
A \otimes 1 & \to & A & 1 \otimes A & \to & A & A \parr \bot & \to & A & \bot \parr A & \to & A \\
A \oplus 0 & \to & A & 0 \oplus A & \to & A & A \,\&\, \top & \to & A & \top \,\&\, A & \to & A \\
A \otimes 0 & \to & 0 & 0 \otimes A & \to & 0 & A \parr \top & \to & \top & \top \parr A & \to & \top
\end{array}$$

Lemma 6.11. *The rewriting system $\mathfrak{D}$ is strongly normalizing, with as normal forms distributed formulas.*

Proof. That normal forms of $\mathfrak{D}$ are distributed formulas follows by Definitions 6.8 and 6.10. For the strong normalization, it suffices to give a function f from formulas to natural numbers such that $A \to B$ implies $f(A) > f(B)$. We define it by induction as follows:

$$\begin{array}{lllll}
f(X^+) & = & f(X^-) & = & 2 \\
f(A \parr B) & = & f(A \otimes B) & = & f(A) \times f(B) \\
f(\bot) & = & f(1) & = & 2 \\
f(A \,\&\, B) & = & f(A \oplus B) & = & f(A) + f(B) + 1 \\
f(\top) & = & f(0) & = & 2
\end{array}$$

One can easily check that for any formula A, $f(A) \geq 2$, and then that the wished property holds. Here are some representative cases:

$$f(A \otimes (B \oplus C)) = f(A) \times (f(B) + f(C) + 1)$$
$$> f(A) \times f(B) + f(A) \times f(C) + 1 = f((A \otimes B) \oplus (A \otimes C))$$

$$f(A \otimes 1) = f(A) \times 2 > f(A)$$

$$f(A \oplus 0) = f(A) + 3 > f(A)$$

$$f(A \otimes 0) = f(A) \times 2 > f(0)$$

☐

Remark 6.12. The rewriting system $\mathfrak{D}$ is not confluent. Indeed, consider the formula $(X \oplus Y) \otimes (Z \oplus W)$. It has two normal forms, according to whether the first rewriting applied is the rule $A \otimes (B \oplus C) \to (A \otimes B) \oplus (A \otimes C)$ or $(A \oplus B) \otimes C \to (A \otimes C) \oplus (B \otimes C)$. In the first case, one get

$$(X \oplus Y) \otimes (Z \oplus W) \to ((X \oplus Y) \otimes Z) \oplus ((X \oplus Y) \otimes W)$$

whose unique normal form is $((X \otimes Z) \oplus (Y \otimes Z)) \oplus ((X \otimes W) \oplus (Y \otimes W))$. In the second case, we have

$$(X \oplus Y) \otimes (Z \oplus W) \to (X \otimes (Z \oplus W)) \oplus (Y \otimes (Z \oplus W))$$

of unique normal form $((X \otimes Z) \oplus (X \otimes W)) \oplus ((Y \otimes Z) \oplus (Y \otimes W))$.

Nonetheless, $\mathfrak{D}$ without the four upper rewriting rules in Definition 6.10 is confluent; in particular $\mathfrak{D}$ is confluent when restricted to MLL or to ALL. Furthermore, while $\mathfrak{D}$ is not confluent, it is Church-Rosser modulo associativity and commutativity of the four connectives.

Proposition 6.13. *If the equational theory denoted $\mathcal{L}$ (recall Table 6.1 on Page 199) is complete for isomorphisms between distributed formulas, then it is complete for isomorphisms between arbitrary formulas.*

Proof. Consider an isomorphism $A \simeq B$ between two arbitrary formulas A and B. Let A_d and B_d be associated distributed formulas, obtained as normal forms of the rewriting system $\mathfrak{D}$ (using Lemma 6.11). As each rule of this rewriting system corresponds to a valid equality in the theory $\mathcal{L}$, we have $A =_{\mathcal{L}} A_d$ and $B =_{\mathcal{L}} B_d$.

By soundness of $\mathcal{L}$ (Theorem 6.3) and as linear isomorphism is a congruence, we deduce $A_d \simeq A \simeq B \simeq B_d$. The completeness hypothesis on $\mathcal{L}$ for distributed formulas yields $A_d =_{\mathcal{L}} B_d$ from $A_d \simeq B_d$. Thus $A =_{\mathcal{L}} A_d =_{\mathcal{L}} B_d =_{\mathcal{L}} B$. □

Therefore, we can now consider only distributed formulas. We will use this to study units, as well as when solving the unit-free case to prove there are only commutativity and associativity isomorphisms left.

6.2.4 Patterns in distributed isomorphisms

In this section, we prove units in distributed isomorphisms are in very specific sets of rules, which are the following patterns.

Definition 6.14 (Patterns)**.**

- We call a $\top/0$**-pattern** the following sub-proof:

$$\dfrac{}{\vdash \top, 0}\ (\top)$$

- Likewise, we set $1/\bot$**-pattern** the sub-proof:

$$\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$$

- A $1/\oplus/\bot$**-pattern** is a 1-rule followed by a (possibly empty) sequence ρ of $\oplus_i$-rules and finally a $\bot$-rule:

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash F}\,\rho}{\vdash \bot, F}\,(\bot)$$

The main goal of this part is proving $\top$- 1- and $\bot$-rules in proofs of distributed isomorphisms belong to these patterns. The two $\top/0$- and $1/\bot$-patterns are those we truly wish for. However, we will not find directly $1/\bot$-patterns, but the more generic $1/\oplus/\bot$-patterns; this is not a problem, for using $\oplus_i - \bot$ commutations turns a $1/\oplus/\bot$-pattern into a $1/\bot$-pattern.

We will need at some point that the following sequents cannot be proved.

Fact 6.15. *The sequents* $\vdash$, $\vdash 0$ *and* $\vdash 0, 0$ *have no proof in* MALL^2. *More generally, no sequent made of only* 0 *formulas is provable in* MALL^2.

Proof. If such a proof π exists, then *w.l.o.g.* it is cut-free (Corollary 2.43). The last rule of π can only be a mix_2-rule, leading to two proofs also on these sequents. By induction on the number of rules of π, such a proof cannot exist. □

We will also use the following grammar.

Definition 6.16. Given a formula A, we define the grammar

$$\bigoplus[A] ::= A \mid \bigoplus[A] \oplus B \mid B \oplus \bigoplus[A]$$

where B ranges over all formulas. The occurrence of A which is the base case of this grammar is called its distinguished occurrence.

We first analyze the behavior of units in proofs equal to id_A up to rule commutation, and prove they belong to these patterns (Section 6.2.4.1). We only do so for a *distributed* formula A as we have already seen it is enough in Section 6.2.3, and as the above patterns hold only for these formulas. We can then obtain our result, for this property is preserved by cut anti-reduction. However, it is not so easy to prove it, and we will consider slices to do so (Section 6.2.4.2). This finally allows us to remove the units (Section 6.2.4.3).

6.2.4.1 Patterns in identities up to rule commutation

We study here patterns containing units in proofs equal to an identity up to rule commutation. The tedious case study on rule commutations and the properties to find in this proof are a great example on why we do not want to do the full proof of completeness in sequent calculus, but wish instead to use proof-nets.

Proposition 6.17. *Let* π *be a proof equal, up to rule commutation, to* id_A *with* A *a* distributed *formula. Then:*

- *the* $\top$*-rules of* π *are in a* $\top/0$*-pattern* $\dfrac{}{\vdash \top, 0}\,(\top)$ *with* $\top$ *in* A *being the dual of* 0 *in* $A^\perp$, *or vice-versa (*$\top$ *in* $A^\perp$ *being the dual of* 0 *in* A*);*

- *$\bot$-rules and 1-rules come by pairs in a $1/\oplus/\bot$-pattern*

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash F}\,\rho}{\vdash \bot, F}\,(\bot)$$

  with $\bot$ in A being the dual of 1 in $A^\perp$, or vice-versa;

- *there is no sequent in π of the shape $\vdash B \,\&\, C$;*

- *there is no mix_2-rule in π.*

Proof. The key idea is to find properties of id_A preserved by all rule commutations and ensuring the properties described in the statement. Hence, we prove a stronger property: any sequent S of a proof π obtained through a sequence of rule commutations from id_A for a distributed formula A respects:

(1) the formulas of S are distributed;

(2) if $\top$ is a formula of S, then $S = \vdash \top, 0$ with 0 in $A^\perp$ the dual of $\top$ if $\top$ is a sub-formula of A (or vice-versa);

(3) if $\bot$ is a formula of S, then $S = \vdash \bot, \bigoplus[1]$ where the distinguished 1 is the dual of $\bot$ in $A^\perp$ if $\bot$ is a sub-formula of A (or vice-versa), and the sub-proof of π above S is a sequence of $\oplus_i$ rules leading to the 1-rule of the distinguished 1, with in addition a $\bot$-rule inside this sequence;

(4) if $B \,\&\, C$ is a formula of S, then $S = \vdash B \,\&\, C, \bigoplus[C^\perp \oplus B^\perp]$ where the distinguished $C^\perp \oplus B^\perp$ is the dual of $B \,\&\, C$ in $A^\perp$ if $B \,\&\, C$ is a sub-formula of A (or vice-versa), and in the sub-proof of π above S the $\oplus$-rules of the distinguished $C^\perp \oplus B^\perp$ are a $\oplus_2$-rule in the left branch of the $\&$-rule of $B \,\&\, C$, and a $\oplus_1$-rule in its right branch;

(5) if S contains several negative formulas or several positive formulas, then its negative formulas are all $⅋$-formulas or negated atoms;

(6) there is no mix_2-rule in π.

Remark that (5) is a corollary of properties (2), (3) and (4). As in $\overset{r}{\vdash\!\dashv}$ there is no commutation with a *cut*-rule (in particular no $cut - \top$ commutation) and no $\otimes - \top$ commutation creating a sub-proof with a *cut*-rule, it follows that π is cut-free and has the sub-formula property, making (1) trivially true. We prove that the identity id_A respects properties (2), (3), (4) and (6), and that these properties are preserved by any rule commutation of $\overset{r}{\vdash\!\dashv}$.

The identity id_A *respects the properties.* We prove by induction on the distributed formula A properties (2), (3) and (4). Notice that sub-formulas of A are also distributed. By symmetry, assume A is positive.

If $A \in \{X^+, 1, 0\}$ where X is an atom, then:

$$\mathrm{id}_A \in \left\{ \dfrac{}{\vdash X^-, X^+}\,(ax) \quad ; \quad \dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash \bot, 1}\,(\bot) \quad ; \quad \dfrac{}{\vdash \top, 0}\,(\top) \right\}$$

Each of these proofs respects (2), (3), (4) and (6).

Assume the result holds for B and C, and that $A = B \otimes C$. Then

$$\mathrm{id}_A = \quad \dfrac{\dfrac{\begin{array}{c}\mathrm{id}_B \\ \vdash B^\perp, B\end{array} \qquad \begin{array}{c}\mathrm{id}_C \\ \vdash C^\perp, C\end{array}}{\vdash C^\perp, B^\perp, B \otimes C}\ (\otimes)}{\vdash C^\perp \parr B^\perp, B \otimes C}\ (\parr)$$

By induction hypothesis, id_A respects (6). We have to prove the sequents $\vdash C^\perp, B^\perp, B \otimes C$ and $\vdash C^\perp \parr B^\perp, B \otimes C$ respect the properties. The latter respects (2), (3) and (4) trivially for it has neither a $\top$, $\perp$ nor $\&$ formula. As $C^\perp \parr B^\perp$ is distributed, it follows that neither $C^\perp$ nor $B^\perp$ can be a $\top$, $\perp$ or $\&$ formula, and as such the former sequent also respects the properties.

Suppose $A = B \oplus C$ with sequents of id_B and id_C respecting the properties. Now

$$\mathrm{id}_A = \quad \dfrac{\dfrac{\begin{array}{c}\mathrm{id}_C \\ \vdash C^\perp, C\end{array}}{\vdash C^\perp, B \oplus C}\ (\oplus_2) \qquad \dfrac{\begin{array}{c}\mathrm{id}_B \\ \vdash B^\perp, B\end{array}}{\vdash B^\perp, B \oplus C}\ (\oplus_1)}{\vdash C^\perp \ \&\ B^\perp, B \oplus C}\ (\&)$$

By induction hypothesis, id_A respects (6). The sequent $\vdash C^\perp \& B^\perp, B \oplus C$ respects (2), (3) and (4), as the $\oplus$ is the dual of the $\&$. By symmetry, we show the properties are also fulfilled by $\vdash B^\perp, B \oplus C$, and they will be respected by $\vdash C^\perp, B \oplus C$ with a similar proof. As the formulas are distributed, $B^\perp$ cannot be a $\top$ formula, hence the sequent respects (2). If $B^\perp$ is not a $\perp$ nor a $\&$ formula, then (3) and (4) hold for $\vdash B^\perp, B \oplus C$. If it is, then using that $\vdash B^\perp, B$ respects (3) and (4), and that $B \oplus C$ belongs to $\bigoplus[B]$, properties (3) and (4) follow for $\vdash B^\perp, B \oplus C$.

Every possible rule commutation preserves the properties. We show that each rule commutation preserves properties (2), (3) and (4), using every time the notations from Tables 1.5 to 1.20 in Definition 1.14, on Pages 37 to 52. Remark that the preservation of (6) is trivial, except for a $C^{\otimes}_{\top}$ or a $C^{mix_2}_{\top}$ commutation as these are the only ones creating rules not already in the proof; hence we will consider no commutation with mix_2, safe in these cases. By symmetry, we treat only one case for $\otimes - \otimes$, $\parr - \otimes$, $\& - \otimes$ and $\oplus_i - \otimes$ commutations, and we also consider only the $\oplus_1$-rule and not the $\oplus_2$ one.

$\top$-*commutations* Using property (2), we cannot do any rule commutation between a $\top$-rule and a $\parr$, $\otimes$, $\&$, $\oplus_i$, $\perp$ or $\top$-rule. The only other commutation is a $\top - mix_2$ one. Still by (2), such a commutation can only be one of the following four:

$$\dfrac{}{\vdash \top, 0}\ (\top) \ \overset{r}{\vdash\!\dashv}\ \dfrac{\dfrac{}{\vdash \top, 0}\ (\top) \quad \begin{array}{c}\pi \\ \vdash\end{array}}{\vdash \top, 0}\ (mix_2) \qquad \dfrac{}{\vdash \top, 0}\ (\top) \ \overset{r}{\vdash\!\dashv}\ \dfrac{\dfrac{}{\vdash \top}\ (\top) \quad \begin{array}{c}\pi \\ \vdash 0\end{array}}{\vdash \top, 0}\ (mix_2)$$

$$\dfrac{}{\vdash \top, 0}\ (\top) \ \overset{r}{\vdash\!\dashv}\ \dfrac{\begin{array}{c}\pi \\ \vdash\end{array} \quad \dfrac{}{\vdash \top, 0}\ (\top)}{\vdash \top, 0}\ (mix_2) \qquad \dfrac{}{\vdash \top, 0}\ (\top) \ \overset{r}{\vdash\!\dashv}\ \dfrac{\begin{array}{c}\pi \\ \vdash 0\end{array} \quad \dfrac{}{\vdash \top}\ (\top)}{\vdash \top, 0}\ (mix_2)$$

However, there is no proof in MALL^2 of $\vdash$ nor of $\vdash 0$ (Fact 6.15). Hence, such a commutation is impossible, meaning one can do no commutation involving a $\top$-rule.

$\bot$-*commutations* Using property (3), we cannot do any rule commutation between a $\bot$-rule and a $\parr$, $\otimes$, $\&$ or $\bot$-rule. A commutation between a $\bot$ and a $\oplus_1$-rule preserves property (3): we have by hypothesis Γ empty and $A \oplus B$ in the grammar for A belongs to it. It also respects (2) and (4) trivially. A similar result holds for a $\bot - \oplus_2$ commutation.

$C^{\parr}_{\parr}$ *commutation* We have to show the properties for $\vdash A, B, C \parr D, \Gamma$. Because $\vdash A \parr B, C \parr D, \Gamma$ respects them, negative formulas of Γ are $\parr$-formulas or negated atoms by (5). By distributivity, if A (or B) is a negative formula, then it must be a $\parr$ one or a negated atom. Thus, $\vdash A, B, C \parr D, \Gamma$ fulfills (2), (3) and (4).

$C^{\oplus_j}_{\oplus_i}$ *commutation* We treat only a $C^{\oplus_1}_{\oplus_2}$ commutation, the others being similar. We show the properties for $\vdash A, C \oplus D, \Gamma$. As $\vdash A \oplus B, C \oplus D, \Gamma$ respects them, negative formulas of Γ are $\parr$-formulas or negated atoms by (5). If A is positive, a $\parr$ or a negated atom, then we are done. Otherwise, as $\vdash A, D, \Gamma$ fulfills the properties, it follows Γ is empty and D of the desired shape. By (1), D is not 0, thus A is not $\top$. Whether A is $\bot$ or $\&$, the sequent $\vdash A, C \oplus D$ respects the properties.

$C^{\otimes}_{\otimes}$ *commutation* We have to show the properties for $\vdash A \otimes B, C, \Gamma, \Delta$. Because they are respected by $\vdash A \otimes B, C \otimes D, \Gamma, \Delta, \Sigma$, negative formulas of Γ and Δ are $\parr$-formulas or negated atoms by (5). If C is positive, a $\parr$ or a negated atom, then we are done. Otherwise, as $\vdash B, C, \Delta$ fulfills the properties, it follows Δ is empty and C of the desired shape, so C is a 0, 1 or $\oplus$-formula. This is impossible as $C \otimes D$ is distributed by (1).

$C^{\&}_{\&}$, $C^{\parr}_{\&}$, $C^{\&}_{\parr}$, $C^{\otimes}_{\&}$ *and* $C^{\&}_{\otimes}$ *commutations* These cases are impossible by property (4).

$C^{\oplus_1}_{\&}$ *and* $C^{\&}_{\oplus_1}$ *commutations* Remember we consider only the $\oplus_1$-rule by symmetry. In these cases, (4) for $\vdash A \& B, C \oplus D, \Gamma$ implies Γ empty and $C \oplus D$ of the desired shape. Thus C of the desired shape, for $C \oplus D$ is not the distinguished formula as it has the same rule $\oplus_1$ in both branches of the $\&$-rule, proving the result for $\vdash A \& B, C$. For $\vdash A, C \oplus D$ (and similarly $\vdash B, C \oplus D$), A cannot be a $\top$ by (1), and if it is a $\bot$ or a $\&$, then the hypothesis on $\vdash A, C$ implies that the properties are also respected in $\vdash A, C \oplus D$.

$C^{\oplus_1}_{\parr}$ *and* $C^{\parr}_{\oplus_1}$ *commutations* Let us show the properties for $\vdash A, B, C \oplus D, \Gamma$ in the first commutation and $\vdash A \parr B, C, \Gamma$ in the second. As they hold for $\vdash A, B, C, \Gamma$, negative formulas in $\vdash A, B, C, \Gamma$ are $\parr$-formulas or negated atoms by (5) and the result follows.

$C^{\oplus_1}_{\otimes}$ *commutation* We have to prove $\vdash A, C \oplus D, \Gamma$ respects the properties. Because $\vdash A \otimes B, C \oplus D, \Gamma, \Delta$ fulfills them, negative formulas of Γ are $\parr$ or negated atoms by (5). If A is a negative formula other than a $\parr$ or an atom, then for $\vdash A, C, \Gamma$ respects the properties we have that Γ is empty and C of the desired shape. By (1), C is not a 0, so A is not a $\top$. But then $C \oplus D$ also has the wished shape for A, and $\vdash A, C \oplus D$ fulfills the properties.

$C^{\otimes}_{\oplus_1}$ *commutation* We prove $\vdash A \otimes B, C, \Gamma, \Delta$ respects the properties. As they are fulfilled by $\vdash A \otimes B, C \oplus D, \Gamma, \Delta$, negative formulas of Γ and Δ are $\parr$ or atoms by (5). As $A \otimes B$ is distributed by (1), A cannot be a 0, 1 nor $\oplus$ formula, so by $\vdash A, C, \Gamma$ fulfilling the properties it follows that C cannot be a negative formula other than a $\parr$ or an atom. The conclusion follows.

$C^{\otimes}_{\parr}$ *commutation* We prove the properties for $\vdash A, B, C \otimes D, \Gamma, \Delta$. As they are respected by $\vdash A \parr B, C \otimes D, \Gamma, \Delta$, according to (5) negative formulas of Γ and Δ can only be $\parr$-formulas or atoms. As $A \parr B$ is distributed by (1), A and B are positive or $\parr$-formulas or atoms. The conclusion follows.

$C^{\parr}_{\otimes}$ *commutation* We prove the properties for $\vdash A \parr B, C, \Gamma$. As $\vdash A, B, C, \Gamma$ respects them, by (5) negative formulas of Δ and C can only be $\parr$-formulas or atoms, proving the result.

Therefore, we proved id_A respects these properties, and that they are preserved by all rule commutations. The conclusion follows. $\square$

6.2.4.2 Surgery on slices

Recall the definition of slices, Definition 1.6 on Page 28. Cut-elimination can be extended from proofs to slices except that some reduction steps produce failures for slices: when a $\&_i$ faces a $\oplus_i$ and conversely. The reduction of the slice

$$\dfrac{\dfrac{\vdash A, \Gamma}{\vdash A \oplus B, \Gamma}\,(\oplus_1) \qquad \dfrac{\vdash B^\perp, \Delta}{\vdash B^\perp \,\&\, A^\perp, \Delta}\,(\&_1)}{\vdash \Gamma, \Delta}\,(cut)$$

is a failure since the selected sub-formulas of $A \oplus B$ and its dual do not match. More precisely, we have a failure exactly when reducing a key case between $\&_i$ and $\oplus_j$ for $i = j$ – when $i \neq j$ (*i.e.* $j = 1 - i$) the cut reduces correctly as follows:

$$\dfrac{\dfrac{\vdash A_i, \Gamma}{\vdash A_1 \oplus A_2, \Gamma}\,(\oplus_i) \qquad \dfrac{\vdash A_i^\perp, \Delta}{\vdash A_2^\perp \,\&\, A_1^\perp, \Delta}\,(\&_{1-i})}{\vdash \Gamma, \Delta}\,(cut) \quad \xrightarrow{\beta} \quad \dfrac{\vdash A_i, \Gamma \qquad \vdash A_i^\perp, \Delta}{\vdash \Gamma, \Delta}\,(cut)$$

Given two slices $s \in \mathcal{S}(\pi)$ and $r \in \mathcal{S}(\rho)$ with respective conclusions $\vdash A, \Gamma$ and $\vdash A^\perp, \Delta$, their composition by cut $s \overset{A}{\bowtie} r$ reduces either to a slice of a normal form of $\pi \overset{A}{\bowtie} \rho$ or to a failure.

Lemma 6.18. *Let π_1 and π_2 be proofs such that $\pi_1 \xrightarrow{\beta} \pi_2$. For each $s_2 \in \mathcal{S}(\pi_2)$, there exists $s_1 \in \mathcal{S}(\pi_1)$ such that $s_1 \xrightarrow{\beta} s_2$ or $s_1 = s_2$. Reciprocally, for each $s_1 \in \mathcal{S}(\pi_1)$, s_1 reduces to a failure or there exists $s_2 \in \mathcal{S}(\pi_2)$ such that $s_1 \xrightarrow{\beta} s_2$ or $s_1 = s_2$.*

Proof. We can check that each cut-elimination step respects this property, with the equality case coming from a reduction in $\pi_1 \xrightarrow{\beta} \pi_2$ on rules not in the considered slice s_2 (resp. s_1). $\square$

Corollary 6.19. *Let π_1, π_2 and τ be proofs such that $\pi_1 \overset{A}{\bowtie} \pi_2 \xrightarrow{\beta^*} \tau$. For each $s \in \mathcal{S}(\tau)$, there exist $s_1 \in \mathcal{S}(\pi_1)$ and $s_2 \in \mathcal{S}(\pi_2)$ such that $s_1 \overset{A}{\bowtie} s_2 \xrightarrow{\beta^*} s$.*

Proof. By induction on the sequence $\pi_1 \overset{A}{\bowtie} \pi_2 \xrightarrow{\beta^*} \tau$, using Lemma 6.18. $\square$

Lemma 6.20. *Let π_1 and π_2 be cut-free proofs of $\vdash \Gamma$ with $\top$-rules all in $\top/0$-patterns. Assume that $\pi_1 \overset{r^*}{\longmapsto\!\!\!\!\dashv} \pi_2$, where in this sequence there is no rule commutation involving a $\top$-rule. Then, for each slice $s_1 \in \mathcal{S}(\pi_1)$, there exists a unique $s_2 \in \mathcal{S}(\pi_2)$ such that s_1 and s_2 make the same choices for additive connectives in Γ.*

Proof. This can be easily checked for each equation in $\overset{r}{\vdash\!\dashv}$, save for those involving $\top$ (precisely for the $\top - \&$, $\top_{\otimes}$ and $\top - mix_2$ commutations). We then conclude by induction on the length of the sequence. $\square$

Take a choice $\mathcal{C}$ of premises for some additive connectives of a sequent, *i.e.* an additive-resolution of the sequent (defined with proof-nets in Section 4.1.1). We say a slice is **on** $\mathcal{C}$ if for each $\&_i$ and $\oplus_i$-rule in this slice, the chosen premise for the corresponding connective in $\mathcal{C}$ is also i.

Lemma 6.21. *Given a choice $\mathcal{C}$ of premises for additive connectives of A (but not $A^\perp$), there exists a unique slice of $\mathcal{S}(\mathrm{id}_A)$ on it, which furthermore makes on $A^\perp$ the dual choices of $\mathcal{C}$.*

Proof. Direct induction on A, following the definition of id_A on Table 1.1. $\square$

We now prove a partial reciprocal to Corollary 6.19. Notice in the following statement that the assumption is on the composition over A, and the conclusion on the existence of a slice for the composition over B, the other formula.

Lemma 6.22. *Let π and π' be cut-free proofs respectively of $\vdash A^\perp, B$ and $\vdash B^\perp, A$, whose composition over A reduces to id_B up to rule commutation. Set ρ a β-normal form of $\pi \overset{B}{\bowtie} \pi'$. Then, for any slice s of π, there exists a slice s' of π' such that $s \overset{B}{\bowtie} s'$ reduces to a slice of ρ.*

Proof. Take $s \in \mathcal{S}(\pi)$, and denote by $\mathcal{C}$ the choices made in s on $\&$ and $\oplus$ connectives of the formula B. We will use that the composition over the formula A reduces to id_B up to $\overset{r^*}{\vdash\!\dashv}$ to find $s' \in \mathcal{S}(\pi')$ that makes the dual choices of $\mathcal{C}$ on additive connectives of $B^\perp$. This ensures no failure happens during the reduction of $s \overset{B}{\bowtie} s'$, which thus reduces to a slice of ρ (Lemma 6.18).

By hypothesis, call τ a cut-free proof resulting from cut-elimination of $\pi \overset{A}{\bowtie} \pi'$, with $\tau \overset{r^*}{\vdash\!\dashv} \mathrm{id}_B$. By Lemma 6.21, there is a (unique) slice of id_B with choices $\mathcal{C}$ on B and dual choices $\mathcal{C}^\perp$ on $B^\perp$. Applying Proposition 6.17, all proofs in the sequence $\tau \overset{r^*}{\vdash\!\dashv} \mathrm{id}_B$ have $\top$-rules with only 0 in their context, and there cannot be any commutation involving a $\top$-rule in this sequence. Using Lemma 6.20, there is a slice t of τ with choices $\mathcal{C}$ and $\mathcal{C}^\perp$. According to Corollary 6.19, we have slices $r \in \mathcal{S}(\pi)$ and $r' \in \mathcal{S}(\pi')$ whose composition on A reduces to t. In particular, r makes choices $\mathcal{C}$ on B and r' choices $\mathcal{C}^\perp$ on $B^\perp$, as these choices are those in the resulting slice t, and no reversed cut-elimination step can modify them (a $\top - cut$ commutative case can erase such choices, but taking it in the other direction can only create choices). Therefore, r' makes on $B^\perp$ the dual choices of s on B, and we can take this slice as s'. $\square$

The previous result, Lemma 6.22, will be used to prove that given a non-ax-rule r in π, r is not erased in all slices by $\& - \oplus_i$ key cases during normalization. More precisely, taking a slice s containing r, of principal connective in, say $A^\perp$, the lemma gives a slice $s' \in \mathcal{S}(\pi')$ that has no failure for a composition over B. As r does not introduce a cut formula, the only way for it to be erased during the reduction is by a $\top - cut$ commutative case, in which case its principal formula becomes a sub-formula of the sequent on which a $\top$-rule is applied. This will be enough to conclude in our cases, as we know that the resulting normal form only has $\top$-rules of the shape $\dfrac{}{\vdash \top, 0}\,(\top)$ (using Theorem 2.50 and Proposition 6.17).

Lemma 6.23. *Let r be a non-ax rule in a cut-free slice s of conclusion $\vdash \Gamma, A$, with the conclusion sequent of r of the shape $\vdash \Sigma_\Gamma, \Sigma_A$ where Σ_Γ is a sub-sequent of Γ and Σ_A of A. Also assume the principal formula (if any) of r belongs to Σ_Γ,* i.e. *is not a sub-formula of A. Take s' a cut-free slice* *of conclusion* $\vdash A^\perp, \Delta$ *such that* $s \overset{A}{\bowtie} s'$ *reduces to some slice* s''*. Then either there is in* s'' *a rule of the same kind as* r*, applied on a sequent* $\vdash \Sigma_\Gamma, \Theta$ *with the same principal formula (if any) as* r*, or* Σ_Γ *is a sub-sequent of a* $\top$*-rule in* s''*, whose main* $\top$*-formula is not in* Σ_Γ*.*

Proof. By hypothesis, r does not introduce a *cut*-formula, for these formulas are sub-formulas of A or $A^\perp$. Therefore, the only reductions in $s \overset{A}{\bowtie} s' \xrightarrow{\beta^*} s''$ that can erase a rule are $\top - cut$ commutative cases, for the $\&_i - \oplus_j$ key cases in the reduction do not lead to a failure. If no such erasure happens, then we are done: other cut commutative cases involving r may modify its conclusion sequent, but only in the part coming from A, thus Σ_Γ is preserved.

If a $\top - cut$ reduction erases r, then Σ_Γ is in the context of the resulting $\top$-rule t, possibly as a sub-sequent. Furthermore, the principal $\top$-formula of t is not in Σ_Γ. This is enough to conclude, using an induction on the number of steps in $s \overset{A}{\bowtie} s' \xrightarrow{\beta^*} s''$. Notice that t may be erased too during the reduction but, like r, this would lead to Σ_Γ being a sub-sequent of the context of another $\top$-rule. □

6.2.4.3 Patterns in proofs of isomorphisms

A first step is that there is *w.l.o.g.* no mix_2-rule in a proof of an isomorphism, a fact we will use implicitly from now on by assuming we are in MALL instead of MALL^2.

Lemma 6.24. *If* $A \overset{\pi,\pi'}{\simeq} B$ *with* A *and* B *distributed, then there is no* mix_2*-rule in* π *nor in* π'*.*

Proof. Assume towards a contradiction there is a mix_2-rule r in π, and call s a slice this rule r belongs to. The rule r is applied on a sequent $\vdash \Gamma_{A^\perp}, \Gamma_B$, with $\Gamma_{A^\perp}$ occurrences of sub-formulas of $A^\perp$ and Γ_B of B – as the cut-free π has for conclusion sequent $\vdash A^\perp, B$, see Fact 1.7.

Set ρ a normal form of $\pi \overset{B}{\bowtie} \pi'$ (using Corollary 2.43). By Theorem 2.49, $\rho \overset{r^*}{\longmapsto} \mathrm{id}_A$. According to Lemma 6.22, there exists a slice $s' \in \mathcal{S}(\pi')$ such that $s \overset{B}{\bowtie} s'$ reduces to a slice $s'' \in \mathcal{S}(\rho)$. Applying Lemma 6.23 to r, r is either preserved or it is absorbed by a $\top$-rule (during a $\top - cut$ commutative case) and $\Gamma_{A^\perp}$ is in the context of a $\top$-rule, possibly as a sub-sequent, and does not contain the main $\top$-formula of this rule. But by Proposition 6.17, there are no mix_2-rule in ρ, so we are in the second case; moreover, all $\top$-rules are in a $\top/0$-pattern. Thus, $\Gamma_{A^\perp}$ is a sub-sequent of 0, so either the empty context or 0 itself.

By symmetry, doing the same reasoning on $\pi' \overset{A}{\bowtie} \pi$, one find that Γ_B is a sub-sequent of 0. This means the sub-proof in π of root r is a proof of either $\vdash$, $\vdash 0$ or $\vdash 0, 0$. A contradiction, for none of these three sequents is provable (Fact 6.15). □

Lemma 6.25. *If* $A \overset{\pi,\pi'}{\simeq} B$ *with* A *and* B *distributed, then all* $\top$*-rules in* π *and* π' *are in a* $\top/0$*-pattern.*

Proof. Consider t a $\top$-rule $\dfrac{}{\vdash \Gamma_{A^\perp}, \Gamma_B}\,(\top)$ in π, with $\Gamma_{A^\perp}$ occurrences of sub-formulas of $A^\perp$ and Γ_B of B – as the cut-free π has for conclusion sequent $\vdash A^\perp, B$, see Fact 1.7. By symmetry, say the main $\top$-formula of t belongs to $\Gamma_{A^\perp}$, *i.e.* is a sub-formula of $A^\perp$. Call s a slice the $\top$-rule t belongs to.

Set ρ a normal form of $\pi \overset{B}{\bowtie} \pi'$ (using Corollary 2.43). By Theorem 2.49, $\rho \overset{r^*}{\longmapsto} \mathrm{id}_A$. According to Lemma 6.22, there exists a slice $s' \in \mathcal{S}(\pi')$ such that $s \overset{B}{\bowtie} s'$ reduces to a slice $s'' \in \mathcal{S}(\rho)$. Applying Lemma 6.23 to t, t is either preserved and $\Gamma_{A^\perp}$ as well, or t is absorbed by another $\top$-rule (during a $\top - cut$ commutative case) and $\Gamma_{A^\perp}$ stays in the context of a $\top$-rule, possibly as a sub-sequent.

But by Proposition 6.17, the only $\top$-rules of ρ, and so of s'', are $\dfrac{}{\vdash \top, 0}\ (\top)$ rules, with $\top$ being the dual occurrence of 0. Thus, $\top$ and 0 are not both sub-formula of $A^\perp$, and it follows $\Gamma_{A^\perp}$ is either a sub-sequent of $\top$ or one of 0. For $\Gamma_{A^\perp}$ contains $\top$, we conclude that $\Gamma_{A^\perp}$ is $\top$. Moreover, it follows that t had not been erased during a $\top - cut$ commutative case, using Lemma 6.23 – otherwise there would be at least two $\top$-formulas in the resulting $\top$-rule. This implies that Γ_B cannot be empty: if it were, t could not commute with any cut-rule, as it could not do a $\top - cut$ commutative case for its sequent does not have a cut formula, which is a sub-formula of B, and no other cut-elimination step can change this – excepted a $\top - cut$ commutation erasing the rule, which cannot happen here. Thus, if Γ_B were empty then t would be a rule of s'', impossible as it would be a $\dfrac{}{\vdash \top}\ (\top)$ rule, and thus not in a $\top$/0-pattern.

Similarly, set τ a normal form of $\pi \overset{A}{\bowtie} \pi'$. By Theorem 2.49 and Lemmas 6.22 and 6.23, t is either preserved during the reduction (in a slice) and Γ_B as well, or t is absorbed by another $\top$-rule and Γ_B stays in the context of a $\top$-rule, possibly as a sub-sequent. But t cannot be preserved: its main formula is a sub-formula of $A^\perp$, and in τ there are only occurrences of sub-formulas of B and $B^\perp$. Therefore, Γ_B in τ is in the context of a $\top$-rule, and does not contain its principal $\top$-formula. Using Proposition 6.17 again, the only $\top$-rules of τ are $\dfrac{}{\vdash \top, 0}\ (\top)$ rules, with $\top$ being the dual occurrence of 0. Hence, Γ_B must be a sub-sequent of 0. As Γ_B cannot be empty, Γ_B is 0.

Thus, any $\top$-rule t in π (and π' by symmetry) is of the shape $\dfrac{}{\vdash \top, 0}\ (\top)$. ∎

Lemma 6.26. *Let π be a proof whose $\top$-rules are all in a $\top$/0-pattern. There is a cut-elimination strategy in π whose $\top - cut$ commutative cases are all of the form:*

$$\dfrac{\dfrac{}{\vdash \top, 0}\ (\top) \qquad \dfrac{}{\vdash \top, 0}\ (\top)}{\vdash \top, 0}\ (cut) \quad \xrightarrow{\beta} \quad \dfrac{}{\vdash \top, 0}\ (\top)$$

Proof. Our strategy is the following. First, while we can apply a cut-elimination step which is not a $\top - cut$ nor a $cut - cut$ commutative case, we do such a reduction step. These operations preserve that all $\top$-rules are in $\top$/0 patterns.

If no such reduction is possible, consider a highest cut-rule, *i.e.* one with no cut-rule above it. The only possible cases that can be applied using this cut-rule and rules above it are by hypothesis $\top - cut$ commutative cases. Thus, the cut-rule is below a $\top$-rule, so necessarily one of its premises is $\dfrac{}{\vdash \top, 0}\ (\top)$, with $\top$ not the formula we cut on. But then 0 is the formula we cut on, so there is a $\top$-formula on the other premise; we are in the following situation:

$$\dfrac{\dfrac{}{\vdash \top, 0}\ (\top) \qquad \dfrac{\phi}{\vdash \top, \Gamma}\ r}{\vdash \top, \Gamma}\ (cut)$$

We prove the rule r above the premise $\vdash \top, \Gamma$ of the cut-rule is the $\top$-rule corresponding to the cut formula $\top$. If r were not a $\top$-rule, then one could apply a commutative or ax key case, which cannot be. Plus, r cannot be a $\top$-rule corresponding to another $\top$-formula, because such a rule would have two $\top$-formulas in its conclusion. Thence, r is the $\top$-rule introducing the formula we cut on, and our sub-proof is:

$$\dfrac{\dfrac{}{\vdash \top, 0}\ (\top) \qquad \dfrac{}{\vdash \top, 0}\ (\top)}{\vdash \top, 0}\ (cut)$$

A $\top - cut$ commutative case yields $\dfrac{}{\vdash \top, 0}\,(\top)$ which is the allowed $\top - cut$ reduction step.

This reduction strategy terminates (Proposition 2.38) and reaches a cut-free proof. □

Lemma 6.27. *If $A \overset{\pi,\pi'}{\simeq} B$ with A and B distributed, then there is no sequent of the shape $\vdash D \,\&\, E$ in π (and π').*

Proof. Assume *w.l.o.g.* $D \,\&\, E$ is a sub-formula of $A^\perp$, and let s be a slice containing the sequent $\vdash D \,\&\, E$. Pose ϕ a normal form of $\pi \overset{B}{\bowtie} \pi'$ obtained by following the strategy given by Lemma 6.26. By Lemma 6.22, there exists a slice $s' \in \mathcal{S}(\pi')$ such that $s \overset{B}{\bowtie} s'$ reduces to a slice s'' of ϕ. Since $D \,\&\, E$ is a sub-formula of $A^\perp$, it is not a cut formula during the reduction, and the sequent $\vdash D \,\&\, E$ remains in s'', so in ϕ. Indeed, reducing cuts using these steps preserves having the sequent $\vdash D \,\&\, E$, and there is no failure in the reduction. This is trivial for all steps except $\top - cut$. In the case a $\top - cut$ step, by hypothesis on the reduction strategy, it cannot erase the sequent $\vdash D \,\&\, E$ from the proof. Thus, the sequent $\vdash D \,\&\, E$ belongs to ϕ, which is equal to the identity up to rule commutation (Theorem 2.50). This is impossible by Proposition 6.17. □

Lemma 6.28. *If $A \overset{\pi,\pi'}{\simeq} B$ with A and B distributed, then all $\perp$-rules and 1-rules in π (and in π') belong to $1/\oplus/\perp$-patterns.*

Proof. In π, we look at a possible rule r below a sequent $\vdash \bigoplus[1]$ (Definition 6.16). It cannot be a $\otimes$-rule by distributivity, nor a $⅋$-rule for the sequent has a unique formula, nor a $\&$-rule due to Lemma 6.27 – nor a mix_2-rule by Lemma 6.24. If r is a $\oplus_i$-rule, then we keep a sequent $\vdash \bigoplus[1]$, and if it is a $\perp$-rule then it is one of the required shape.

As a consequence, each 1-rule is followed by some $\oplus_i$-rules and possibly a $\perp$-rule; let us call a **$1/\oplus$-pattern** a 1-rule followed by a maximal such sequence of $\oplus_i$-rules. If a $1/\oplus$-pattern stops without a $\perp$-rule below it, we have only one formula in the conclusion sequent of the proof: impossible as π is a proof of $\vdash A^\perp, B$. Thus, the $\perp$-rule exists and to each 1-rule we can associate a $\perp$-rule leading to a $1/\oplus/\perp$-pattern. Henceforth, there are at least as many $\perp$-rules as 1-rules, and as the patterns they belong to have no $\&$-rule, this also holds in any slice.

Consider a slice s of π. By Lemma 6.22, there exists a slice $s' \in \mathcal{S}(\pi')$ such that $s \overset{B}{\bowtie} s'$ reduces to a slice s'' of ρ, the latter being a normal form of $\pi \overset{B}{\bowtie} \pi'$ obtained by following the strategy given by Lemma 6.26. Moreover, s'' contains as many $\perp$-rules as 1-rules, as in ρ they belong to a $1/\oplus/\perp$-pattern (Theorem 2.50 and Proposition 6.17), so are in the same slices. Furthermore, in s'', each 1 from A (resp. $A^\perp$) corresponds to a $\perp$ from $A^\perp$ (resp. A).

Remark that, in the reduction $s \overset{B}{\bowtie} s' \xrightarrow{\beta^*} s''$, the only steps that may erase a $\perp$ or 1-rule are $\perp - 1$ and $\top - cut$ cases. But a $\top - cut$ commutative case cannot erase non-$\top$-rules by definition of our cut-elimination strategy. Furthermore, a $\perp - 1$ key case erases one 1-rule and one $\perp$-rule. Therefore, with r_s the number of r-rules of a slice s, we have $\perp_s + \perp_{s'} = 1_s + 1_{s'}$ as $\perp_{s''} = 1_{s''}$ and any reduction step in $s \overset{B}{\bowtie} s' \xrightarrow{\beta^*} s''$ preserves this equality. But, by our analysis at the beginning of this proof, $1_s \leq \perp_s$ and $1_{s'} \leq \perp_{s'}$. We conclude $1_s = \perp_s$, *i.e.* that s has as many $\perp$-rules than 1-rules.

However, each 1-rule, belonging to a $1/\oplus/\perp$-pattern, belongs to exactly the same slices as the corresponding $\perp$-rule of the pattern. Hence, a $\perp$-rule not in a $1/\oplus/\perp$-pattern would yield a slice s with strictly more $\perp$-rules than 1-rules (taking s any slice containing this $\perp$-rule). Thus, every $\perp$-rule belongs to a $1/\oplus/\perp$-pattern. □

6.2.5 Completeness with units from unit-free completeness

In $1/\oplus/\bot$-patterns, moving each $\bot$-rule up to the associated 1-rule (which can be done up to β-equality by Proposition 2.46) allows us to consider units as fresh atoms introduced by ax-rules, whence reducing the problem to the unit-free sub-system.

Lemma 6.29. *Set X and Y unsigned atoms, and σ the substitution $[X^+/0, X^-/\top, Y^+/1, Y^-/\bot]$. Take π a proof of $\vdash A^\bot, B$ and ϕ one of $\vdash B^\bot, A$, such that A is distributed and $\pi \overset{B}{\bowtie} \phi =_\beta \mathrm{id}_A$. Assume all $\top$-rules of π and ϕ belong to $\top/0$-patterns while all their 1 and $\bot$-rules belong to $1/\oplus/\bot$-patterns.*

Then there exists proofs π' and ϕ' respectively of $\vdash \sigma(A)^\bot, \sigma(B)$ and $\vdash \sigma(B)^\bot, \sigma(A)$, such that $\pi' \overset{\sigma(B)}{\bowtie} \phi' =_\beta \mathrm{id}_{\sigma(A)}$. Moreover, π' (resp. ϕ') is uniquely determined from π (resp. ϕ'): it is the proof obtained by commuting each $\bot$-rule towards the associated 1-rule in its $1/\oplus/\bot$-pattern, then applying the substitution σ, replacing $\top/0$-patterns and $1/\bot$-patterns by ax-rules respectively on X^+ and Y^+.

Proof. As a first step, we prove that *w.l.o.g.* all 1 and $\bot$-rules of π and ϕ belong to $1/\bot$-patterns. Using $\bot - \oplus_i$ rules commutations to move each $\bot$-rule just below the 1-rule above it, we build π_1 and ϕ_1 such that π_1 and ϕ_1 have $\top$-rules only in $\top/0$-patterns, as well as $\bot$ and 1-rules only in $1/\bot$-patterns. Plus, $\pi \overset{r^*}{\mapsto} \pi_1$ and $\phi \overset{r^*}{\mapsto} \phi_1$, whence $\pi_1 \overset{B}{\bowtie} \phi_1 \overset{r^*}{\mapsto} \pi \overset{B}{\bowtie} \phi$. By Proposition 2.46, $\pi_1 \overset{B}{\bowtie} \phi_1 =_\beta \pi \overset{B}{\bowtie} \phi$. As π_1 (resp. ϕ_1) depends only on π (resp. ϕ), our claim holds.

We reduce cuts in $\pi \overset{B}{\bowtie} \phi$ following a particular strategy, ensuring that the proofs obtained during the reduction have $\top$-rules in $\top/0$ patterns, and $\bot$ and 1-rules in $1/\bot$-patterns. This strategy is a specialization of the one given in the proof of Lemma 6.26.

First, while we can apply a step of cut-elimination which is not a $\top - cut$, $\bot - cut$, $\bot - 1$ or $cut - cut$ case, we do such a reduction step. These operations preserve that all $\top$-rules are in $\top/0$ patterns, and $\bot$ and 1-rules in $1/\bot$-patterns.

If no such reduction is possible, consider a highest cut-rule, *i.e.* one with no cut-rule above it. The only possible cases that can be applied using this cut-rule and rules above it are by hypothesis $\top - cut$, $\bot - cut$ or $\bot - 1$.

- If a $\top - cut$ commutative case can be applied, then the cut-rule is below a $\top$-rule, so necessarily one of its premises is $\dfrac{}{\vdash \top, 0}\,(\top)$, with $\top$ not the formula we cut on. But then 0 is the formula we cut on, so there is a $\top$-formula on the other premise; we are in the following situation:

$$\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\tau}{\vdash \top, \Gamma}\,r}{\vdash \top, \Gamma}\,(cut)$$

  We prove the rule r above the premise $\vdash \top, \Gamma$ of the cut-rule is the $\top$-rule corresponding to the cut formula $\top$. If it were not the case, then r commutes with the cut-rule (because our proofs are atomic-axiom). But r cannot be a $\bot$-rule (which cannot have a $\top$-formula in its context), nor a $\top$-rule corresponding to another $\top$-formula (because such a rule would have two $\top$-formulas in its conclusion). Thence, r is the $\top$-rule introducing the formula we cut on. Thus, our sub-proof is:

$$\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{}{\vdash \top, 0}\,(\top)}{\vdash \top, 0}\,(cut)$$

A $\top - cut$ commutative case yields $\dfrac{}{\vdash \top, 0}\ (\top)$ as if we had done an ax key case.

- If a $\bot - cut$ commutative case can be applied, then the cut-rule is below a $\bot$-rule, so necessarily one of its premises is $\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$, with $\bot$ not the formula we cut on. But then 1 is the formula we cut on, so there is a $\bot$-formula on the other premise; we are in the following situation:

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot) \qquad \dfrac{\tau}{\vdash \bot, \Gamma}\ r}{\vdash \bot, \Gamma}\ (cut)$$

  We prove the rule r above the premise $\vdash \bot, \Gamma$ of the cut-rule is the $\bot$-rule corresponding to the cut formula $\bot$. If it were not the case, then r commutes with the cut-rule. But r cannot be a $\top$-rule (which cannot have a $\bot$-formula in its context), nor a $\bot$-rule corresponding to another $\bot$-formula (because such a rule would have two $\bot$-formulas in its conclusion). Thence, r is the $\bot$-rule introducing the formula we cut on. Thus, our sub-proof is:

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot) \qquad \dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)}{\vdash \bot, 1}\ (cut)$$

  We apply a $\bot - cut$ commutative case, followed by a $\bot - 1$ key case, obtaining $\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$ as if we had done an ax key case.

- No $\bot - 1$ key case can be applied, for 1-rules have below them a $\bot$-rule, so not a cut-rule.

This reduction strategy terminates (Proposition 2.38) and reaches a normal form τ, with $\top$, $\bot$ and 1-rules only in $\top/0$ and $1/\bot$ patterns – this is preserved by our strategy.

Remember σ is the substitution $[X^+/0, X^-/\top, Y^+/1, Y^-/\bot]$. We extend σ to proofs where all $\top$-rules are in $\top/0$ patterns, and $\bot$ and 1-rules in $1/\bot$-patterns, by following the usual definition of substitution on proofs, except when reaching a $\top/0$-pattern or $1/\bot$-pattern, that we replace by an ax-rule respectively on X^+ and on Y^+.

Remark one can reach $\sigma(\tau)$ by cut-elimination from $\sigma(\pi) \overset{\sigma(B)}{\bowtie} \sigma(\phi)$, for the reductions we did on units could as well have been done by ax-key cases: no $\top - cut$, nor $\bot - cut$, nor $\bot - 1$ case was used, except for cases that could be simulated using ax key cases. Moreover, $\sigma(\mathrm{id}_A) = \mathrm{id}_{\sigma(A)}$: we only have to look at what become $\top$, 1 and $\bot$-rules; these rules in id_A are in the identified patterns, and we replace these patterns by ax-rules, hence the claim. Lastly, in $\tau \overset{r^*}{\mapsto} \mathrm{id}_A$ we can assume not to commute any $\bot$-rule because we start and end with 1-rules and $\bot$-rules in $\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \bot, 1}\ (\bot)$ shapes only, and such commutations could only move the $\bot$-rule below or above some $\oplus_i$-rules according to Proposition 6.17. Thus, $\sigma(\tau) \overset{r^*}{\mapsto} \sigma(\mathrm{id}_A) = \mathrm{id}_{\sigma(A)}$. Using Proposition 2.46, it follows $\sigma(\tau) =_\beta \mathrm{id}_{\sigma(A)}$, and therefore $\sigma(\pi) \overset{\sigma(B)}{\bowtie} \sigma(\phi) =_\beta \mathrm{id}_{\sigma(A)}$. $\square$

Lemma 6.30. *Set A and B* distributed *formulas, as well as X and Y two unsigned atoms and $\sigma = [X^+/0, X^-/\top, Y^+/1, Y^-/\bot]$. If $A \simeq B$ then $\sigma(A) \simeq \sigma(B)$.*

Proof. Suppose $A \overset{\pi,\phi}{\simeq} B$ by Lemma 6.6. By Lemmas 6.25 and 6.28, π and ϕ have $\top$-rules only in $\top/0$-patterns and $\perp$ and 1-rules in $1/\oplus/\perp$-patterns. We apply Lemma 6.29: the two proofs given by this result yield $\sigma(A) \simeq \sigma(B)$. □

Theorem 6.31. *Take A and B formulas of* MALL*. Set A' (resp. B') a normal form of A (resp. B) for $\mathfrak{D}$ (recall Definition 6.10 on Page 205, normalizing by Lemma 6.11). Let X and Y be fresh atoms for both A and B, and $\sigma = [X^+/0, X^-/\top, Y^+/1, Y^-/\perp]$. Then $A \simeq B \iff \sigma(A') \simeq \sigma(B')$.*

Proof. Remark that $A \simeq B \iff A' \simeq B'$ as $\simeq$ is an equivalence relation (Lemma B.3), using that the rewriting rules of $\mathfrak{D}$ are between isomorphic formulas (Theorem 6.3).

Assume first that $A' \simeq B'$, and take X and Y any atoms. According to Lemma 6.30, $\sigma(A') \simeq \sigma(B')$.

Reciprocally, suppose $\sigma(A') \simeq \sigma(B')$ with X and Y fresh atoms – for A and B, or equivalently for A' and B'. Then $A' \simeq B'$ follows by substituting X by 0 and Y by 1, as they were fresh, thus $[X^+/0, X^-/\top, Y^+/1, Y^-/\perp][0/X, 1/Y]$ is the identity substitution, and because substitution on an atom preserves isomorphisms (Lemma B.6). □

Theorem 6.32 (Isomorphisms completeness from unit-free completeness)**.** *If $\mathcal{L}^\dagger$ is complete for isomorphisms in* MALL_{uf}*, then $\mathcal{L}$ is complete for isomorphisms in* MALL *(*i.e. $A \simeq B \implies A =_{\mathcal{L}} B$*).*

Proof. Take A and B formulas in MALL, possibly with units, such that $A \simeq B$. We assume A and B to be distributed thanks to Proposition 6.13. It suffices to prove that $A =_{\mathcal{L}^\dagger} B$. By Theorem 6.31, $A[X^+/0, X^-/\top, Y^+/1, Y^-/\perp] \simeq B[X^+/0, X^-/\top, Y^+/1, Y^-/\perp]$, where X and Y are fresh atoms for both A and B. For we assume $\mathcal{L}^\dagger$ to be complete for unit-free isomorphisms, this yields $A[X^+/0, X^-/\top, Y^+/1, Y^-/\perp] =_{\mathcal{L}^\dagger} B[X^+/0, X^-/\top, Y^+/1, Y^-/\perp]$. We conclude $A =_{\mathcal{L}^\dagger} B$ by substituting X by 0 and Y by 1, as X and Y were fresh. □

6.3 Completeness

Our method to prove the completeness of $\mathcal{L}^\dagger$ relates closely to the one used by Balat and Di Cosmo in [BD99], with some more work due to the distributivity isomorphisms. We work on proof-nets, as they highly simplify the problem by representing proofs up to rule commutation [HG16]. Thus, we start by transposing the study of isomorphisms to the syntax of proof-nets, never to speak of sequent calculus again (Section 6.3.1). We then present particular shapes of proof-nets (Section 6.3.2), and study identity proof-nets (Section 6.3.3). Afterwards, we prove isomorphisms yield proof-nets of these particular shapes (Section 6.3.4). Having previously reduced the problem to distributed formulas by Proposition 6.13, we can consider even more constrained proof-nets (Section 6.3.5). These are the key differences with the proof in MLL_{uf} from [BD99], where some properties are given for free as there is no slice nor distributivity isomorphism. After this point the problem is similar to MLL_{uf}, with commutativity and associativity only. We thus conclude with the same method used in [BD99]: restricting the problem to so-called *non-ambiguous* formulas (Section 6.3.6), isomorphisms are easily characterized (Section 6.3.7).

As we adapt and generalize the proof from [BD99], many concepts (bipartite proof-nets, non-ambiguous formulas, renamings, ...) and results given in this section are (adapted) from this paper.

6.3.1 Isomorphisms in proof-nets

The goal of this section is to shift the study of isomorphisms to the syntax of proof-nets. To this aim, we use results proved in Section 4.2, linking proofs of sequent calculus and proof-nets.

Definition 6.33 (Identity proof-net)**.** We call **identity proof-net** of a MALL_{uf} formula A the proof-net $\mathfrak{P}_f(\mathrm{id}_A)$ – namely the unique desequentialization of the axiom-expansion of $ax_A = \dfrac{}{\vdash A^\perp, A}\,(ax)$. By abuse of notations, we also note id_A the identity proof-net of A; it will always be clear from the context whether this corresponds to a proof or to a proof-net.

A notion of isomorphism can be defined directly on proof-nets.

Definition 6.34. We write $A \overset{\theta,\psi}{\simeq} B$ when θ and ψ are two cut-free proof-nets respectively on $A^\perp, B$ and $B^\perp, A$ such that both of their compositions reduce by cut-elimination to identity proof-nets:

$$\theta \overset{B}{\bowtie} \psi \xrightarrow{\beta^*} \mathrm{id}_A \qquad\qquad \psi \overset{A}{\bowtie} \theta \xrightarrow{\beta^*} \mathrm{id}_B$$

In MALL_{uf}, our three notions of isomorphisms are equivalent.

Lemma 6.35. *Set θ and ψ cut-free proof-nets respectively of $A^\perp, B$ and $B^\perp, A$ such that $\theta \overset{B}{\bowtie} \psi$ reduces to id_A. For any proofs π and ϕ (of sequent calculus) such that $\mathfrak{P}_f(\pi) = \theta$ and $\mathfrak{P}_f(\phi) = \psi$, we have $\phi \overset{A}{\bowtie} \pi =_\beta \mathrm{id}_B$.*

Proof. Consider a normal form τ of $\pi \overset{B}{\bowtie} \phi$, existing by Corollary 2.43. By Lemma 4.25, it follows $\mathcal{B}(\theta) \overset{B}{\bowtie} \mathcal{B}(\psi)$ has for normal form $\mathfrak{P}_f(\tau)$. However, cut-elimination in proof-nets in confluent (Theorem 4.12), and by hypothesis $\theta \overset{B}{\bowtie} \psi$ reduces to the identity proof-net of A, which is the desequentialization of the proof id_A. Thence, $\mathfrak{P}_f(\tau) = \mathfrak{P}_f(\mathrm{id}_A)$. Furthermore, using Theorem 4.19, if two proofs have the same proof-net, then they are related by rule commutations: $\tau \overset{r^*}{\longmapsto\!\!\dashv} \mathrm{id}_A$. Thanks to Proposition 2.46, we obtain $\pi \overset{B}{\bowtie} \phi =_\beta \tau =_\beta \mathrm{id}_A$. ☐

Theorem 6.36 (Type isomorphisms in proof-nets)**.** *Set A and B two MALL_{uf} formulas. The followings are equivalent:*

(1) $A \simeq B$

(2) there exists proofs π and π' such that $A \overset{\pi,\pi'}{\simeq} B$

(3) there exists proof-nets θ and θ' such that $A \overset{\theta,\theta'}{\simeq} B$

Proof. That Items (1) and (2) are equivalent is simply Lemma 6.6. We prove that Items (2) and (3) are equivalent by double implication.

Set π and ϕ proofs such that $A \overset{\pi,\phi}{\simeq} B$. Applying Lemma 4.25 twice, and as $\mathfrak{P}_f(\mathrm{id}_A)$, $\mathfrak{P}_f(\mathrm{id}_B)$, $\mathfrak{P}_f(\pi)$ and $\mathfrak{P}_f(\phi)$ are cut-free (Fact 4.17) and the unique elements respectively of $\mathfrak{P}(\mathrm{id}_A)$, $\mathfrak{P}(\mathrm{id}_B)$, $\mathfrak{P}(\pi)$ and $\mathfrak{P}(\phi)$, we obtain $\mathcal{B}(\mathfrak{P}_f(\pi) \overset{B}{\bowtie} \mathfrak{P}_f(\phi)) = \mathfrak{P}_f(\mathrm{id}_A)$ and $\mathcal{B}(\mathfrak{P}_f(\phi) \overset{A}{\bowtie} \mathfrak{P}_f(\pi)) = \mathfrak{P}_f(\mathrm{id}_B)$. Thus, $A \overset{\mathfrak{P}_f(\pi),\mathfrak{P}_f(\phi)}{\simeq} B$.

Assume there exist two proof-nets θ and ψ such that $A \overset{\theta,\psi}{\simeq} B$. By Theorem 4.18, there exist atomic-axiom cut-free proofs π and ϕ such that $\mathfrak{P}_f(\pi) = \theta$ and $\mathfrak{P}_f(\phi) = \psi$ (they are cut-free by Fact 4.17, so $\mathfrak{P}$ is the singleton containing the image of $\mathfrak{P}_f$). Using Lemma 6.35 twice, one get $\pi \overset{B}{\bowtie} \phi =_\beta \mathrm{id}_A$ and $\phi \overset{A}{\bowtie} \pi =_\beta \mathrm{id}_B$, whence $A \overset{\pi,\phi}{\simeq} B$. □

6.3.2 Remarkable shapes of proof-nets

When studying isomorphisms, and retractions in the next chapter, some special shapes of proof-nets naturally arise.

Definition 6.37 (Bipartite, Half-bipartite)**.** Consider a (cut-free) set of linkings θ on a cut sequent A, B.

We call θ **bipartite** if each of its links is between a leaf of A and a leaf of B – in other words, there is no link between leaves of A, nor between leaves of B.

If θ has no link between leaves of A, we call it **half-bipartite in** A.

Remark 6.38. A set of linkings θ of cut sequent A, B is bipartite if and only if it is half-bipartite in A and in B, hence the name of half-bipartite.

Definition 6.39 (Full, Ax-unique)**.** A set of linkings θ is called **full** if all of the leaves of its cut sequent have (at least) one link on it. Furthermore, if for any leaf there exists a unique link on it (possibly shared among several linkings), then θ is said ***ax*-unique**.

For instance:

- the top-left proof-net of Figure 6.1 (Page 221) is non-bipartite, half-bipartite in its right conclusion, full and non-ax-unique; meanwhile, its top-right proof-net is bipartite and non-full (so non-ax-unique);
- the three proof-nets on Figure 6.2 (Page 222) are bipartite and ax-unique;
- both proof-nets on Figure 6.3 (Page 224) are bipartite, full and non-ax-unique.

Bipartiteness is preserved by composition followed by cut-elimination, while ax-uniqueness is preserved be composition.

Lemma 6.40. *Let θ and ψ be bipartite sets of linkings of respective cut sequents $A, B^\perp$ and B, C. Their composition over B reduces, after cut-elimination, to a bipartite set of linkings.*

Proof. The resulting set of linkings, call it ζ, has for cut sequent A, C. By bipartiteness, any link in $\theta \overset{B}{\bowtie} \psi$ containing a leaf of the cut pair is either of the form (l, m) with l in A and m in $B^\perp$, or (m, n) with m in B and n in C. Therefore, the only links in ζ that were not in θ nor in ψ are those resulting from the replacement of a pair of links (l, m) and $(m^\perp, n)$ with a link (l, n), where l is a leaf of A, m one of B and n one of C – there cannot be a bigger "chain" of leaves as it would need a link with both leaves in the cut pair (recall Definition 4.14 of Turbo Cut-elimination). Hence, the new axiom link (l, n) is between a leaf of A and one of C. □

Lemma 6.41. *If θ and θ' are two ax-unique (resp. full) sets of linkings respectively on $[\Sigma]$ $A^\perp, \Gamma$ and $[\Sigma']$ A, Γ', then $\theta \overset{A}{\bowtie} \theta'$ is ax-unique (resp. full).*

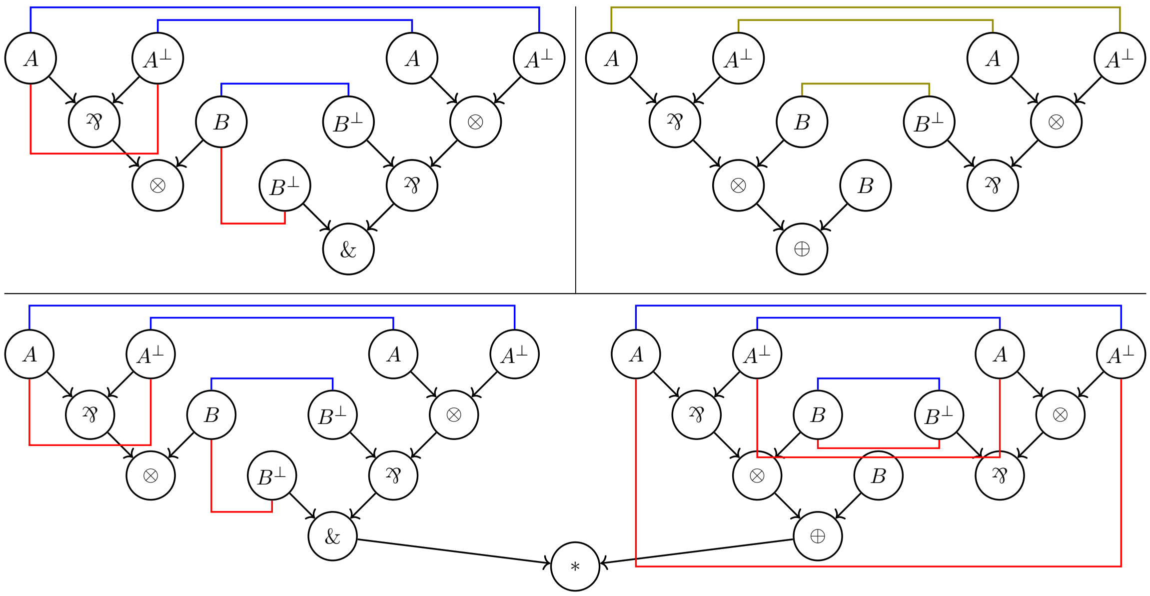


Figure 6.1: Non bipartite proof-net (top-left), non full proof-net (top-right) and one of their compositions yielding the identity proof-net (bottom)

Proof. An axiom link of $\theta \overset{A}{\bowtie} \theta'$ is either an axiom link of θ or one of θ'. Meanwhile, a leaf of $\theta \overset{A}{\bowtie} \theta'$ is either a leaf of θ or one of θ'. Hence the result. $\square$

Nevertheless, applying cut-elimination steps does not preserve fullness nor ax-uniqueness, because eliminating a $\& - \oplus$ cut-elimination step can delete linkings. Furthermore, neither fullness, ax-uniqueness nor bipartiteness is preserved by cut anti-reduction. A counter-example is given on Figure 6.1, with a non bipartite proof-net and a non full one whose composition reduces to a bipartite ax-unique identity proof-net (in fact we will prove that all identity proof-nets are bipartite ax-unique, Corollary 6.45).

As a last remark, when considering sub-systems of MALL_{uf} some of these shapes are given for free.

Lemma 6.42. *A proof-structure on a cut sequent whose formulas are in MLL_{uf} is composed of a single linking and is ax-unique.*

Proof. Such a proof-structure has exactly one additive resolution, which is also its unique $\&$-resolution. Thus, it consists of a unique linking λ by (P1). By (P0), the additive resolution of λ contains all cut pairs, *i.e.* is on the entire cut sequent. Thus, λ is a partition of the leaves of the whole cut sequent in pairs, so each leaf has a link on it, which is unique as there is no other linking than λ. $\square$

Lemma 6.43. *Let θ be a set of linkings on a cut sequent A, B whose formulas are in ALL_{uf}. Then θ is bipartite, and each of its linkings is composed of a single link.*

Proof. Any additive resolution contains exactly one leaf of A and one of B, because connectives of these formulas are all additive ones. $\square$

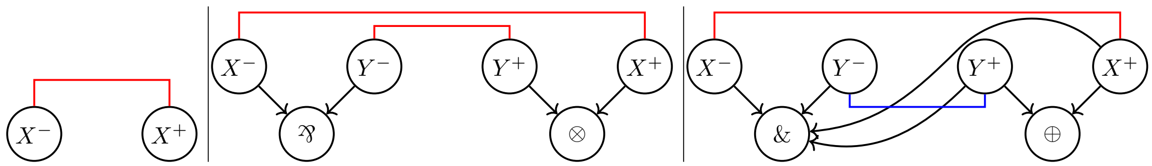


Figure 6.2: Identity proof-nets (from left to right: atom, ⅋\\$\otimes$ and &\\$\oplus$)

6.3.3 Properties of identity proof-nets

Using an induction on the formula A, we prove some results on id_A, the identity proof-net of A; in particular it is bipartite ax-unique. See Figure 6.2 for a graphical intuition.

Lemma 6.44. *The axiom links of an identity proof-net are exactly the $(l, l^\perp)$, for any leaf l.*

Proof. By induction on the formula – see Figure 6.2. $\square$

Corollary 6.45. *An identity proof-net is bipartite ax-unique.*

Proof. This follows from Lemma 6.44. $\square$

Lemma 6.46. *Let λ be a linking of an identity proof-net and v an additive vertex in its additive resolution. Then $v^\perp$ is also inside with, as premise kept, the dual premise of the one kept for v.*

Proof. Assume *w.l.o.g.* that the left premise of v is kept in λ. There is a left-ancestor l of v in the additive resolution of λ, hence with a link $a \in \lambda$ on it. By Lemma 6.44, $a = (l, l^\perp)$. As $l^\perp$ is a right-ancestor of $v^\perp$, the conclusion follows. $\square$

The next result allows to go from exactly one linking on any &-resolution of $A^\perp, A$ (by (P1)) to exactly one linking on any additive resolution of A.

Lemma 6.47. *Given an additive resolution of A, exactly one linking of id_A is on it.*

Proof. Consider such an additive resolution R. There is an associated &-resolution R' of $A^\perp, A$ by taking the choices of premise of R on A and, for a &-vertex w of $A^\perp$, taking the dual premise chosen in R for the $\oplus$-vertex $w^\perp$. By Lemma 6.46, a linking λ is on R if and only if it is on R'. Meanwhile, by (P1) there is a unique linking λ on R'; thus the same holds on R. $\square$

6.3.4 Bipartite full proof-nets

We can now prove that proof-nets of isomorphisms are bipartite full. The difficulty is that, as written in Section 6.3.2, neither fullness, ax-uniqueness nor bipartiteness is preserved by cut anti-reduction. However, if *both* compositions yield identity proof-nets, we get bipartiteness and fullness.

Lemma 6.48. *Let θ and θ' be cut-free proof-nets of respective cut sequents $A^\perp, B$ and $B^\perp, A$, such that $\theta' \overset{A}{\bowtie} \theta \xrightarrow{\beta^*} \mathrm{id}_B$. For any linking $\lambda \in \theta$, there exists $\lambda' \in \theta'$ such that $\lambda \cup \lambda'$ matches in $\theta \overset{B}{\bowtie} \theta'$, the composition over B of θ and θ'.*

Proof. Let us consider a linking $\lambda \in \theta$, and call $\mathcal{C}$ the choices of premises on additive connectives of B that λ makes. We search some $\lambda' \in \theta'$ making the dual choices of premises on additive connectives of $B^\perp$ as compared to $\mathcal{C}$. Consider the composition of θ and θ' over A. It reduces to the identity proof-net of B by hypothesis. By Lemma 6.47, there exists a unique linking ν in the identity proof-net of B corresponding to $\mathcal{C}$. Furthermore, this linking ν of the identity proof-net is derived from some $\mu \cup \mu'$ for μ a linking of θ and μ' one of θ', with $\mu \cup \mu'$ matching for a cut over A: a linking in the identity proof-net is a linking of the form $\mu \cup \mu'$ where axiom links (l, m_1), $(m_1^\perp, m_2)$, $\dots$, $(m_n^\perp, l^\perp)$ in μ and μ' are replaced with $(l, l^\perp)$, with l a leaf of B and the m_i and $m_i^\perp$ of $A^\perp$ and A (because an identity proof-net has only links of the form $(l, l^\perp)$ by Lemma 6.44). Therefore, μ makes the choices $\mathcal{C}$ on B and $\mu \cup \mu'$ matches for the composition of θ and θ' over both A and B. But λ makes the same choices $\mathcal{C}$ on B as μ: $\lambda \cup \mu'$ also matches for a cut over B. □

Remark 6.49. Lemma 6.48 is the analogue of Lemma 6.22 in proof-nets. Indeed, Lemma 6.22 states that given two sderivations π and π' composing to the identity on B, and s a slice of π, there exists a slice s' of π' such that $s \overset{B}{\bowtie} s'$ does not fail when reducing cuts – *i.e.* the two slices make the dual choises for additive connectives in B. Seeing a linking as a slice, this corresponds to having two matching linkings.

Corollary 6.50. *Assuming $A \overset{\theta,\theta'}{\simeq} B$, θ and θ' are bipartite.*

Proof. We proceed by contradiction: *w.l.o.g.* there is a link a in some linking $\lambda \in \theta$ which is between leaves of $A^\perp$. Remember that in the notation $A \overset{\theta,\theta'}{\simeq} B$, both proof-nets θ and θ' are assumed cut-free. Thence, one can apply Lemma 6.48: there exists $\lambda' \in \theta'$ such that $\lambda \cup \lambda'$ matches for a cut over B. For a does not involve leaves of B, it stays in the linking of the normal form resulting from $\lambda \cup \lambda'$ (after eliminating all cuts in the composition). But this normal form is by hypothesis the identity proof-net of A, which is bipartite by Corollary 6.45. Thus, there cannot be an axiom link between leaves of $A^\perp$ inside: contradiction. □

Lemma 6.51. *Assume θ and θ' are cut-free proof-nets respectively on $A^\perp, B$ and $B^\perp, A$, such that $\theta \overset{B}{\bowtie} \theta' \xrightarrow{\beta^*} \mathrm{id}_A$. Then any leaf of $A^\perp$ (resp. A) has at least one axiom link on it in θ (resp. θ').*

Proof. Towards a contradiction, assume *w.l.o.g.* a leaf l of $A^\perp$ has no link on it in θ. Then, $\theta \overset{B}{\bowtie} \theta'$ has no link on l either. Moreover, reducing cuts cannot create links using l, for the only created links (k, n) are those coming from the merging of links (k, m) and $(m^\perp, n)$. Whence, in the normal form of $\theta \overset{B}{\bowtie} \theta'$ there is no link on l. However, the identity proof-net of A is *ax*-unique by Corollary 6.45, thence full: contradiction. □

Remark 6.52. Lemma 6.51 holds not only for isomorphisms but more generally for retractions, which are formulas A and B such that there exist proofs of $\vdash A^\perp, B$ and $\vdash B^\perp, A$ whose composition by cut over B (but not necessarily over A) is equal to the axiom on $\vdash A^\perp, A$ up to axiom-expansion and cut-elimination. An example of retraction is given by the proof-nets on Figure 6.1, yielding a retraction $(A \parr A^\perp) \otimes B \lhd ((A \parr A^\perp) \otimes B) \oplus B$, which is not an isomorphism. See the next chapter for more details on retractions, in the framework of $\mathsf{MLL}^{0,2}$.

Theorem 6.53. *Assuming $A \overset{\theta,\theta'}{\simeq} B$, θ and θ' are bipartite full.*

Proof. This is thanks to Corollary 6.50 and Lemma 6.51. □

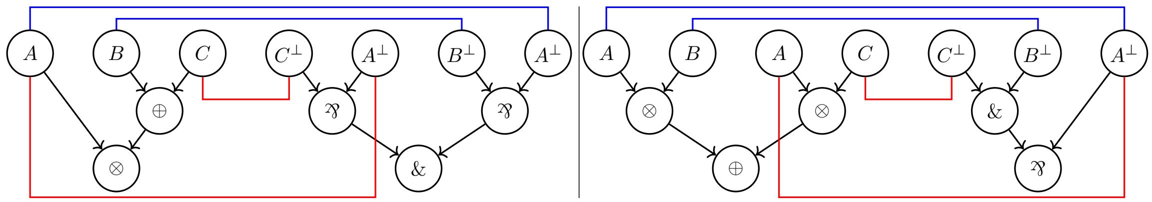


Figure 6.3: Proof-nets for $A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$

6.3.5 *Ax*-uniqueness

In general, isomorphisms do not yield *ax*-unique proof-nets. A counter-example is the distributivity $A \otimes (B \oplus C) \simeq (A \otimes B) \oplus (A \otimes C)$, see Figure 6.3. Nonetheless, distributivity equations are the only ones in $\mathcal{L}^\dagger$ not giving *ax*-unique proof-nets, and we should not have to consider them for we can consider only distributed formulas (Proposition 6.13). Therefore, on distributed formulas, distributivity isomorphisms can be ignored, and isomorphisms between distributed formulas can be proved bipartite *ax*-unique.

6.3.5.1 Preliminary results

We mostly use the correctness criterion through the fact we can sequentialize, *i.e.* recover a proof tree from a proof-net by Theorem 4.18. However, in order to prove *ax*-uniqueness, we make a direct use of the correctness criterion to deduce geometric properties of proof-nets. This is done through Corollary 6.54, which is a corollary from Lemma 4.28 whose proof uses (P3). This part of the proof benefits from the specificity of this syntax.

We recall that given Λ a set of linkings and w a &, Λ^w is the set of all linkings in Λ whose additive resolution does not contain the right argument of w. We use Λ^w to deduce from the toggling condition (P3) a simpler and easier to maniupulate (but weaker) condition that a proof-net respects. Then, given proof-nets of an isomorphism that are not *ax*-unique, we will show they do not respect this simpler condition, hence we will reach a contradiction.

Corollary 6.54. *Let ω be a jump-free switching cycle in a proof-net θ. There exists a subset of linkings $\Lambda \subseteq \theta$ and a &-vertex w toggled by Λ such that $\omega \subseteq \mathcal{G}_\Lambda$, $w \notin \omega$, $\omega \not\subseteq \mathcal{G}_{\Lambda^w}$ and there exists an axiom link $a \in \omega$ depending on w in Λ.*

Proof. Take Λ a minimal saturated subset of θ with $\mathcal{G}_\Lambda$ containing ω. We conclude through Lemma 4.28. □

For v and u vertices in a tree, their **first common descendant** is the vertex of the tree which is a descendant of both v and u and which has no ancestor respecting this property. Equivalently, looking at a tree as a partial order of minimal element the root, the first common descendant is the infimum.

Lemma 6.55. *Let θ be a proof-net and A a formula (or a cut pair) of its cut sequent with a jump arc $l \xrightarrow{j} w$ between $l, w \in T(A)$. If w is not a descendant of l, then their first common descendant v is a ⅋.*

Proof. As there is a jump $l \xrightarrow{j} w$, there exist linkings $\lambda, \lambda' \in \theta$ such that w is the only & toggled by $\{\lambda; \lambda'\}$, and a link $a \in \lambda \backslash \lambda'$ using the leaf l. In particular, the jump $l \xrightarrow{j} w$ is in $\mathcal{G}_{\{\lambda;\lambda'\}}$. For l and

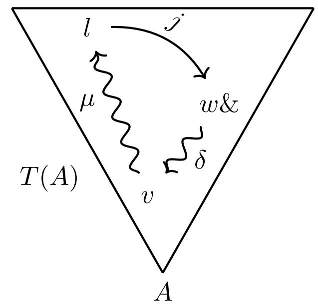


Figure 6.4: Illustration of the proof of Lemma 6.55

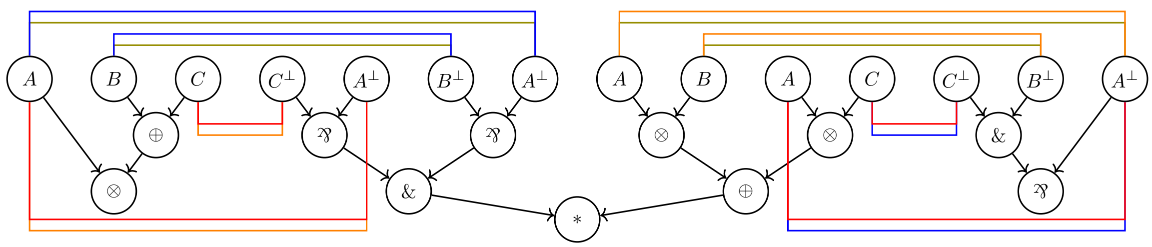


Figure 6.5: Proof-nets from Figure 6.3 composed by cut on $(A \otimes B) \oplus (A \otimes C)$

w are both in the additive resolution of λ, both premises of v are also in this additive resolution, thus v cannot be an additive connective, so not a & nor a $\oplus$-vertex.

Assume by contradiction that v is a $\otimes$. Call δ the path in $T(A)$ from w to v, and μ the one from v to l (see Figure 6.4). Then, $(l \xrightarrow{j} w) \cdot \delta \cdot \mu$ is a switching cycle in $\mathcal{G}_{\{\lambda;\lambda'\}}$. According to (P3), there exists a & toggled by $\{\lambda; \lambda'\}$ not in any switching cycle of $\mathcal{G}_{\{\lambda;\lambda'\}}$. A contradiction, for w is the only & toggled by $\{\lambda; \lambda'\}$. Whence, v can only be a ⅋. $\square$

6.3.5.2 Isomorphisms of distributed formulas

Now, let us prove that isomorphisms of distributed formulas are bipartite ax-unique. We will consider proof-nets corresponding to an isomorphism that we cut and where we eliminate all cuts not involving atoms. To give some intuition, let us consider the non-ax-unique proof-nets of Figure 6.3 (on Page 224). Composing them together by cut on $(A\otimes B)\oplus(A\otimes C)$ gives the proof-net illustrated on Figure 6.5. Reducing all cuts not involving atoms yields the proof-net on Figure 6.6, that we call an *almost reduced composition*. We stop there because of the switching cycle produced by the two links on A (dashed in blue on Figure 6.6), less visible in the non-reduced composition of Figure 6.5. However, reducing all cuts gives the identity proof-net, which has no switching cycle: during these reductions, both links on A are merged. By using almost reduced composition, we are going to prove that links preventing ax-uniqueness yield switching cycles, and moreover that these cycles are due to non-distributed formulas only.

Definition 6.56 (Almost reduced composition)**.** Take θ and θ' cut-free proof-nets respectively on Γ, B and $B^\perp, \Delta$. The **almost reduced composition** over B of θ and θ' is the proof-net resulting

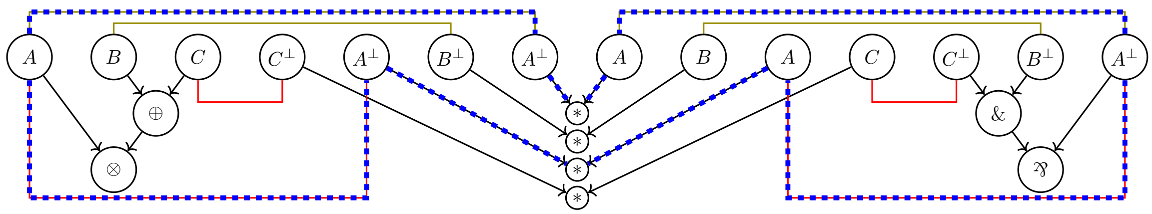


Figure 6.6: An almost reduced composition of the proof-nets on Figure 6.3

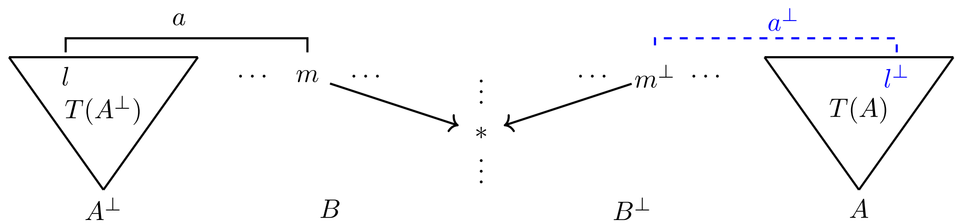


Figure 6.7: Illustration of Lemma 6.57

from the composition over B of θ and θ' where we repeatedly reduce all cuts not involving atoms (*i.e.* not applying step (a) of Definition 4.9).

The previous definition uses convergence of non-atomic cut-elimination (Remark 4.13).

Let us fix A and B two MALL_{uf} (not necessarily distributed yet) formulas as well as θ and θ' such that $A \overset{\theta,\theta'}{\simeq} B$. By Theorem 6.53, θ and θ' are bipartite full. We denote by ψ the almost reduced composition over B of θ and θ'. Here, we can extend our duality on vertices and premises (defined in Section 4.1) to links.

Lemma 6.57. *An axiom link $a = (l, m)$ belongs to some linking $\lambda \in \psi$ if and only if, up to swapping l and m, l is a leaf of $A^\perp$ (resp. A), m is in the leaves of B (resp. $B^\perp$) and there is an axiom link $(l^\perp, m^\perp)$ in the same linking λ, that we will denote $a^\perp = (l^\perp, m^\perp)$ (see Figure 6.7).*

Proof. By symmetry, we only need to prove the "if" statement. Linkings of ψ are disjoint union of linkings in θ and θ'. By symmetry, assume $(l, m) \in \lambda \in \psi$ comes from a linking in θ. As θ is bipartite, one of the leaves, say l, is in $A^\perp$ and the other, m, is a leaf of B. For the cut $m * m^\perp$ belongs to the additive resolution of λ (as m is inside), $m^\perp$ is a leaf in this resolution. Thus, there is a link $(m^\perp, l') \in \lambda$ for some leaf l', which necessarily belongs to A by bipartiteness of θ'. It remains to prove $l' = l^\perp$. If we were to eliminate all cuts in ψ, we would get the identity proof-net on A by hypothesis. But eliminating the cut $m * m^\perp$ yields a link (l, l'), which is not modified by the elimination of the other atomic cuts. By Lemma 6.44, $l' = l^\perp$ follows. □

Lemma 6.58. *Let λ be a linking of ψ, and v an additive vertex in its additive resolution. Then $v^\perp$ is also inside, with as premise kept the dual premise of the one kept for v.*

Proof. Assume *w.l.o.g.* that the left premise of v is kept in λ. There is a left-ancestor l of v in the additive resolution of λ, hence with a link $a \in \lambda$ on it. By Lemma 6.57, we have $a^\perp \in \lambda$, using $l^\perp$. As $l^\perp$ is a right-ancestor of $v^\perp$, the conclusion follows. □

Lemma 6.59. *Let w and p be respectively a &-vertex and a $\oplus$-vertex in ψ, with w an ancestor of p. Then for any axiom link a depending on w in ψ, a also depends on $p^\perp$ in ψ.*

Proof. There exist linkings $\lambda, \lambda' \in \psi$ such that w is the only & toggled by $\{\lambda; \lambda'\}$ and $a \in \lambda \backslash \lambda'$. We consider a linking $\lambda_{p^\perp}$ defined by taking an arbitrary &-resolution of λ where we choose the other premise for $p^\perp$ (and arbitrary premises for &-vertices introduced this way, meaning for &-ancestors of $p^\perp$): by (P1), there exists a unique linking on it. By Lemma 6.58, the additive resolutions of λ and $\lambda_{p^\perp}$ (resp. λ and λ') differ, on A and $A^\perp$, exactly on ancestors of p and $p^\perp$ (resp. w and $w^\perp$). Thus, the additive resolutions of λ' and $\lambda_{p^\perp}$ also differ, on A and $A^\perp$, exactly on ancestors of p and $p^\perp$, for w is an ancestor of p. In particular, $\{\lambda; \lambda_{p^\perp}\}$, as well as $\{\lambda'; \lambda_{p^\perp}\}$, toggles only $p^\perp$. If $a \in \lambda_{p^\perp}$, then a depends on $p^\perp$ in $\{\lambda'; \lambda_{p^\perp}\}$. Otherwise, a depends on $p^\perp$ in $\{\lambda; \lambda_{p^\perp}\}$. $\square$

The key result to use distributivity is that a positive vertex "between" a leaf l and a &-vertex w in the same tree prevents them from interacting, *i.e.* there is no jump $l \xrightarrow{j} w$.

Lemma 6.60. *Let $l \xrightarrow{j} w$ be a jump arc in ψ, with l not an ancestor of w and $l, w \in T(A^\perp)$ (resp. $T(A)$). Denoting by n the first common descendant of l and w, there is no positive vertex in the path between n and w in $T(A^\perp)$ (resp. $T(A)$).*

Proof. As there is a jump $l \xrightarrow{j} w$, there exist linkings $\lambda, \lambda' \in \psi$ and a leaf m of B such that w is the only & toggled by $\{\lambda; \lambda'\}$ and $a = (l, m) \in \lambda \backslash \lambda'$. By Lemma 6.55, n is a ⅋-vertex. Let p be a vertex on the path between n and w in $T(A^\perp)$. We prove by contradiction that p can neither be a $\oplus$ nor a $\otimes$-vertex.

Suppose p is a $\oplus$-vertex. By Lemma 6.59, a depends on $p^\perp$, and so does $a^\perp$ through Lemma 6.57: there is a jump arc $l^\perp \xrightarrow{j} p^\perp$. Applying Lemma 6.55, the first common descendant of $l^\perp$ and $p^\perp$, which is $n^\perp$, is a ⅋-vertex: a contradiction as it is a $\otimes$-vertex.

Assume now p to be a $\otimes$-vertex. Our reasoning is illustrated on Figure 6.8. For p is a $\otimes$, there is a leaf o which is an ancestor of p in the additive resolution of λ, from a different premise of p than w; it is used by a link $b = (o, q) \in \lambda$. Remark $q \neq m$, for a and b are two distinct links in the same linking λ. Then the switching cycle $l \xrightarrow{j} w \to p \leftarrow o \overset{b}{—} q \to * \leftarrow q^\perp \overset{b^\perp}{—} o^\perp \to p^\perp \to n^\perp \leftarrow l^\perp \overset{a^\perp}{—} m^\perp \to * \leftarrow m \overset{a}{—} l$ (dashed in blue on Figure 6.8) belongs to $\mathcal{G}_{\{\lambda;\lambda'\}}$, with unlabeled arrows being paths in the syntactic forest. Contradiction: w, the only & toggled by $\{\lambda; \lambda'\}$, cannot be in any switching cycle of $\mathcal{G}_{\{\lambda;\lambda'\}}$ by (P3). $\square$

Theorem 6.61. *Assuming $A \overset{\theta,\theta'}{\simeq} B$ with A and B distributed, θ and θ' are bipartite ax-unique.*

Proof. We already know that θ and θ' are bipartite full thanks to Theorem 6.53. We reason by contradiction and assume *w.l.o.g.* that θ is not *ax*-unique: there exist a leaf l of $A^\perp$ and two distinct leaves l_0 and l_1 of B with links $a = (l, l_0)$ and $b = (l, l_1)$ in θ. We consider ψ the almost reduced composition of θ and θ' over B, depicted on Figure 6.9. By Lemma 6.48, a and b are also links in ψ: the linkings they belong to in θ have matching linkings in θ', and we did not eliminate atomic cuts. Using Lemma 6.57, we have in $\mathcal{G}_\psi$ a switching cycle $\omega = l \overset{a}{—} l_0 \to * \leftarrow l_0^\perp \overset{a^\perp}{—} l^\perp \overset{b^\perp}{—} l_1^\perp \to * \leftarrow l_1 \overset{b}{—} l$.

Let Λ be a set of linkings and w an associated &-vertex given by Corollary 6.54 applied to ω. The vertex w belongs to either $T(A)$ or $T(A^\perp)$: up to swapping them, w is in $T(A^\perp)$. By Corollary 6.54, a, $a^\perp$, b or $b^\perp$ depends on w. So a or b depends on w by Lemma 6.57; *w.l.o.g.* a depends on w. Remark l is not an ancestor of w: if it were, by symmetry assume it is a left-ancestor. Whence a

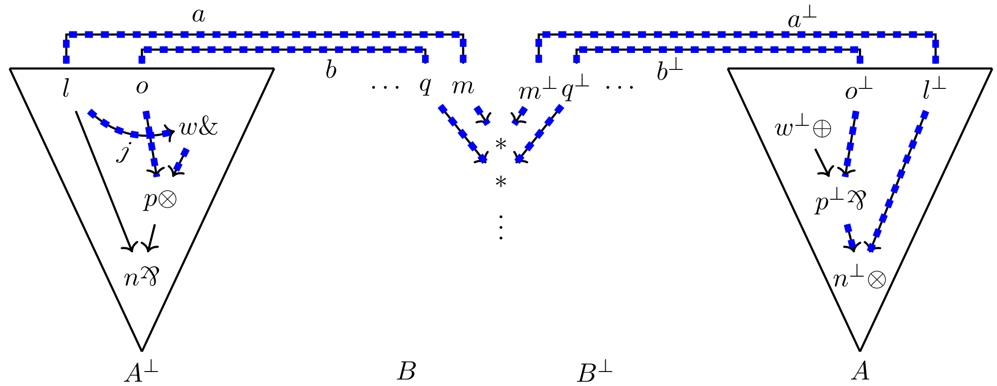


Figure 6.8: Switching cycle containing w if p is a $\otimes$-vertex in the proof of Lemma 6.60

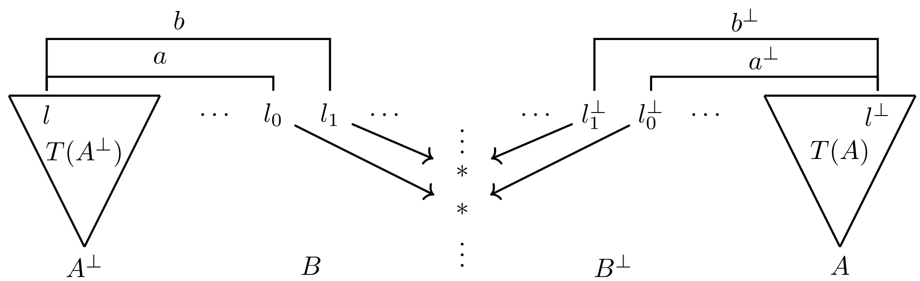


Figure 6.9: Almost reduced composition ψ of θ and θ' by cut over B in the proof of Theorem 6.61

and b belong to Λ^w, so $a^\perp$ and $b^\perp$ too (Lemma 6.57); thus $\omega \subseteq \mathcal{G}_{\Lambda^w}$, contradicting Corollary 6.54. Hence, l is not an ancestor of w, and we can apply Lemma 6.55: the first common descendant n of l and w in $T(A^\perp)$ is a ⅋. Using Lemma 6.60, there is no $\otimes\backslash\oplus$-vertex on the path between the ⅋-vertex n and its ancestor the &-vertex w in $T(A^\perp)$. But then, considering the first &-vertex in this path, there is a sub-formula of the shape $- ⅋ (- \,\&\, -)$ or $(- \,\&\, -) ⅋ -$ in the distributed $A^\perp$, a contradiction. $\square$

6.3.6 Non-ambiguous formulas

Once our study is restricted to bipartite *ax*-unique proof-nets, we can also restrict formulas further.

Definition 6.62 (Non-ambiguous formula)**.** A formula A is said **non-ambiguous** if each unsigned atom in A occurs at most once (counting both its positive and negative occurrences). Otherwise, A is called **ambiguous**.

Example 6.63. The formula $X^+ \,\&\, Y^+$ is non-ambiguous, whereas $X^+ \,\&\, X^-$ and $(A \otimes B) \oplus (A \otimes C)$ are ambiguous.

Lemma 6.64. *If A is non-ambiguous, then $A^\perp$ is too.*

Proof. Because the positive (resp. negative) atoms of $A^\perp$ are exactly the negations of the negative (resp. positive) atoms of A. □

Remark 6.65. Our definition of non-ambiguous is not exactly the same as the one from [BD99], it is stronger (there are less formulas that are non-ambiguous). In [BD99], a formula A is non-ambiguous if each atom in A occurs at most once positive and once negative. This makes for instance $X^+ \& X^-$ non-ambiguous, whereas by our definition it is ambiguous. We prefer to make formulas like $X^+ \parr X^-$ ambiguous, because when renaming we can go to this stronger setting, and it allows stronger results – see for instance Remark 6.85 or Lemma 6.66, the latter being false with the definition from [BD99], with as a counter-example the sole proof-structure on $(X^+ \parr X^-) \otimes Y^+, Y^-$, which is a connected proof-net.

Lemma 6.66. *Let θ be a set of linkings on A, B. If A is non-ambiguous, then θ is half-bipartite in A.*

Proof. There simply cannot be any axiom link between leaves of A, because two such leaves are not on the same unsigned atom. □

The reduction to non-ambiguous formulas requires to restrict to distributed formulas first: in $(A \otimes B) \oplus (A \otimes C) \simeq A \otimes (B \oplus C)$ we need the two occurrences of A to factorize. The goal of this section is to prove that we can consider only non-ambiguous formulas (Corollary 6.83) and that isomorphisms for these formulas correspond simply to the *existence* of proof-nets (Theorem 6.91). These two results are an adaptation of the work on MLL by Balat & Di Cosmo [BD99, Section 3]. As many results here will also be of use in the next chapter when studying retractions, we give them with more general hypotheses that needed right now (with half-bipartiteness instead of bipartiteness for example) so that they apply in both cases without doing twice the work.

6.3.6.1 Renaming to non-ambiguous formulas

Here is given the main result used to reduce the study of isomorphisms or retractions to the case where we have a non-ambiguity hypothesis, namely Theorem 6.81.

We call **atomic-substitution** the operation $[Y_1/X_1, \dots, Y_n/X_n]$ of replacement of the atoms X_i of a formula by the signed atom Y_i – *i.e.* with X_i^+ replaced by Y_i^+ and X_i^- replaced by Y_i^-. In other words, an atomic-substitution is a substitution where the image of each atom is a signed atom.

We will consider atomic-substitutions extended to proof-nets, *i.e.* if σ is an atomic-substitution and θ a proof-net, $\sigma(\theta)$ will be the proof-net obtained from θ by relabeling all leaves X_i appearing in it by $\sigma(X_i)$. Remark that with a general substitution, there would be some work to do here as the definition of proof-nets we took has only atomic-axioms (but a more complex notion of proof-nets with non-atomic-axioms can be defined). We also use a more general notion, **renaming**, that may replace *different occurrences* of the *same* atom by different signed atoms in a proof-net, *i.e.* substitute on leaves (or occurrences) instead of atoms.

Definition 6.67 (Renaming)**.** An application α from the set of *leaves* of a proof-net θ to a set of signed atoms is a **renaming** if $\alpha(\theta)$, the graph obtained by substitution of each leaf l of θ by $\alpha(l)$, is a proof-net.

Note that if the cut sequent of θ contains ambiguous formulas, then two different occurrences of the same atom can be renamed differently, unlike what happens in the case of substitutions.

Lemma 6.68. *Take θ a proof-net on the cut sequent $[\Sigma]\ \Gamma$. An application α from the set of leaves of θ to a set of signed atoms is a renaming if and only if for all links $a = (l, m)$ of θ, $\alpha(l) = \alpha(m)^\perp$ and for all cut pairs $A^\perp * A$ of Σ, $\alpha(A^\perp) = \alpha(A)^\perp$.*

In particular, if θ is cut-free, then α is a renaming if and only if for all links $a = (l, m)$ of θ, $\alpha(m) = \alpha(l)^\perp$.

Proof. The direct way follows from the definition of a linking and that a proof-net is a set of linkings. For the converse way, remark there is a bijection between additive resolutions (resp. &-resolutions, cut pairs) of θ and $\alpha(\theta)$. The hypothesis on cut pairs implies that $\alpha(\Sigma)$ is a set of cut pairs, hence that $[\alpha(\Sigma)]\ \alpha(\Gamma)$ is a cut sequent. The hypothesis on links ensures each $\alpha(\lambda)$ is a linking, for $\lambda \in \theta$. There are four items to check in the definition of a proof-net, (P0) to (P3). As they are independent from the labels of the leaves, they hold in $\alpha(\theta)$ because they do in θ. □

Lemma 6.69. *Take θ a proof-net on the cut sequent A, B. Assume it is half-bipartite in A and that there is at most one link on each leaf. The definition of α only on leaves in A is sufficient to uniquely define a renaming α on θ – with no conditions on the definition of α on A.*

Proof. Assume α is defined on leaves of A only. Extend it to a complete function on leaves of θ as follows. Given a leaf l of B, if it is linked to a leaf m of A, then it is only linked to this leaf m by assumption; in this case, we set $\alpha(l) = \alpha(m)^\perp$. Otherwise, if l is linked to no leaf of A, we define $\alpha(l) = l$.

The resulting $\alpha(\theta)$ is indeed a proof-net: the only thing to check is that links are between dual leaves (Lemma 6.68). Take a link $a = (l, m)$, with l an occurrence of X and m one of $X^\perp$, X being a signed atom. By assumption neither is linked to other leaves, so we only have to check that their labels after renaming are still dual. We have three cases: l and m are both in A, both in B, or one is in A and the other in B.

- As two leaves of A are not linked together by half-bipartiteness, the first case cannot happen.
- If two leaves of B are linked together, by hypothesis neither is linked to other leaves, and in particular to a leaf in A: we have $\alpha(l) = X$ and $\alpha(m) = X^\perp$.
- By symmetry, say l is in A and m in B. Then, by definition of α, $\alpha(m) = \alpha(l)^\perp$ as wished. □

This last result allows us to rename some particular proof-nets so as to have one or all conclusions non-ambiguous.

Lemma 6.70. *Let A and B be MALL_{uf} formulas and θ a proof-net on the cut sequent A, B which is ax-*unique *and half-bipartite in A. Then there exists a renaming α of θ such that $\alpha(A)$ is non-ambiguous.*

Proof. It is sufficient to define α only on leaves of A by Lemma 6.69. One can define α such that $\alpha(A)$ is non-ambiguous, by sending leaves whose unsigned atoms are the same to different fresh atoms (using that the set of atoms $\mathcal{X}$ is infinite). Indeed, the resulting formula has distinct atoms, *i.e.* no atom of $\alpha(A)$ occurs twice in $\alpha(A)$, even one positively and one negatively: $\alpha(A)$ is non-ambiguous. □

Lemma 6.71. *Consider a bipartite ax-unique set of linkings θ on the cut sequent A, B. Then A is non-ambiguous if and only if B is.*

Proof. By symmetry, assume A is non-ambiguous. As θ is ax-unique, each leaf l of B is associated to a unique leaf $f(l)$, by means of an axiom link. By bipartiteness, $f(l)$ is in A, thus by non-ambiguity it is the only leaf of A having its (unsigned) atom. As axiom links are between leaves with dual atoms, it follows that l is the only leaf of B having this (unsigned) atom. Thence, $\alpha(B)$ is also non-ambiguous. $\square$

Renamings on identities are particularly well-behaved.

Lemma 6.72. *Consider a renaming α of id_A. Then $\alpha(A^\perp) = \alpha(A)^\perp$, and $\alpha(\mathrm{id}_A) = \mathrm{id}_{\alpha(A)}$.*

Proof. By Lemma 6.44, the links of id_A are exactly the $a = (l^\perp, l)$ for some leaf l of A. As α is a renaming, it follows $\alpha(l^\perp) = \alpha(l)^\perp$ for every leaf l of A. Thence, $\alpha(A^\perp) = \alpha(A)^\perp$, as a renaming acts only on labels of leaves.

Moreover, $\alpha(\mathrm{id}_A)$ is a proof-net on $\alpha(A)^\perp, \alpha(A)$ whose links are exactly the $(l^\perp, l)$ for l a leaf of $\alpha(A)$, *i.e.* with the same links as $\mathrm{id}_{\alpha(A)}$ by Lemma 6.44. Plus, it also has the same additive resolutions for each linking as those of $\mathrm{id}_{\alpha(A)}$, which are the "same" as the ones of id_A: the additive resolution of a linking $\alpha(\lambda)$ of $\alpha(\mathrm{id}_A)$ is equal to the additive resolution of a linking of $\mathrm{id}_{\alpha(A)}$, which is also the renaming of the additive resolution of the corresponding linking λ of id_A. Thus, $\alpha(\mathrm{id}_A) = \mathrm{id}_{\alpha(A)}$. $\square$

Lemma 6.73. *Consider two renamings α and α' respectively of proof-nets θ and θ', of cut sequents $[\Sigma]\ A^\perp, \Gamma$ and $[\Sigma']\ A, \Gamma'$. Assume that $\alpha(A^\perp) = \alpha'(A)^\perp$, so that $\alpha(\theta)$ and $\alpha(\theta')$ can still be composed after renaming. Then their disjoint union $\alpha \sqcup \alpha'$ is a renaming of $\theta \overset{A}{\bowtie} \theta'$, with $\alpha \sqcup \alpha'(\theta \overset{A}{\bowtie} \theta') = \alpha(\theta) \overset{\alpha'(A)}{\bowtie} \alpha'(\theta')$.*

Proof. We use Lemma 6.68: it suffices to prove its two hypotheses. A link (l, m) of $\theta \overset{A}{\bowtie} \theta'$ is either a link of θ or one of θ'. Hence, it suffices to apply the direct way of Lemma 6.68 to obtain either $\alpha(l) = \alpha(m)^\perp$ or $\alpha'(l) = \alpha'(m)^\perp$. Thus, $\alpha \sqcup \alpha'(l) = \alpha \sqcup \alpha'(m)^\perp$.

For the second hypothesis, take a cut pair of $A^\perp * A, \Sigma, \Sigma'$. If this cut pair is $A^\perp * A$, then we have by assumption $\alpha(A^\perp) = \alpha'(A)^\perp$, whence $\alpha \sqcup \alpha'(A^\perp) = \alpha \sqcup \alpha'(A)^\perp$. Otherwise, it is some cut pair $B^\perp * B$ either in Σ or Σ'. It then suffices to apply the direct way of Lemma 6.68 to obtain either $\alpha(B^\perp) = \alpha(B)^\perp$ or $\alpha'(B^\perp) = \alpha'(B)^\perp$. Thus, $\alpha \sqcup \alpha'(B^\perp) = \alpha \sqcup \alpha'(B)^\perp$. $\square$

Lemma 6.74. *Let θ and θ' be proof-nets respectively on $A^\perp, B$ and $B^\perp, A$ such that $\theta \overset{B}{\bowtie} \theta'$ reduces by cut-elimination to id_A. Consider two renamings α and α' respectively of θ and θ', such that $\alpha(B^\perp) = \alpha'(B)^\perp$. Then $\alpha(\theta) \overset{\alpha(B)}{\bowtie} \alpha'(\theta')$ reduces by cut-elimination to $\mathrm{id}_{\alpha'(A)}$. In particular, $\alpha(A^\perp) = \alpha'(A)^\perp$.*

Proof. By hypothesis, the composition of $\alpha(\theta)$ and $\alpha'(\theta')$ by cut over $\alpha(B)$ is well-defined, hence is a proof-net (Lemma 4.7). Using Lemma 6.73, $\alpha \sqcup \alpha'$ is a renaming of $\theta \overset{B}{\bowtie} \theta'$, and thus induces after cut-elimination a renaming of id_A – for cut-elimination does not depend on labels. Thanks to Lemma 6.72, $\alpha \sqcup \alpha'(A^\perp) = \alpha \sqcup \alpha'(A)^\perp$, so $\alpha(A^\perp) = \alpha'(A)^\perp$, and $\alpha \sqcup \alpha'(\mathrm{id}_A) = \mathrm{id}_{\alpha \sqcup \alpha'(A)} = \mathrm{id}_{\alpha'(A)}$. $\square$

The following concept makes sense mainly for MLL proof-nets, but we will need it for isomorphisms in MALL. Hence, we state it in a general way.

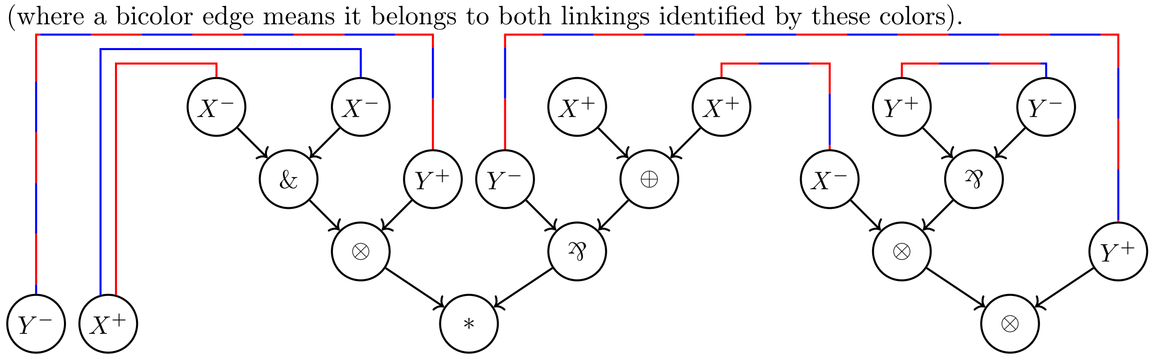
Definition 6.75 (Class of a leaf)**.** Take a set of linkings θ on $[A^\perp * A, \Sigma]\ \Gamma$. Consider the total simple loop-free undirected graph with vertices the leaves of θ and an edge (l, m) if there is a link (l, m) in θ or if l belongs to A, m to $A^\perp$ and $m = l^\perp$. We call this graph the **GOI projection graph on $A^\perp * A$ of θ.**

We define the **class** for $A^\perp * A$ of a leaf l of θ as the vertices connected to l in the previous graph. This defines an equivalence relation on the leaves of θ.

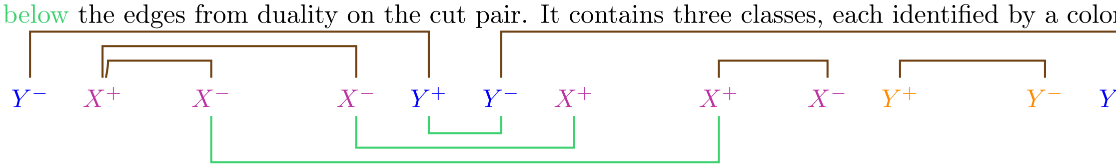
Example 6.76. Consider the following proof-net, on the cut sequent

$$[((X^- \mathbin{\&} X^-) \otimes Y^+) * (Y^- \parr (X^+ \oplus X^+))]\ Y^-, X^+, (X^- \otimes (Y^+ \parr Y^-)) \otimes Y^+$$

(where a bicolor edge means it belongs to both linkings identified by these colors).



Its GOI projection graph (on its unique cut pair) is the following, with leaves in the same order from left to right as in the proof-net above. We put above the edges coming from axiom links, and below the edges from duality on the cut pair. It contains three classes, each identified by a color.



Lemma 6.77. *Let θ be a set of linkings on $[A^\perp * A]\ \Gamma$, and consider a class for $A^\perp * A$ of θ. There exists an unsigned atom X such that any leaf of this class is labeled either X^+ or X^-.*

Proof. Take any leaf l in this class, its label is some X^+ or some X^-. Remark that any pair of leaves linked in the GOI projection graph has dual labels. As a class is a connected component in this graph, all leaves of a class have for label either the label of l or its dual. Therefore, any leaf in the class is labeled either X^+ or X^-. □

Corollary 6.78. *Take an ax-unique set of linkings θ on $[A^\perp * A, \Sigma]\ \Gamma$. A class for $A^\perp * A$ of θ contains either two or zero leaves not in $A^\perp * A$.*

Proof. As θ is *ax*-unique, in the GOI projection graph on $A^\perp * A$ of θ each vertex is of degree one or two – those in $A^\perp * A$ are those of degree two. By Lemma 3.50, this graph is a disjoint union of non-empty simple paths and cycles. Each leaf not in $A^\perp * A$ is an extremity of one of these paths, for they are of degree one. Thus, each class contains either two or zero leaves not in $A^\perp * A$, according to whether this class corresponds to a non-empty simple path or to a cycle. □

When restricted to MLL, using the correctness criterion we can prove a stronger result: each class contains exactly two leaves outside the cut pair, *i.e.* there is no cycle in the GOI projection graph. The case with additive is harder, because a $\& - \oplus_i$ key elimination case can erase linkings, and thus links.

Lemma 6.79. *Take an* $\mathsf{MLL}_{uf}^{0,2}$ *proof-net* θ *on* $[A^\perp * A, \Sigma]\ \Gamma$. *A class for* $A^\perp * A$ *of* θ *contains exactly two leaves not in* $A^\perp * A$.

Proof. By contradiction, assume there is a cycle in the GOI projection graph on $A^\perp * A$ of θ. In θ, reduce all cuts yielded from $A^\perp * A$ except those on these leaves, using only $\parr - \otimes$ key cases (this is convergent by Remark 4.13). The resulting proof-net ψ contains the GOI projection graph as a sub-graph, up to replacing the links $l — l^\perp$ for l in $A^\perp * A$ by $l \to * \leftarrow l^\perp$. In particular, a cycle in the GOI projection graph yields a cycle in ψ, which is a switching one as it uses only axiom links and premises of $*$-vertices. Contradiction with (P2). $\square$

Lemma 6.80. *Let* θ *be a set of linkings on* $[A^\perp * A]\ \Gamma$. *If there is a link* (l, m) *in the result of fully eliminating the cut-formula* $A^\perp * A$ *(well-defined by Remark 4.13), then* l *and* m *belong to the same class for* $A^\perp * A$ *in* θ *(and are not leaves of* $A^\perp * A$*).*

Proof. A link (l, m) of the reduced form is either a link of θ or one obtained from a sequence of $n + 2$ links (l, l_0), $((l_i^\perp, l_{i+1}))_{i \in [\![0;n-1]\!]}$, $(l_n^\perp, m)$ of θ where each l_i is a leaf of $A^\perp * A$, with as dual leaf $l_i^\perp$ in this cut pair (Definition 4.14 of Turbo Cut-elimination). In both cases, l and m belong to the same class for $A^\perp * A$ in θ. $\square$

We can now prove our main result, which will then be instantiated to the case of isomorphisms, and to retractions in the next chapter.

Theorem 6.81 (Renaming preserves composition to identities)**.** *Let* θ *and* θ' *be proof-nets respectively on* $A^\perp, B$ *and* $B^\perp, A$. *Assume* θ *and* θ' *are ax-unique and half-bipartite in respectively* $A^\perp$ *and* A. *Furthermore, suppose* $\theta \overset{B}{\bowtie} \theta'$ *reduces by cut-elimination to* id_A. *There exist renamings* α *and* α' *respectively of* θ *and* θ' *such that:*

(1) $\alpha(A^\perp)$ *and* $\alpha'(A)$ *are non-ambiguous;*

(2) $\alpha'(B^\perp) = \alpha(B)^\perp$*;*

(3) $\alpha(A^\perp) = \alpha'(A)^\perp$*;*

(4) $\alpha(\theta) \overset{\alpha(B)}{\bowtie} \alpha'(\theta')$ *reduces by cut-elimination to* $\mathrm{id}_{\alpha'(A)}$*;*

(5) if $\theta' \overset{A}{\bowtie} \theta$ *reduces by cut-elimination to* id_B, *then* $\alpha'(\theta') \overset{\alpha(A)}{\bowtie} \alpha(\theta)$ *reduces by cut-elimination to* $\mathrm{id}_{\alpha(B)}$.

Proof. We begin by defining α and α'. Let f be any injective function from the classes for $B^\perp * B$ of $\theta \overset{B}{\bowtie} \theta'$ to the set of positive atoms of $\mathcal{X}$, existing by cardinality. Consider $\mathcal{C}$ one of these classes; all leaves inside are of labels X^+ or X^- for some unsigned atom X (Lemma 6.77). For a leaf l inside this class we define:

- if l is in $A^\perp$ or B, and of label X^+ (resp. X^-), set $\alpha(l) = f(\mathcal{C})$ (resp. $\alpha(l) = f(\mathcal{C})^\perp$);

- if l is in A or $B^\perp$, and of label X^+ (resp. X^-), set $\alpha'(l) = f(\mathcal{C})$ (resp. $\alpha'(l) = f(\mathcal{C})^\perp$).

This defines functions α and α', respectively on leaves of $A^\perp, B$ and of $B^\perp, A$, which are renamings respectively of θ and θ'. Indeed, we only have to check that any link $a = (l, m)$ is between dual leaves after the renaming (Lemma 6.68). As l and m belong to the same class, and are of dual labels before the renaming, they are also of dual labels after the renaming by definition. We now prove the five wanted items.

As $\theta \overset{B}{\bowtie} \theta'$ reduces by cut-elimination to id_A, whose axiom links are the $(l, l^\perp)$ for l in A (Lemma 6.44), according to Lemma 6.80 we have l and $l^\perp$ in the same class. But $\theta \overset{B}{\bowtie} \theta'$ is *ax*-unique for θ and θ' are (Lemma 6.41), so by Corollary 6.78 these two leaves are the only ones of $A^\perp, A$ in this class. Thus, $\alpha(A^\perp)$ is non-ambiguous, as all leaves of $A^\perp$ belong to distinct classes. A similar result holding for $\alpha'(A)$, this ensures Item (1).

That $\alpha'(B^\perp) = \alpha(B)^\perp$ follows from the fact that for a leaf l of B, labeled X, the dual leaf $l^\perp$ of $B^\perp$ belongs to the same equivalence class, with as label $X^\perp$. Hence, Item (2) holds.

Using Lemma 6.74, as well as Item (2) we just proved, we obtain Items (3) and (4). Finally, suppose now that $\theta' \overset{A}{\bowtie} \theta$ reduces by cut-elimination to id_B. One proves Item (5) thanks to Lemma 6.74. $\square$

Lemma 6.82 (Distributed ambiguous isomorphic formulas)**.** *Let A and B be distributed formulas such that $A \overset{\theta,\theta'}{\simeq} B$. There exist an atomic-substitution σ and distributed formulas A' and B',* non-ambiguous*, such that $A = \sigma(A')$, $B = \sigma(B')$ and $A' \overset{\psi,\psi'}{\simeq} B'$ for some proof-nets ψ and ψ'.*

Proof. The proof-nets θ and θ' are bipartite *ax*-unique (Theorem 6.61), on cut sequents $A^\perp, B$ and $B^\perp, A$ respectively. By Theorem 6.81, there exists renamings α and α', respectively of θ and θ', such that $\alpha(A^\perp)$ and $\alpha'(A)$ are non-ambiguous, and $\alpha'(A) \overset{\alpha(\theta),\alpha'(\theta')}{\simeq} \alpha(B)$, with in particular $\alpha(A^\perp) = \alpha'(A)^\perp$ and $\alpha'(B^\perp) = \alpha(B)^\perp$.

Pose $A' \coloneqq \alpha'(A)$ and $B' \coloneqq \alpha(B)$, hence $A' \overset{\alpha(\theta),\alpha'(\theta')}{\simeq} B'$ and $\alpha(\theta)$ has for sequent ${A'}^\perp, B'$. Formulas A' and B' are distributed, as renaming acts only on leaves, and non-ambiguous (for $\alpha(A^\perp)$, $\alpha(B)$ are non-ambiguous and Lemma 6.64).

On $\alpha(\theta)$ one can define a renaming α^{-1} such that $\alpha^{-1}(B') = B$, as $\alpha(\theta)$ is cut-free, bipartite and *ax*-unique (Lemma 6.68). Remark it is equivalent to define α^{-1} on $\alpha(\theta)$ or only on leaves of B'. Moreover, $\theta = \alpha^{-1}(\alpha(\theta))$ and then $\alpha^{-1}({A'}^\perp) = A^\perp$ follow from $\alpha^{-1}(B') = B$.

Recall that all leaves of B' are on distinct unsigned atoms by definition of non-ambiguous. Hence, the definition of α^{-1} on B' can be seen as a substitution σ on signed atoms of B', with:

- if X occurs as X^+ in B', then $\sigma(X) = \alpha^{-1}(l(X^+))$ where $l(X^+)$ is the unique leaf of B' with label X^+;
- if X occurs as X^- in B', then $\sigma(X) = \alpha^{-1}(l(X^-))^\perp$ where $l(X^-)$ is the unique leaf of B' with label X^-.

Thus, $\theta = \alpha^{-1}(\alpha(\theta)) = \sigma(\alpha(\theta))$: in particular, $\sigma(B') = B$ and $\sigma({A'}^\perp) = A^\perp$, so $\sigma(A') = A$ (by Fact 1.5). Finally, A' and B' are distributed non-ambiguous formulas such that $A' \overset{\alpha(\theta),\alpha'(\theta')}{\simeq} B'$, $A = \sigma(A')$ and $B = \sigma(B')$. $\square$

Corollary 6.83 (Reduction to distributed non-ambiguous formulas)**.** *The set of couples of distributed formulas A and B such that $A \simeq B$ is the set of instances, by an atomic-substitution, of couples of distributed* non-ambiguous *formulas A' and B' such that $A' \simeq B'$. It is also the set of instances, by a substitution, of couples of distributed* non-ambiguous *formulas A' and B' such that $A' \simeq B'$.*

Proof. By Theorem 6.36 and Lemma 6.82, given an isomorphism $A \simeq B$ between distributed formulas, one gets non-ambiguous distributed formulas A' and B' such that $A' \simeq B'$ and A (resp. B) is an instance by an atomic substitution of A' (resp. B'). This atomic substitution can also be seen as a "standard" substitution.

Reciprocally, a substitution (and thus also an atomic-substitution) preserves isomorphisms (Lemma B.6), so that given non-ambiguous distributed formulas A' and B' as well as a substitution σ, $A' \simeq B' \implies \sigma(A') \simeq \sigma(B')$. ☐

6.3.6.2 Simplification with non-ambiguous formulas

The goal of this part is to show isomorphisms between non-ambiguous formulas correspond simply to the existence of proof-nets, speaking no more about cut-elimination nor identity proof-nets. This is done through Theorem 6.90: when A is non-ambiguous, having proof-nets on $A^\perp, B$ and $B^\perp, A$ is enough to know their composition over B composes to id_A – simply because this is the sole proof-net on $A^\perp, A$. This result will be deduced by proving properties close to the ones of identity proof-nets from Section 6.3.3.

Lemma 6.84. *Let θ be a set of linkings on the sequent $A^\perp, A$, with A a non-ambiguous formula. Axiom links of θ are of the form $(l^\perp, l)$ for l a leaf of A.*

Proof. Let a be an axiom link of θ, between leaves l and m, with the label of l (resp. m) being X (resp. $X^\perp$). By non-ambiguity of A (and so $A^\perp$ by Lemma 6.64), each signed atom is the label of at most one leaf in $A^\perp, A$. Moreover, the leaf labeled $X^\perp$ is the dual of the leaf labeled X. Thus, $m = l^\perp$ and $a = (l, l^\perp)$. ☐

Remark 6.85. Proving Lemma 6.84 with the definition of non-ambiguous from [BD99] (Remark 6.65) is harder, even when assuming θ is a proof-net, unless one also suppose θ to be bipartite. Indeed, consider a formula such as $X^+ \parr X^-$: it is non-ambiguous for this definition, but there may be an axiom link between these two atoms. A similar remark holds for all results of this section.

Lemma 6.86. *Let θ be a set of linkings on $A^\perp, A$, with A a non-ambiguous formula. Take a linking $\lambda \in \theta$ and an additive vertex v in its additive resolution. The vertex $v^\perp$ is in the additive resolution of λ, and λ keeps for $v^\perp$ the dual premise it keeps for v.*

Proof. As v is in the additive resolution $A^\perp, A \upharpoonright \lambda$ of λ, one of its ancestor leaves, say l, is in $A^\perp, A \upharpoonright \lambda$: there is a link $a \in \lambda$ on it. By Lemma 6.84, $a = (l, l^\perp)$. But $l^\perp$ is an ancestor of $v^\perp$, so $v^\perp$ is in $A^\perp, A \upharpoonright \lambda$, with as premise the dual premise chosen for v. ☐

Lemma 6.87. *Let A be a non-ambiguous formula, θ and θ' proof-structures on $A^\perp, A$. Then $\theta = \theta'$.*

Proof. Take $\lambda \in \theta$ a linking. It is on some &-resolution R of $A^\perp, A$. By (P1), there exists a unique linking $\lambda' \in \theta'$ on R. We have to prove $\lambda = \lambda'$. They have the same additive resolution, for their choice on a $\oplus$-vertex P is determined by the premise taken for the &-vertex $P^\perp$, which is in R

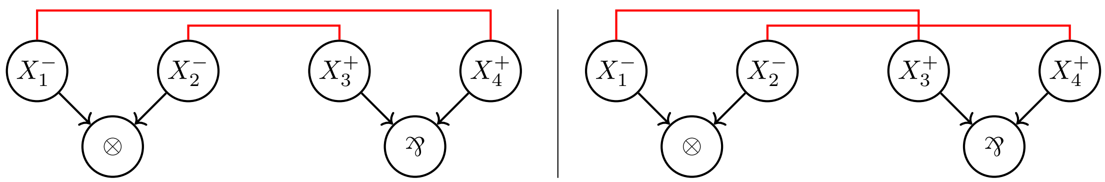


Figure 6.10: Identity proof-net (left-side) and swap (right-side) of $X^+ ⅋ X^+$

(Lemma 6.86). They have the same axiom links on this additive resolution, because any leaf on it is linked to its dual (Lemma 6.84). Therefore, $\lambda = \lambda'$, so $\theta \subseteq \theta'$. By symmetry, the same reasoning yields $\theta' \subseteq \theta$, thus $\theta = \theta'$. ☐

Corollary 6.88. *Let A be a non-ambiguous formula. There is exactly one proof-structure on $A^\perp, A$: the identity proof-net of A.*

Proof. This follows from Lemma 6.87. ☐

Remark 6.89. This property does not hold outside of non-ambiguous formulas, even distributed. For instance, there are two bipartite ax-unique proof-nets on $X_1^- \otimes X_2^-, X_3^+ ⅋ X_4^+$ (where each X_i is an occurrence of the atom X): the identity proof-net, with axiom links (X_1^-, X_4^+) and (X_2^-, X_3^+), and the "swap" with axiom links (X_1^-, X_3^+) and (X_2^-, X_4^+) – see Figure 6.10.

Theorem 6.90. *Let θ and θ' be proof-structures respectively on $A^\perp, B$ and $B^\perp, A$, with A a non-ambiguous formula. Then their composition over B reduces to the identity proof-net of A.*

Proof. The composition of θ and θ' by cut reduces to a proof-structure on $A^\perp, A$. By Corollary 6.88, this can only be the identity proof-net of A. ☐

Theorem 6.91 (Non-ambiguous isomorphisms)**.** *Let A and B be non-ambiguous formulas. If there exist proof-structures θ and ψ respectively on $A^\perp, B$ and $B^\perp, A$, then $A \overset{\theta,\psi}{\simeq} B$.*

Proof. By Theorem 6.90 both compositions yield identity proof-nets, whence $A \overset{\theta,\psi}{\simeq} B$. ☐

6.3.7 Completeness for unit-free MALL

We now prove the completeness of $\mathcal{L}^\dagger$ for MALL_{uf} by reasoning as in Section 4 of [BD99], with some more technicalities for we have to reorder not only ⅋-vertices but also &-vertices. Remember that a proof-net contains a sequentializing – *i.e.* terminal splitting – vertex (Lemma 4.31), and that removing a sequentializing vertex produces proof-net(s) (Lemmas 4.45 to 4.48).

Lemma 6.92. *In a bipartite full proof-net on $A_l \odot A_r, B$, where $\odot \in \{\otimes; \oplus\}$, the root of $A_l \odot A_r$ is not sequentializing.*

Proof. Let l be a leaf of A_l and r one of A_r. By bipartiteness and fullness, there are leaves m and s of B with axiom links (l, m) and (r, s) in the proof-net, see Figure 6.11. As there is a path in $T(B)$ between m and s, whether $\odot = \oplus$ or $\odot = \otimes$, it is not sequentializing. ☐

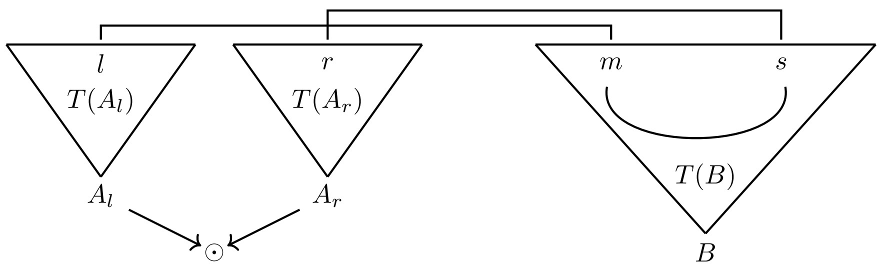


Figure 6.11: Illustration of the proof of Lemma 6.92

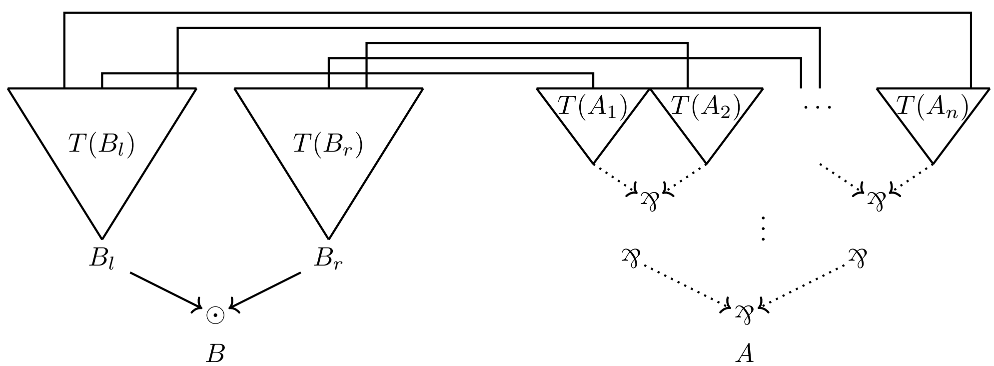


Figure 6.12: Proof-net of Lemma 6.93 with all terminal $\parr$-vertices removed

Lemma 6.93 (Reordering $\parr$-vertices)**.** *Let θ be a bipartite ax-unique proof-net on A, B with $A = A_l \parr A_r$, $B = B_l \odot B_r$, $\odot \in \{\otimes; \oplus\}$ and A a distributed formula. Then $\odot = \otimes$ and there exist two bipartite ax-unique proof-nets respectively on A'_l, B_l and A'_r, B_r, where $A'_l \parr A'_r$ is equal to $A_l \parr A_r$ up to associativity and commutativity of $\parr$.*

Proof. We remove all terminal (hence sequentializing) $\parr$-vertices, all in A, without modifying the linkings. The resulting graph is a proof-net on $A_1, \ldots, A_n, B_l \odot B_r$ (see Figure 6.12). The roots of the new trees A_i cannot be &-vertices because A is distributed: so they are $\otimes \backslash \oplus$-vertices or atoms. These $\otimes \backslash \oplus$-vertices are not sequentializing, since by bipartiteness and fullness every leaf of each A_i is connected to the formula $B_l \odot B_r$ (reasoning as in the proof of Lemma 6.92). Thus, the sequentializing vertex of this proof-net is necessarily $B_l \odot B_r$. It follows $\odot = \otimes$, because all leaves of B are connected to leaves in $A_1, \ldots, A_n$, so if $\odot = \oplus$ then $B_l \odot B_r$ cannot be sequentializing. Removing the sequentializing $B_l \otimes B_r$ gives two proof-nets, with a partition of the A_i into two classes: those linked to leaves of B_l and the others linked to leaves of B_r (and any leaf is in one of the two cases by fullness). We recover from these proof-nets bipartite *ax*-unique ones by adding $\parr$-vertices under the A_i in an arbitrary order, yielding formulas A'_l (with those linked to B_l) and A'_r (with those linked to B_r). As we only removed and put back $\parr$-vertices, $A'_l \parr A'_r$ is equal to $A_l \parr A_r$ up to associativity and commutativity of $\parr$. $\square$

Lemma 6.94 (Reordering &-vertices)**.** *Let θ be a bipartite ax-unique proof-net on A, B with $A = A_l \mathbin{\&} A_r$, $B = B_l \oplus B_r$ and A a distributed formula. Then there exist two bipartite ax-unique proof-nets respectively on A'_l, B_l and A'_r, B_r, where $A'_l \mathbin{\&} A'_r$ is equal to $A_l \mathbin{\&} A_r$ up to associativity and commutativity of &.*

Proof. We remove all terminal &-vertices in the proof-net, then all terminal ⅋-vertices, all in A. The resulting graphs are proof-nets θ_i (for terminal negative vertices are sequentializing), on cut sequent $A^i_1, \dots, A^i_{n_i}, B_l \oplus B_r$ for the i-th proof-net. An illustration is Figure 6.12, except we have several of these proof-nets, having in common exactly $T(B)$. As in the proof of Lemma 6.93, the roots of the new trees A^i_j cannot be negative vertices because the formula A is distributed: so they are $\otimes\backslash\oplus$-vertices or atoms. These $\otimes\backslash\oplus$-vertices cannot be sequentializing, since by bipartiteness and fullness every leaf of A^i_j is connected to the formula $B_l \oplus B_r$ (reasoning as in the proof of Lemma 6.92). Thus, the sequentializing vertex of each of these proof-nets is necessarily $B_l \oplus B_r$, and we can remove it: for a given i, all A^i_j are linked only to either B_l or B_r. We put back the ⅋-vertices we removed, in the very same order. We then put back the &-vertices we removed, but in another order: we put together all θ_i linked to B_l, and all those to B_r, yielding two proof-nets on B_l, A'_l and B_r, A'_r. These proof-nets are bipartite *ax*-unique ones (because adding and removing ⅋ does not modify the linkings, and & is disjoint union of linkings). We indeed have $A'_l \mathbin{\&} A'_r$ equal to $A_l \mathbin{\&} A_r$ up to associativity and commutativity of &, because we only reordered &-vertices. □

We conclude by induction on the size $s(A)$ of A, which we recall is given by its number of connectives, and is thus unaffected by commutation and associativity of connectives.

Theorem 6.95 (Isomorphisms completeness for MALL_{uf})**.** *Given A and B two distributed MALL_{uf} formulas, if $A \overset{\theta,\psi}{\simeq} B$ for some proof-nets θ and ψ, then $A =_{\mathcal{L}^\dagger} B$.*

Proof. We assume A and B to be non-ambiguous formulas by Corollary 6.83. We reason by induction on the size of A, $s(A)$.[1]

If A and B are atoms (*i.e.* of size 1), then the axiom link in θ between $A^\perp$ and B yields $A = B$ and the property holds. Otherwise, $A^\perp$ and B are *both* non atomic. By Theorem 6.61, θ and ψ are bipartite *ax*-unique. By Lemma 6.92, one of $A^\perp$ and B is negative, otherwise neither the root of $A^\perp$ nor B is sequentializing in θ, contradicting sequentialization (Theorem 4.18). A symmetric reasoning on ψ implies that the other formula is positive. Assume *w.l.o.g.* that $B = B_0 \odot B_1$ is positive (*i.e.* $\odot \in \{\otimes; \oplus\}$) and $A^\perp$ negative. We distinguish cases according to the kind of the roots of $A^\perp$ and B, considering the proof-net θ. If B a $\otimes$-formula and A a &-formula, we instead consider ψ of cut sequent $B^\perp, A$, where A is a $\oplus$-formula and $B^\perp$ a ⅋-formula. Whence, either $A^\perp$ is a ⅋-formula, or B and $A^\perp$ are respectively a $\oplus$-formula and a &-formula.

In the first (resp. second) case, by Lemma 6.93 (resp. Lemma 6.94) $\odot = \otimes$ (in the first case only) and there exist two bipartite *ax*-unique proof-nets θ_0 and θ_1 of respective cut sequents ${A'_0}^\perp, B_0$ and ${A'_1}^\perp, B_1$, with ${A'}^\perp = {A'_1}^\perp ⅋ {A'_0}^\perp$ (resp. ${A'}^\perp = {A'_1}^\perp \mathbin{\&} {A'_0}^\perp$) equal to $A^\perp$ up to associativity and commutativity of ⅋ (resp. &). In particular, ${A'}^\perp =_{\mathcal{L}^\dagger} A^\perp$, and $s(A'_0)$ and $s(A'_1)$ are both less than $s(A)$. To conclude, we only need proof-nets ψ_0 and ψ_1 of respective cut sequents $B_0^\perp, A'_0$ and $B_1^\perp, A'_1$. We will then apply Theorem 6.91 to obtain $A'_0 \overset{\theta_0,\psi_0}{\simeq} B_0$ and $A'_1 \overset{\theta_1,\psi_1}{\simeq} B_1$. Thence, by induction

[1] Remark that $s(A) = s(B)$, because θ is bipartite *ax*-unique (Theorem 6.61), thence A and B have the same number of leaves, so of connectives as they are all binary ones (Fact 1.3).

hypothesis, $A'_0 =_{\mathcal{L}^\dagger} B_0$ and $A'_1 =_{\mathcal{L}^\dagger} B_1$, thus $A' =_{\mathcal{L}^\dagger} B$.[2] As $A =_{\mathcal{L}^\dagger} A'$, we will finally conclude $A =_{\mathcal{L}^\dagger} B$.

Thus, we look for two proof-nets of respective cut sequents $B_0^\perp, A'_0$ and $B_1^\perp, A'_1$. As $A \overset{\theta,\psi}{\simeq} B$, and $A \simeq A'$ by soundness of $\mathcal{L}^\dagger$ (Theorem 6.3), it follows using Theorem 6.36 that $A' \overset{\Theta,\Theta'}{\simeq} B$ for some proof-nets Θ and Θ'.[3] Furthermore, Θ is a bipartite *ax*-unique proof-net (Theorem 6.61) of cut sequent $B^\perp, A'$, *i.e.* on $B_1^\perp \odot^\perp B_0^\perp, A'_0 \circledcirc A'_1$ with $(\odot^\perp, \circledcirc) \in \{(⅋, \otimes); (\&, \oplus)\}$. We had a bipartite *ax*-unique proof-net θ_0 on ${A'_0}^\perp, B_0$, therefore the signed atoms of B_0 are exactly those of A'_0. Similarly, the signed atoms of B_1 are exactly those of A'_1. Whence, no atom of B_0 (resp. B_1) is one of A'_1 (resp. A'_0), for otherwise an atom of A'_0 (resp. A'_1) also occurs in A'_1 (resp. A'_0), contradicting non-ambiguousness of A'. This implies that axiom links in Θ must be between leaves of $B_0^\perp$ and A'_0, and between leaves of $B_1^\perp$ and A'_1. Therefore, once we sequentialize the negative root $\odot^\perp$ of $B^\perp$ in Θ, the positive root $\circledcirc$ of A' is sequentializing. After sequentializing both, we obtain two bipartite *ax*-unique proof-nets, of respective cut sequents $B_0^\perp, A'_0$ and $B_1^\perp, A'_1$. □

The above theorem, associated with preceding results, yields our main contribution for this chapter.

Theorem 6.96 (Isomorphisms completeness)**.**

$\mathcal{L}^\dagger$ is complete for MALL^2_{uf} *Given A and B two MALL^2_{uf} formulas, if $A \simeq B$, then $A =_{\mathcal{L}^\dagger} B$.*

$\mathcal{L}$ is complete for MALL^2 *Given A and B two MALL^2 formulas, if $A \simeq B$, then $A =_{\mathcal{L}} B$.*

Proof. We assume A and B to be distributed formulas by Proposition 6.13. Using Lemma 6.24, we can consider these two sub-systems without the mix_2-rule. The first item is a consequence of Theorems 6.36 and 6.95. The second results from the first one and Theorem 6.32. □

Remark 6.97. Looking at the proof, we get as a sub-result that isomorphisms for various sub-systems of MALL^2 are given by the equational theory $\mathcal{L}$ (resp. $\mathcal{L}^\dagger$) restricted to their formulas, thanks to the sub-formula property:

- $\mathcal{L}$ restricted to formulas of MLL is sound and complete for isomorphisms of MLL and MLL^2;
- $\mathcal{L}^\dagger$ restricted to formulas of MLL_{uf} is sound and complete for isomorphisms of MLL_{uf} and MLL^2_{uf};
- $\mathcal{L}$ restricted to formulas of ALL is sound and complete for isomorphisms of ALL;
- $\mathcal{L}^\dagger$ restricted to formulas of ALL_{uf} is sound and complete for isomorphisms of ALL_{uf}.

6.4 Star-autonomous categories with finite products

Since MALL semantically corresponds to $\star$-autonomous categories with finite products (which also have finite coproducts), we can use our results on MALL to characterize isomorphisms valid in all such categories. For the historical result of how linear logic can be seen as a category, see [See89].

[2] The formula A' is distributed for it is equal up to associativity and commutativity to the distributed A. Whence, A'_0 and A'_1 are also distributed.

[3] Using that isomorphisms form equivalence classes on formulas according to Lemma B.3.

$$\mathcal{D}\left\{\begin{array}{l} \left.\begin{array}{rclcrclcrcl} F\otimes(G\otimes H) &=& (F\otimes G)\otimes H && F\otimes G &=& G\otimes F && F\otimes 1 &=& F \\ (F\otimes G)\multimap H &=& F\multimap(G\multimap H) && && && 1\multimap F &=& F \\ \\ F\,\&\,(G\,\&\,H) &=& (F\,\&\,G)\,\&\,H && F\,\&\,G &=& G\,\&\,F && F\,\&\,\top &=& F \\ F\multimap(G\,\&\,H) &=& (F\multimap G)\,\&\,(F\multimap H) && && && F\multimap\top &=& \top \end{array}\right\}\mathcal{S} \\ \\ \qquad\qquad\qquad\qquad (F\multimap\bot)\multimap\bot = F \end{array}\right.$$

Table 6.2: Type isomorphisms in $\star$-autonomous categories with finite products

We consider objects of $\star$-autonomous categories described by formulas in the language:

$$F ::= X \mid F\otimes F \mid 1 \mid F\multimap F \mid \bot \mid F\,\&\,F \mid \top$$

where $X\in\mathcal{X}$ for $\mathcal{X}$ a fixed set. There is some redundancy here since one could define 1 as $\bot\multimap\bot$ in any $\star$-autonomous category, but we prefer to keep 1 in the language as it is at the core of monoidal categories. Our goal is to prove the theory $\mathcal{D}$ of Table 6.2 to be sound and complete for the isomorphisms of $\star$-autonomous categories with finite products.

We establish this result from the one on MALL, first proving that MALL (with proofs considered up to $\beta\eta$-equality) defines a $\star$-autonomous category with finite products (Section 6.4.1). Then, we conclude using a semantic method based on this syntactic category (Section 6.4.2). In a third step we look at the more general case of symmetric monoidal closed categories – without the requirement of a dualizing object (Section 6.4.3).

6.4.1 MALL as a star-autonomous category with finite products

The logic MALL, with proofs taken up to $\beta\eta$-equality, defines a $\star$-autonomous category with finite products, that we will call $\mathbb{MALL}$. Indeed, we can define it as follows.

Objects of $\mathbb{MALL}$ are formulas of MALL, while its morphisms from A to B are proofs of $\vdash A^\perp, B$, considered up to $\beta\eta$-equality and composed by the *cut*-rule.[4] By definition, a proof of MALL is an isomorphism if and only if, when seen as a morphism, it is an isomorphism in $\mathbb{MALL}$.

We define a bifunctor $\otimes$ on $\mathbb{MALL}$, associating to formulas (*i.e.* objects) A and B the formula $A\otimes B$ and to proofs (*i.e.* morphisms) π_0 and π_1 respectively of $\vdash A_0^\perp, B_0$ and $\vdash A_1^\perp, B_1$ the following proof of $\vdash (A_0\otimes A_1)^\perp, B_0\otimes B_1$:

$$\dfrac{\dfrac{\begin{array}{c}\pi_0\\ \vdash A_0^\perp, B_0\end{array}\qquad\begin{array}{c}\pi_1\\ \vdash A_1^\perp, B_1\end{array}}{\vdash A_1^\perp, A_0^\perp, B_0\otimes B_1}\,(\otimes)}{\vdash A_1^\perp \parr A_0^\perp, B_0\otimes B_1}\,(\parr)$$

One can check that $(\mathbb{MALL}, \otimes, 1, \alpha, \lambda, \rho, \gamma)$ forms a symmetric monoidal category, where 1 is the 1-formula, α are isomorphisms of MALL associated to $(A\otimes B)\otimes C \simeq A\otimes(B\otimes C)$ seen as natural

[4] We recall that $(\cdot)^\perp$ is defined by induction, making it an involution.

isomorphisms of $\mathbb{MALL}$, and similarly for λ with $1 \otimes A \simeq A$, ρ with $A \otimes 1 \simeq A$, and γ with $A \otimes B \simeq B \otimes A$.

Furthermore, define $A \multimap B \coloneqq A^\perp \parr B$ and $ev_{A,B}$ as the following morphism from $A \otimes (A \multimap B)$ to B (*i.e.* a proof of $\vdash (B^\perp \otimes A) \parr A^\perp, B$):

$$\dfrac{\dfrac{\dfrac{}{\vdash B^\perp, B}\,(ax) \qquad \dfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash B^\perp \otimes A, A^\perp, B}\,(\otimes)}{\vdash (B^\perp \otimes A) \parr A^\perp, B}\,(\parr)$$

It can be checked that $\mathbb{MALL}$ is a symmetric monoidal closed category with as exponential object $(A \multimap B, ev_{A,B})$ for objects A and B.

Moreover, one can also check that $\perp$ is a dualizing object for this category, making $\mathbb{MALL}$ a $\star$-autonomous category. This relies on the following morphism from $(A \multimap \perp) \multimap \perp$ to A (which is an inverse of the currying of $ev_{A,\perp}$):

$$\dfrac{\dfrac{}{\vdash 1}\,(1) \qquad \dfrac{\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash A^\perp, \perp, A}\,(\perp)}{\vdash A^\perp \parr \perp, A}\,(\parr)}{\vdash 1 \otimes (A^\perp \parr \perp), A}\,(\otimes)$$

Finally, $\top$ is a terminal object of $\mathbb{MALL}$, and $A \,\&\, B$ is the product of objects A and B, with, as projections π_A and π_B, the following morphisms respectively from $A \,\&\, B$ to A and from $A \,\&\, B$ to B:

$$\dfrac{\dfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash B^\perp \oplus A^\perp, A}\,(\oplus_2) \qquad \text{and} \qquad \dfrac{\dfrac{}{\vdash B^\perp, B}\,(ax)}{\vdash B^\perp \oplus A^\perp, B}\,(\oplus_1)$$

Therefore, $\mathbb{MALL}$ is a $\star$-autonomous category with finite products [See89].

6.4.2 Isomorphisms of star-autonomous categories with finite products

We translate formulas in the language of $\star$-autonomous categories with finite products into MALL formulas and conversely by means of the following translations, with $\ell(\_)$ from the category to the logic and $\partial(\_)$ in the reverse direction:[5]

$$\begin{array}{rcl@{\qquad}rcl}
\ell(X) & = & X^+ & \partial(X^+) & = & X \\
 & & & \partial(X^-) & = & X \multimap \perp \\
\ell(F \otimes G) & = & \ell(F) \otimes \ell(G) & \partial(A \otimes B) & = & \partial(A) \otimes \partial(B) \\
\ell(1) & = & 1 & \partial(1) & = & 1 \\
\ell(F \multimap G) & = & \ell(F)^\perp \parr \ell(G) & \partial(A \parr B) & = & ((\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp)) \multimap \perp \\
\ell(\perp) & = & \perp & \partial(\perp) & = & \perp \\
\ell(F \,\&\, G) & = & \ell(F) \,\&\, \ell(G) & \partial(A \,\&\, B) & = & \partial(A) \,\&\, \partial(B) \\
\ell(\top) & = & \top & \partial(\top) & = & \top \\
 & & & \partial(A \oplus B) & = & ((\partial(A) \multimap \perp) \,\&\, (\partial(B) \multimap \perp)) \multimap \perp \\
 & & & \partial(0) & = & \top \multimap \perp
\end{array}$$

The translation $\ell(\_)$ corresponds exactly to the interpretation of the constructions on objects of a

[5] We set the same $\mathcal{X}$ for the set of unsigned atoms of MALL and for the base elements of the category.

$\star$-autonomous categories with finite products in the concrete category $\mathbb{MALL}$.

Lemma 6.98. *The $\partial(\_)$ and $\ell(\_)$ translations satisfy the following properties:*

- $\partial(A^{\perp}) =_{\mathcal{D}} \partial(A) \multimap \perp$
- $\partial(\ell(F)) =_{\mathcal{D}} F$
- $A =_{\mathcal{L}} B$ *entails* $\partial(A) =_{\mathcal{D}} \partial(B)$

Proof. The second property relies on the first while the third is independent.

- By induction on A:

$$\begin{aligned}
\partial((X^+)^{\perp}) &= \partial(X^-) = X \multimap \perp =_{\mathcal{D}} \partial(X^+) \multimap \perp \\
\partial((X^-)^{\perp}) &= \partial(X^+) = X =_{\mathcal{D}} (X \multimap \perp) \multimap \perp = \partial(X^-) \multimap \perp \\
\partial((A \otimes B)^{\perp}) &= ((\partial(B^{\perp}) \multimap \perp) \otimes (\partial(A^{\perp}) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (((\partial(A) \multimap \perp) \multimap \perp) \otimes ((\partial(B) \multimap \perp) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \otimes \partial(B)) \multimap \perp = \partial(A \otimes B) \multimap \perp \\
\partial((A \parr B)^{\perp}) &= \partial(B^{\perp}) \otimes \partial(A^{\perp}) =_{\mathcal{D}} (\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp) \\
&=_{\mathcal{D}} (((\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp)) \multimap \perp) \multimap \perp \\
&= \partial(A \parr B) \multimap \perp \\
\partial(1^{\perp}) &= \perp =_{\mathcal{D}} 1 \multimap \perp = \partial(1) \multimap \perp \\
\partial(\perp^{\perp}) &= 1 =_{\mathcal{D}} (1 \multimap \perp) \multimap \perp =_{\mathcal{D}} \perp \multimap \perp = \partial(\perp) \multimap \perp \\
\partial((A \,\&\, B)^{\perp}) &= ((\partial(B^{\perp}) \multimap \perp) \,\&\, (\partial(A^{\perp}) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (((\partial(A) \multimap \perp) \multimap \perp) \,\&\, ((\partial(B) \multimap \perp) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \,\&\, \partial(B)) \multimap \perp = \partial(A \,\&\, B) \multimap \perp \\
\partial((A \oplus B)^{\perp}) &= \partial(B^{\perp}) \,\&\, \partial(A^{\perp}) =_{\mathcal{D}} (\partial(A) \multimap \perp) \,\&\, (\partial(B) \multimap \perp) \\
&=_{\mathcal{D}} (((\partial(A) \multimap \perp) \,\&\, (\partial(B) \multimap \perp)) \multimap \perp) \multimap \perp \\
&= \partial(A \oplus B) \multimap \perp \\
\partial(\top^{\perp}) &= \top \multimap \perp = \partial(\top) \multimap \perp \\
\partial(0^{\perp}) &= \top =_{\mathcal{D}} (\top \multimap \perp) \multimap \perp = \partial(0) \multimap \perp
\end{aligned}$$

- By induction on F:

$$\begin{aligned}
\partial(\ell(X)) &= \partial(X^+) = X \\
\partial(\ell(F \otimes G)) &= \partial(\ell(F) \otimes \ell(G)) = \partial(\ell(F)) \otimes \partial(\ell(G)) =_{\mathcal{D}} F \otimes G \\
\partial(\ell(1)) &= \partial(1) = 1
\end{aligned}$$

$$
\begin{aligned}
\partial(\ell(F \multimap G)) &= \partial(\ell(F)^{\perp} \parr \ell(G)) \\
&= ((\partial(\ell(F)^{\perp}) \multimap \perp) \otimes (\partial(\ell(G)) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (((F \multimap \perp) \multimap \perp) \otimes (G \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (F \otimes (G \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} F \multimap ((G \multimap \perp) \multimap \perp) =_{\mathcal{D}} F \multimap G \\
\partial(\ell(\perp)) &= \partial(\perp) = \perp \\
\partial(\ell(F \,\&\, G)) &= \partial(\ell(F) \,\&\, \ell(G)) = \partial(\ell(F)) \,\&\, \partial(\ell(G)) =_{\mathcal{D}} F \,\&\, G \\
\partial(\ell(\top)) &= \partial(\top) = \top
\end{aligned}
$$

- We prove that the image of each equation of Table 6.1 through $\partial(\_)$ is derivable with equations of Table 6.2. Commutativity, associativity and unitality for $\otimes$ and $\&$ are immediate.

$$
\begin{aligned}
\partial(A \parr B) &= ((\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} ((\partial(B) \multimap \perp) \otimes (\partial(A) \multimap \perp)) \multimap \perp = \partial(B \parr A) \\
\partial(A \parr (B \parr C)) &= ((\partial(A) \multimap \perp) \otimes ((((\partial(B) \multimap \perp) \otimes (\partial(C) \multimap \perp)) \multimap \perp) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} ((\partial(A) \multimap \perp) \otimes ((\partial(B) \multimap \perp) \otimes (\partial(C) \multimap \perp))) \multimap \perp \\
&=_{\mathcal{D}} (((\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp)) \otimes (\partial(C) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (((((\partial(A) \multimap \perp) \otimes (\partial(B) \multimap \perp) \multimap \perp) \multimap \perp)) \otimes (\partial(C) \multimap \perp)) \multimap \perp \\
&= \partial((A \parr B) \parr C) \\
\partial(A \parr \perp) &= ((\partial(A) \multimap \perp) \otimes (\perp \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} ((\partial(A) \multimap \perp) \otimes ((1 \multimap \perp) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} ((\partial(A) \multimap \perp) \otimes 1) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \multimap \perp) \multimap \perp =_{\mathcal{D}} \partial(A)
\end{aligned}
$$

Commutativity, associativity and unitality for $\oplus$ follow the same pattern as for $\parr$.

Then we have:

$$
\begin{aligned}
\partial(A \otimes (B \oplus C)) &= \partial(A) \otimes (((\partial(B) \multimap \perp) \,\&\, (\partial(C) \multimap \perp)) \multimap \perp) \\
&=_{\mathcal{D}} ((\partial(A) \otimes (((\partial(B) \multimap \perp) \,\&\, (\partial(C) \multimap \perp)) \multimap \perp)) \multimap \perp) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \multimap (((\partial(B) \multimap \perp) \,\&\, (\partial(C) \multimap \perp)) \multimap \perp) \multimap \perp) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \multimap ((\partial(B) \multimap \perp) \,\&\, (\partial(C) \multimap \perp))) \multimap \perp \\
&=_{\mathcal{D}} ((\partial(A) \multimap (\partial(B) \multimap \perp)) \,\&\, (\partial(A) \multimap (\partial(C) \multimap \perp))) \multimap \perp \\
&=_{\mathcal{D}} (((\partial(A) \otimes \partial(B)) \multimap \perp) \,\&\, ((\partial(A) \otimes \partial(C)) \multimap \perp)) \multimap \perp \\
&= \partial((A \otimes B) \oplus (A \otimes C)) \\
\partial(A \otimes 0) &= \partial(A) \otimes (\top \multimap \perp) \\
&=_{\mathcal{D}} ((\partial(A) \otimes (\top \multimap \perp)) \multimap \perp) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \multimap ((\top \multimap \perp) \multimap \perp)) \multimap \perp \\
&=_{\mathcal{D}} (\partial(A) \multimap \top) \multimap \perp \\
&=_{\mathcal{D}} \top \multimap \perp = \partial(0)
\end{aligned}
$$

$$
\begin{aligned}
\partial(A \mathbin{⅋} (B \mathbin{\&} C)) &= ((\partial(A) \multimap \bot) \otimes ((\partial(B) \mathbin{\&} \partial(C)) \multimap \bot)) \multimap \bot \\
&=_{\mathcal{D}} (\partial(A) \multimap \bot) \multimap (((\partial(B) \mathbin{\&} \partial(C)) \multimap \bot) \multimap \bot) \\
&=_{\mathcal{D}} (\partial(A) \multimap \bot) \multimap (\partial(B) \mathbin{\&} \partial(C)) \\
&=_{\mathcal{D}} ((\partial(A) \multimap \bot) \multimap \partial(B)) \mathbin{\&} ((\partial(A) \multimap \bot) \multimap \partial(C)) \\
&=_{\mathcal{D}} ((\partial(A) \multimap \bot) \multimap ((\partial(B) \multimap \bot) \multimap \bot)) \mathbin{\&} \\
&\qquad\qquad ((\partial(A) \multimap \bot) \multimap ((\partial(C) \multimap \bot) \multimap \bot)) \\
&=_{\mathcal{D}} (((\partial(A) \multimap \bot) \otimes (\partial(B) \multimap \bot)) \multimap \bot) \mathbin{\&} \\
&\qquad\qquad (((\partial(A) \multimap \bot) \otimes (\partial(C) \multimap \bot)) \multimap \bot) \\
&= \partial((A \mathbin{⅋} B) \mathbin{\&} (A \mathbin{⅋} C)) \\
\partial(A \mathbin{⅋} \top) &= ((\partial(A) \multimap \bot) \otimes (\top \multimap \bot)) \multimap \bot \\
&=_{\mathcal{D}} (\partial(A) \multimap \bot) \multimap ((\top \multimap \bot) \multimap \bot) \\
&=_{\mathcal{D}} (\partial(A) \multimap \bot) \multimap \top =_{\mathcal{D}} \top = \partial(\top)
\end{aligned}
$$

□

Theorem 6.99 (Isomorphisms in $\star$-autonomous categories with finite products)**.** *The equational theory $\mathcal{D}$ (Table 6.2) is sound and complete for isomorphisms in $\star$-autonomous categories with finite products.*

Proof. Soundness follows by definition of $\star$-autonomous categories with finite products. For completeness, take an isomorphism $F \simeq G$. It yields an isomorphism $\ell(F) \simeq \ell(G)$ in $\mathbb{MALL}$. As isomorphisms in $\mathbb{MALL}$ are generated by $\mathcal{L}$ (Theorem 6.96), we get $\ell(F) =_{\mathcal{L}} \ell(G)$. From Lemma 6.98, we deduce $F =_{\mathcal{D}} \partial(\ell(F)) =_{\mathcal{D}} \partial(\ell(G)) =_{\mathcal{D}} G$. □

6.4.3 Isomorphisms of symmetric monoidal closed categories with finite products

Isomorphisms in symmetric monoidal closed categories (SMCC) have been characterized [DP97] and proved to correspond to equations in the first two lines of Table 6.2.

We want to extend this result to finite products by proving the soundness and completeness of the theory $\mathcal{S}$ presented in Table 6.2.

Theorem 6.100 (Isomorphisms in SMCC with finite products)**.** *The equational theory $\mathcal{S}$ (Table 6.2) is sound and complete for isomorphisms in symmetric monoidal closed categories with finite products.*

Proof. The language of SMCC with finite products is the language of $\star$-autonomous categories with finite products in which we remove $\bot$. In particular the translation $\ell(\_)$ can be used to translate the associated formulas into MALL formulas. In order to analyze the image of this restricted translation, we consider the following grammar of MALL formulas (*output formulas o* and *input formulas ι* [Lam96]):

$$
\begin{array}{lcccccccccccccc}
o & ::= & X^+ & \mid & o \otimes o & \mid & o \mathbin{⅋} \iota & \mid & \iota \mathbin{⅋} o & \mid & 1 & \mid & o \mathbin{\&} o & \mid & \top \\
\iota & ::= & X^- & \mid & \iota \mathbin{⅋} \iota & \mid & \iota \otimes o & \mid & o \otimes \iota & \mid & \bot & \mid & \iota \oplus \iota & \mid & 0
\end{array}
$$

The dual of an output formula is an input formula and conversely. Moreover no MALL formula is both an input and an output formula – let us call this the **non-ambiguity property**. One can check by induction on a formula F in the language of SMCC with finite products that $\ell(F)$ is an output formula. We define a translation back from output formulas to SMCC formulas (which is well defined thanks to the non-ambiguity property):

$$\begin{array}{rclrcl} X^{+\circ} & = & X & & & \\ (o \otimes o')^\circ & = & o^\circ \otimes o'^\circ & 1^\circ & = & 1 \\ (o \,\&\, o')^\circ & = & o^\circ \,\&\, o'^\circ & \top^\circ & = & \top \\ (o \parr \iota)^\circ & = & (\iota^\perp)^\circ \multimap o^\circ & (\iota \parr o)^\circ & = & (\iota^\perp)^\circ \multimap o^\circ \end{array}$$

We use the notation $\iota^\bullet = (\iota^\perp)^\circ$ (so that $(o^\perp)^\bullet = o^\circ$). We can check, by induction on F, that $\ell(F)^\circ = F$.

$$\begin{aligned} \ell(X)^\circ &= X^{+\circ} = X \\ \ell(F \otimes G)^\circ &= (\ell(F) \otimes \ell(G))^\circ = \ell(F)^\circ \otimes \ell(G)^\circ = F \otimes G \\ \ell(1)^\circ &= 1^\circ = 1 \\ \ell(F \multimap G)^\circ &= (\ell(F)^\perp \parr \ell(G))^\circ = \ell(F)^\circ \multimap \ell(G)^\circ = F \multimap G \\ \ell(F \,\&\, G)^\circ &= (\ell(F) \,\&\, \ell(G))^\circ = \ell(F)^\circ \,\&\, \ell(G)^\circ = F \,\&\, G \\ \ell(\top)^\circ &= \top^\circ = \top \end{aligned}$$

We now prove that when $o =_{\mathcal{L}} A$ (resp. $\iota =_{\mathcal{L}} A$) is an equation from Table 6.1 (or its symmetric version) then A is an output (resp. input) formula and $o^\circ =_{\mathcal{S}} A^\circ$ (resp. $\iota^\bullet =_{\mathcal{S}} A^\bullet$):

- If o or ι is of the shape $A \otimes (B \otimes C)$, we have the following possibilities:
  - A, B and C are output, then $(A \otimes B) \otimes C$ is output and:

$$\begin{aligned} (A \otimes (B \otimes C))^\circ &= A^\circ \otimes (B^\circ \otimes C^\circ) \\ &=_{\mathcal{S}} (A^\circ \otimes B^\circ) \otimes C^\circ = ((A \otimes B) \otimes C)^\circ \end{aligned}$$

  - A and B are output and C is input, then $(A \otimes B) \otimes C$ is input and:

$$\begin{aligned} (A \otimes (B \otimes C))^\bullet &= A^\circ \multimap (B^\circ \multimap C^\bullet) \\ &=_{\mathcal{S}} (A^\circ \otimes B^\circ) \multimap C^\bullet = ((A \otimes B) \otimes C)^\bullet \end{aligned}$$

  - A and C are output and B is input, then $(A \otimes B) \otimes C$ is input and:

$$\begin{aligned} (A \otimes (B \otimes C))^\bullet &= A^\circ \multimap (C^\circ \multimap B^\bullet) \\ &=_{\mathcal{S}} (A^\circ \otimes C^\circ) \multimap B^\bullet =_{\mathcal{S}} (C^\circ \otimes A^\circ) \multimap B^\bullet \\ &=_{\mathcal{S}} C^\circ \multimap (A^\circ \multimap B^\bullet) = ((A \otimes B) \otimes C)^\bullet \end{aligned}$$

  - A is input and B and C are output, then $(A \otimes B) \otimes C$ is input and:

$$\begin{aligned} (A \otimes (B \otimes C))^\bullet &= (B^\circ \otimes C^\circ) \multimap A^\bullet \\ &=_{\mathcal{S}} (C^\circ \otimes B^\circ) \multimap A^\bullet \\ &=_{\mathcal{S}} C^\circ \multimap (B^\circ \multimap A^\bullet) = ((A \otimes B) \otimes C)^\bullet \end{aligned}$$

The symmetric case follows the same pattern, as well as associativity of $\parr$.

- If o or ι is of the shape $A \otimes B$, we have the following possibilities:
  - A and B are output, then $B \otimes A$ is output and $(A \otimes B)^\circ = A^\circ \otimes B^\circ =_{\mathcal{S}} B^\circ \otimes A^\circ = (B \otimes A)^\circ$
  - A is output and B is input, then $B \otimes A$ is input and $(A \otimes B)^\bullet = A^\circ \multimap B^\bullet = (B \otimes A)^\bullet$
  - A is input and B is output, then $B \otimes A$ is input and $(A \otimes B)^\bullet = B^\circ \multimap A^\bullet = (B \otimes A)^\bullet$

  The commutativity of $\parr$ follows the same pattern.

- If o or ι is of the shape $A \otimes 1$ then either A is output and $(A \otimes 1)^\circ = A^\circ \otimes 1 =_{\mathcal{S}} A^\circ$, or A is input and $(A \otimes 1)^\bullet = 1 \multimap A^\bullet =_{\mathcal{S}} A^\bullet$. The symmetric case follows the same pattern, as well as unitality for $\parr$.

- If $o = A \,\&\, (B \,\&\, C)$ then A, B and C are output and $(A \,\&\, B) \,\&\, C$ as well. We have $(A \,\&\, (B \,\&\, C))^\circ = A^\circ \,\&\, (B^\circ \,\&\, C^\circ) =_{\mathcal{S}} (A^\circ \,\&\, B^\circ) \,\&\, C^\circ = ((A \,\&\, B) \,\&\, C)^\circ$. The symmetric case follows the same pattern, as well as associativity of $\oplus$.

- If $o = A \,\&\, B$ then A and B are output and $B \,\&\, A$ as well. We have $(A \,\&\, B)^\circ = A^\circ \,\&\, B^\circ =_{\mathcal{S}} B^\circ \,\&\, A^\circ = (B \,\&\, A)^\circ$. The commutativity of $\oplus$ follows the same pattern.

- If $o = A \,\&\, \top$ then A is output and $(A \,\&\, \top)^\circ = A^\circ \,\&\, \top =_{\mathcal{S}} A^\circ$. The symmetric case follows the same pattern, as well as unitality for $\oplus$.

- If $\iota = A \otimes (B \oplus C)$ then A is output and B and C are input, and $(A \otimes B) \oplus (A \otimes C)$ is input. We have:

$$\begin{aligned}(A \otimes (B \oplus C))^\bullet &= A^\circ \multimap (C^\bullet \,\&\, B^\bullet)\\ &=_{\mathcal{S}} (A^\circ \multimap C^\bullet) \,\&\, (A^\circ \multimap B^\bullet) = ((A \otimes B) \oplus (A \otimes C))^\bullet\end{aligned}$$

  The symmetric case follows the same pattern, as well as distributivity of $\parr$ over $\&$.

- If $\iota = A \otimes 0$ then A is output and $(A \otimes 0)^\bullet = A^\circ \multimap \top =_{\mathcal{S}} \top = 0^\bullet$. The symmetric case follows the same pattern, as well as cancellation of $\parr$ by $\top$.

Assume now that $F \simeq G$ in the class of SMCC with finite products. As $\mathbb{MALL}$ is such a SMCC with finite products, we have $\ell(F) \simeq \ell(G)$ in $\mathbb{MALL}$, thus $\ell(F) =_{\mathcal{L}} \ell(G)$ by Theorem 6.96. As $\ell(F)$ is an output formula, by induction on the length of the equational derivation of $\ell(F) =_{\mathcal{L}} \ell(G)$, we get that all the intermediary steps involve output formulas and each equation is mapped to $=_{\mathcal{S}}$ by $(\_)^\circ$ so that $\ell(F)^\circ =_{\mathcal{S}} \ell(G)^\circ$, and finally $F = \ell(F)^\circ =_{\mathcal{S}} \ell(G)^\circ = G$.

Conversely soundness easily comes from the definition of SMCC and products. $\square$

In $\star$-autonomous categories with finite products, we automatically have finite coproducts given by $A \oplus B \coloneqq ((A \multimap \bot) \,\&\, (B \multimap \bot)) \multimap \bot$ and $0 \coloneqq \top \multimap \bot$. From equations of Table 6.2, one can derive:

$$\left.\begin{array}{rcl} A \oplus B &=& B \oplus A\\ A \oplus (B \oplus C) &=& (A \oplus B) \oplus C\\ A \oplus 0 &=& A\\ A \otimes (B \oplus C) &=& (A \otimes B) \oplus (A \otimes C)\\ A \otimes 0 &=& 0\\ (A \oplus B) \multimap C &=& (A \multimap C) \,\&\, (B \multimap C)\\ 0 \multimap C &=& \top \end{array}\right\}\mathcal{C}$$

In the weaker setting of SMCC, finite products do not induce finite coproducts. It justifies the possibility of considering them separately. The case of products only was Theorem 6.100. We are now looking at both products and coproducts on one side, and coproducts only on the other side.

As a preliminary result, let us give a necessary condition for formulas to be isomorphic in this symmetric monoidal closed setting.

Lemma 6.101. *If $A \simeq B$ in symmetric monoidal closed categories with finite products and coproducts with A and B distributed (*i.e.* $\ell(A)$ and $\ell(B)$ are distributed) then there exist cut-free proofs of $A \vdash B$ and $B \vdash A$ in* IMALL *(intuitionistic multiplicative additive linear logic) [Bie95] whose left 0 rules introduce $0 \vdash 0$ sequents only and right $\top$ rules introduce $\top \vdash \top$ sequents only.*

Proof. If A and B are isomorphic in SMCC with finite products and coproducts, the associated isomorphisms can be represented as IMALL proofs which we can assume to be cut-free and atomic-axiom. These proofs can be interpreted as MALL proofs (corresponding to the fact that $\mathbb{MALL}$ is an SMCC with finite products and coproducts). By Lemma 6.25, these MALL proofs have their $\top$ rules introducing $\vdash \top, 0$ sequents only which gives the required property on the IMALL proofs we started with. $\square$

We conjecture that isomorphisms in SMCC with both finite products *and* finite coproducts correspond to adding the equations of theory $\mathcal{C}$ to $\mathcal{S}$ (Table 6.2). However, our approach through $\star$-autonomous categories does not work since for example $\top \multimap (\top \oplus \top)$ and $(0 \mathbin{\&} 0) \multimap 0$ are isomorphic in $\star$-autonomous categories but not in SMCC with finite products and coproducts:

$$\begin{aligned}
\top \multimap (\top \oplus \top) &=_{\mathcal{D}} \top \multimap (((\top \multimap \bot) \mathbin{\&} (\top \multimap \bot)) \multimap \bot)\\
&=_{\mathcal{D}} (\top \otimes (0 \mathbin{\&} 0)) \multimap \bot =_{\mathcal{D}} ((0 \mathbin{\&} 0) \otimes \top) \multimap \bot\\
&=_{\mathcal{D}} (0 \mathbin{\&} 0) \multimap (\top \multimap \bot) =_{\mathcal{D}} (0 \mathbin{\&} 0) \multimap 0
\end{aligned}$$

or directly by interpreting these formulas into MALL, one gets:

$$\ell(\top \multimap (\top \oplus \top)) =_{\mathcal{L}} 0 \mathbin{⅋} (\top \oplus \top) =_{\mathcal{L}} (\top \oplus \top) \mathbin{⅋} 0 =_{\mathcal{L}} \ell((0 \mathbin{\&} 0) \multimap 0)$$

(the analogue of the non-ambiguity property does not hold). This isomorphism however is not valid in the SMCC setting as shown by Lemma 6.101 since all cut-free proofs of $(0\&0) \multimap 0 \vdash \top \multimap (\top\oplus\top)$ in IMALL have the following shape:

$$\dfrac{\dfrac{\dfrac{}{(0 \mathbin{\&} 0) \multimap 0, \top \vdash \top}\ (\top R)}{(0 \mathbin{\&} 0) \multimap 0, \top \vdash \top \oplus \top}\ (\oplus_i R)}{((0 \mathbin{\&} 0) \multimap 0 \vdash \top \multimap (\top \oplus \top)}\ (\multimap R)$$

One could also investigate SMCC with finite coproducts only (without products). It is important to notice that an initial object 0 in a SMCC induces that $0 \multimap A$ is a terminal object for any A. This first means that we cannot uncorrelate completely products and coproducts. It also means that the theory of isomorphisms includes the equation $0 \multimap A \simeq 0 \multimap B$ even if it does not occur in $\mathcal{C}$ (it might be the only missing equation). Regarding a characterization through $\star$-autonomous categories, it is again not possible since, if we denote by $\top$ a terminal object, $(\top \multimap 0) \multimap 0$ and

$\top \multimap (\top \otimes \top)$ are isomorphic in $\star$-autonomous categories (using $0 =_{\mathcal{D}} \top \multimap \bot$):

$$\begin{aligned}(\top \multimap 0) \multimap 0 &=_{\mathcal{D}} (\top \multimap (\top \multimap \bot)) \multimap (\top \multimap \bot)\\ &=_{\mathcal{D}} ((\top \otimes \top) \multimap \bot) \multimap (\top \multimap \bot)\\ &=_{\mathcal{D}} (((\top \otimes \top) \multimap \bot) \otimes \top) \multimap \bot\\ &=_{\mathcal{D}} (\top \otimes ((\top \otimes \top) \multimap \bot)) \multimap \bot\\ &=_{\mathcal{D}} \top \multimap (((\top \otimes \top) \multimap \bot) \multimap \bot) =_{\mathcal{D}} \top \multimap (\top \otimes \top)\end{aligned}$$

(or $\ell((\top \multimap 0) \multimap 0) =_{\mathcal{L}} (\top \otimes \top) \parr 0 =_{\mathcal{L}} 0 \parr (\top \otimes \top) =_{\mathcal{L}} \ell(\top \multimap (\top \otimes \top))$ in MALL). But they are not isomorphic in SMCC with initial and terminal objects by Lemma 6.101 since all cut-free proofs of $(\top \multimap 0) \multimap 0 \vdash \top \multimap (\top \otimes \top)$ in IMALL have the following shape (up to permuting the premises of the ($\otimes$R)-rule):

$$\dfrac{\dfrac{\dfrac{}{(\top \multimap 0) \multimap 0 \vdash \top}(\top R) \quad \dfrac{}{\top \vdash \top}(\top R)}{(\top \multimap 0) \multimap 0, \top \vdash \top \otimes \top}(\otimes R)}{(\top \multimap 0) \multimap 0 \vdash \top \multimap (\top \otimes \top)}(\multimap R) \quad \text{or} \quad \dfrac{\dfrac{\dfrac{}{(\top \multimap 0) \multimap 0, \top \vdash \top}(\top R) \quad \dfrac{}{\top}(\top R)}{(\top \multimap 0) \multimap 0, \top \vdash \top \otimes \top}(\otimes R)}{(\top \multimap 0) \multimap 0 \vdash \top \multimap (\top \otimes \top)}(\multimap R)$$

6.5 Isomorphisms up to cut-elimination only

This section is a quick digression on what happens if proofs are not considered up to $\beta\eta$-equality, but only β-equality. This means that in the definition of isomorphisms (Definition 6.1) one replaces $=_{\beta\eta o}$ by $=_{\beta}$.

To begin with, remark that considering proofs only up to η-equality is nonsensical from the point of view of isomorphisms (and also retractions, discussed in the next chapter): when composing two proofs, one gets a proof with a *cut*-rule, that cannot be removed up to η-equality. So there are no formulas A and B that are isomorphic if one consider only $=_{\eta}$ in Definition 6.1!

Let us prove that, in $\mathsf{MALL}^{0,2}$, two formulas are isomorphic, when considering proofs only up to $=_{\beta}$, if and only if there are equal. More precisely, we prove that the only cut-free isomorphisms are proofs composed of a unique ax-rule. This means that even if it looks like axiom-expansion brings nothing, this is true only once axioms are expanded.

Lemma 6.102. *Let A and B be two formulas in $\mathsf{MALL}^{0,2}$ along proofs π of $\vdash A^{\bot}, B$ and ϕ of $\vdash B^{\bot}, A$ such that $\pi \overset{B}{\bowtie} \phi =_{\beta} ax_A$ and $\phi \overset{A}{\bowtie} \pi =_{\beta} ax_B$. Then $A = B$.*

Proof. Without any loss of generality, one can assume π and ϕ to be cut-free proofs, up to eliminating cuts thanks to weak normalization of cut-elimination (Corollary 2.43). Set τ_A (resp. τ_B) any cut-free proof such that $\pi \overset{B}{\bowtie} \phi \xrightarrow{\beta^*} \tau_A$ (resp. $\phi \overset{A}{\bowtie} \pi \xrightarrow{\beta^*} \tau_A$). Thanks to Theorem 2.49, one gets $\tau_A \overset{r^*}{\vdash\!\dashv} ax_A$ and $\tau_B \overset{r^*}{\vdash\!\dashv} ax_B$. But no rule commutation can be applied in a proof made of only an ax-rule: $\tau_A = ax_A$ and $\tau_B = ax_B$ for any normal forms τ_A and τ_B of $\pi \overset{B}{\bowtie} \phi$ and $\phi \overset{A}{\bowtie} \pi$ respectively.

Assume by contradiction that the rule r at the root of π is not an ax-rule, and by symmetry suppose its main formula is not $A^{\bot}$. Remark r cannot be a mix_0-rule as π is not on the empty sequent. Then, in $\pi \overset{B}{\bowtie} \phi$ the rule r commutes with the *cut*-rule. Doing this cut-elimination step, followed by a normalization of the resulting proof (Corollary 2.43), one gets a proof with at its root a rule of the same kind as r. But the resulting proof should be ax_A by our previous reasoning: contradiction.

Therefore π, and likewise ϕ, is composed of a unique ax-rule, which implies in particular that $A = B$. □

Remark in the above proof that we needed both compositions to lead to axioms, which is why this result cannot be generalized to the case of retractions.

Finally, one could imagine to modify the definition of isomorphisms (Definition 6.1) to keep only β-equality, but asking for the composition to be equal to id_A instead of ax_A. This gives the exact same theory as the one described here thanks to Lemma 6.6, even in propositional linear logic with all optional rules (*i.e.* the full system excepted for quantifiers).

6.6 Perspectives

Extending the result of Balat and Di Cosmo in [BD99], we gave an equational theory characterizing type isomorphisms in multiplicative-additive linear logic with units as well as in $\star$-autonomous categories with finite products: the one described on Table 6.1 on Page 199 (and on Table 6.2 for $\star$-autonomous categories). Proof-nets were a major tool to prove completeness, as notions like fullness and ax-uniqueness are much harder to define and manipulate in sequent calculus. However, we could not use them for taking care of the (additive) units, because there is no known appropriate notion of proof-nets for MALL with additive units. We have thus been forced to work in the sequent calculus, considering rule commutations.

Remark that we characterized not only type isomorphisms but also the isomorphisms themselves! They are obtained through the compositions of the "canonical ones" given in the proof of Theorem 6.3 (all proofs being considered up to $\beta\eta$-equality). Indeed, reducing to distributed formulas can be done using the isomorphisms associated to distributivity, renaming to a non-ambiguous unit-free formula acts only on labels, and we conclude at the end with a study of isomorphisms (under their proof-net form) in Theorem 6.95.

The immediate question to address is the extension of our results to the characterization of type isomorphisms for LL, including the exponential connectives. This is clearly not immediate since the interaction between additive and exponential connectives is not well described in proof-nets, in particular the $\overset{oe}{\rightsquigarrow}$ transformation which is needed to get the exponential isomorphisms. Another direction is considering MALL_{uf} with quantifiers, for proof-nets can described this sub-system.

Conjecture 6.103. *An equational theory sound and complete for isomorphisms of linear logic without the optional rules, but with the ?-Rétoré transformation as well as identification of proofs up to $! - ?_c$ and $! - ?_w$ commutations (see Remark 1.15), is the one given by the equations of $\mathcal{L}$ (Table 6.1 on Page 199) with in addition the following equations:*

Exponentials	*Quantifiers*
$!(A \mathbin{\&} B) = !A \otimes !B$	$\forall X(A \mathbin{\&} B) = \forall X A \mathbin{\&} \forall X B$
$?(A \oplus B) = ?A \parr ?B$	$\exists X(A \oplus B) = \exists X A \oplus \exists X B$
$!\top = 1$	$\forall X \top = \top$
$?0 = \bot$	$\exists X 0 = 0$
	$\forall X A \parr B = \forall X(A \parr B)$ *if X not free in B*
	$\exists X A \otimes B = \exists X(A \otimes B)$ *if X not free in B*
	$\forall X \forall Y A = \forall Y \forall X A$
	$\exists X \exists Y A = \exists Y \exists X A$

(the same equations are conjectured for first order and second order quantifiers, e.g. $\forall_1 X \forall_2 Y A = \forall_2 Y \forall_1 X A$*).*

Like always, it is easy to prove soundness, and all the difficulty lies in completeness. A good definition of a distributive formula in the general setting would be to rewrite the non-commutative equations of the above table from left to right (up to commutativity of the connectives, as we did for distributivity):

$!(A \mathbin{\&} B) \to !A \otimes !B$	$!\top \to 1$	$?(A \oplus B) \to ?A \parr ?B$	$?0 \to \bot$
$\forall X(A \mathbin{\&} B) \to \forall X A \mathbin{\&} \forall X B$	$\forall X \top \to \top$	$\exists X(A \oplus B) \to \exists X A \oplus \exists X B$	$\exists X 0 \to 0$
$\forall X A \parr B \to \forall X(A \parr B)$ if X not free in B		$\exists X A \otimes B \to \exists X(A \otimes B)$ if X not free in B	
$B \parr \forall X A \to \forall X(B \parr A)$ if X not free in B		$B \otimes \exists X A \to \exists X(B \otimes A)$ if X not free in B	

This way, one can assume there is no & nor $\top$ below a ! and that there is no $\forall$ below a $\parr$ (as X not free in B can always be satisfied up to α-renaming), reminiscent of what we did here for distributivity (no & below a $\parr$).

Adding optional rules or Rétoré transformations may create new isomorphisms. For instance, in $\mathsf{MALL}^{\emptyset}$ one has $\top \simeq 0$ (and more generally, this is true as soon as $\vdash 0, 0$ is provable). Indeed, the two associated proofs are:

$$\dfrac{}{\vdash 0, 0}\ (\emptyset) \qquad\qquad \dfrac{}{\vdash \top, \top}\ (\top)$$

where it does not matter on which occurrence of $\top$ the $\top$-rule is applied, for both possibilities are related by $\overset{r}{\longmapsto\!\dashv}$, hence by $=_\beta$. One can check that the compositions of these proofs yield identities, using a single $\top - \mathit{cut}$ step (and possibly before a $\top - \top$ commutation to have the $\top$-rule on the non-*cut* occurrence).

Chapter 7

Retractions for Multiplicative Linear Logic

Retractions are a generalization of isomorphisms: in category theory, A is a *retract* of B, denoted $A \trianglelefteq B$, if there exist morphisms $A \xrightarrow{f} B$ and $B \xrightarrow{g} A$ such that $g \circ f = \mathrm{id}_A$ – as in the following commutative diagram:

$$\begin{tikzcd} A \arrow[loop left, "\mathrm{id}_A"] \arrow[rr, bend left, "f"] & & B \arrow[ll, bend left, "g"] \end{tikzcd}$$

This provides a natural notion of sub-typing: the morphism f can be seen as an encoding of elements of A into elements of B, and g as a decoding. As for isomorphisms, one would like an inequational theory characterizing retractions, with the same difficulties for the completeness part. Nevertheless, there are way fewer results about retractions than isomorphisms in the literature. A reasonable explanation for this lack of results would be its difficulty. At the best of my knowledge, an inequational theory corresponding to retractions has only been given for simply typed affine λ-calculus (where each abstraction uses its argument at most once) for inhabited atomic types in [RU02][1] – which extends a similar result from [dPS92] to the case with many atoms – and for simply typed λ-calculus in [BL85] with terms taken only up to β-equality, not $\beta\eta$-equality. The corresponding inequational theory from [RU02] is the one on Table 7.1, with isomorphisms $\simeq$ and strict retractions $\lhd$ which are not isomorphisms. Other results, weaker than an inequational theory, include for instance the decidability of retractions in the simply typed λ-calculus [Pad01].

The question of retractions is mostly open in the case of linear logic with not even a conjecture on what retractions are, in contrast to isomorphisms where an equational theory is conjectured for full linear logic (see Conjecture 6.103 in Section 6.6). Even in MLL, the simpler sub-system for most problems, no characterization is known. What is known is only a strict retraction (a retraction which is not an isomorphism), the *Beffara retraction* $A \lhd (A \multimap A) \multimap A = (A \otimes A^{\perp}) \parr A$. One

[1] More precisely, there is a subtlety in the definition of retraction from [RU02, Page 1]: a type A is a retract of B if there exists a type environment E, terms M and N that can be typed as $E \vdash M : A \to B$ and $E \vdash N : B \to A$ such that $N \circ M =_{\beta\eta} id_A$. In the case E is empty, they call it an embedding instead of a retraction, and what is characterized are affine retractions. In the case where all atomic types are inhabited, both concepts are equivalent.

$\simeq$	$\lhd$
$A \to B \to C \simeq B \to A \to C$	$A \lhd B \to A$
	$A \lhd (A \to X) \to X$ if A is $Y_1 \to Y_2 \to \cdots \to X$

Table 7.1: Inequational theory for retractions in simply typed affine λ-calculus [RU02]

can check the proof-nets associated to this retraction are not bipartite. In bigger sub-systems, such as MALL, the problem looks even harder, with more retractions; for instance the one depicted on Figure 6.1, but there also is a retraction $A \lhd A \oplus A$. Even only considering exponentials, one has new retractions such as $!A \trianglelefteq !!A$ and $?!A \trianglelefteq ?!?!A$. This is in sharp contrast with isomorphisms, where isomorphisms for MELL are conjectured to be the same as in MLL.

In all that follows, we implicitly assume to be in $\mathsf{MLL}^{0,2}$, retractions being complicated enough even in this restricted case. Notice we already know the isomorphisms of MLL^2 from the previous chapter. Moreover, MLL is quite close to the simply typed linear λ-calculus (terms where each argument is used exactly one), and the solution to the wider simply typed affine λ-calculus gives an over-theory for retractions in the linear case – because adding more constructors or equations, or relaxing constraints, can only give more retractions. As in the preceding chapter, we will use a syntactic method and not a semantic one, because it is hard to guess a model preserving retractions when one does not even known which retractions there are, so what the model should respect. We give in this chapter some results about retractions in this setting, and in particular a proof of the Cantor-Bernstein-Schröeder property, which was not proved before at the best of our knowledge. Unfortunately, we did not manage to give an inequational theory for the general problem of strict retractions $A \lhd B$. Still, we managed to solve the problem for $X \lhd B$ with X a signed atom. In other words, we found all universal super-types in multiplicative linear logic.

Outline We start by giving the necessary definitions: retractions, the Cantor-Bernstein-Schröeder property, … (Section 7.1). We then proceed in a similar fashion as we did for isomorphisms in the previous chapter, with less difficulties as we can reuse properties proved in this chapter. We prove units do not give more retractions than if they were just atoms, except for their unitality isomorphisms; this allows us to reduce the problem to MLL_{uf}, and hence to use proof-nets. We also reduce the problem in the presence of mix_2- and mix_0-rules, possibly with *mix*-Rétoré, *i.e.* reduce the $\mathsf{MLL}^{0,2}$ case, to the MLL_{uf} one (Section 7.2). Mimicking what we did for isomorphisms, we can again reduce to non-ambiguous formulas, except that because the type-retraction relation is not symmetric, we can only do it for one of the two formulas (Section 7.3). At this point, we have enough to prove the Cantor-Bernstein-Schröeder property for MLL (Section 7.4). Afterwards, we proceed with some general results, which are unfortunately insufficient to solve the general case (Section 7.5). Still, associated with some more specific lemmas they allow us to find all retracts to an atom (Section 7.6). We end this chapter by explaining why our method does not scale to the general case (Section 7.7), and by speaking about decidability and complexity of retractions in wider sub-systems of linear logic (Section 7.8). As an aside, we argue retractions in $\mathsf{MLL}^{0,2}$ without axiom-expansion are retractions to an atom (Section 7.9).

7.1 Linear Retractions

Definition 7.1 (Retraction)**.** Consider (a sub-system of) linear logic with $\overset{o}{\vdash\!\dashv}$ some (possibly none, or all) of the Rétoré transformations. There is a **retraction** between formulas A and B, denoted $A \trianglelefteq B$, if there exist proofs π of $\vdash A^{\perp}, B$ and π' of $\vdash B^{\perp}, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A$:

$$\pi \overset{B}{\bowtie} \pi' \ = \ \dfrac{\overset{\pi}{\vdash A^{\perp}, B} \qquad \overset{\pi'}{\vdash B^{\perp}, A}}{\vdash A^{\perp}, A}\,(cut) \ =_{\beta\eta o} \ \dfrac{}{\vdash A^{\perp}, A}\,(ax) \ = \ ax_A$$

When $A \trianglelefteq B$ is a retraction, we say A is the **retract** formula whereas B **retracts** to A. We also call $A \trianglelefteq B$ a retraction **to** A.

Remark 7.2. By definition, if $A \trianglelefteq B$ then $A^{\perp} \trianglelefteq B^{\perp}$, with the very same proofs. Notice it is *not* $B^{\perp} \trianglelefteq A^{\perp}$!

Fact 7.3 ($\simeq \subseteq \trianglelefteq$)**.** *By definition, if $A \simeq B$ then $A \trianglelefteq B$,* i.e. $\simeq \subseteq \trianglelefteq$.

Definition 7.4 (Strict Retraction)**.** We say a formula A is a **strict retract** of a formula B, that we denote $A \triangleleft B$, when $A \trianglelefteq B$ and $A \not\simeq B$.

A general result one can wish for is the Cantor-Bernstein-Schröeder property.

Definition 7.5 (Cantor-Bernstein-Schröeder property)**.** A logic (or more generally a category) has the **Cantor-Bernstein-Schröeder property** when the following holds: if $A \trianglelefteq B$ and $B \trianglelefteq A$ then $A \simeq B$.

The Cantor-Bernstein-Schröeder property is also called Cantor-Bernstein or Schröeder-Bernstein property. We choose the full name to remove any possible ambiguity. This property does not hold directly by the definitions of retractions and isomorphisms, because the same proofs must compose to the identities in both directions so as to yield an isomorphism, while the two retractions can give different proofs (of the same sequents).

This notion comes as a generalization of the corresponding result from set theory about injections, which is here in terms of split monomorphisms. A stronger notion of Cantor-Bernstein-Schröeder property exists, with monomorphisms instead of split monomorphisms.

The Cantor-Bernstein-Schröeder property also has a stronger variant where we look not only on objects but also on the underlying morphisms.

Definition 7.6 (Strong Cantor-Bernstein-Schröeder property)**.** A logic (or more generally a category) has the **strong Cantor-Bernstein-Schröeder property** when the following holds: if $A \trianglelefteq B$, with as morphisms $\pi : A \to B$ and $\pi' : B \to A$, and $B \trianglelefteq A$, then $A \simeq B$, with as morphisms $\pi : A \to B$ and $\pi' : B \to A$.

As for isomorphisms, one can consider normal forms for the involved proofs.

Definition 7.7. We write $A \overset{\pi,\pi'}{\trianglelefteq} B$ when the *atomic-axiom*, *cut-free*, $\overset{o}{\rightsquigarrow}$*-normal* proofs π of $\vdash A^{\perp}, B$ and π' of $\vdash B^{\perp}, A$ respects $\pi \overset{B}{\bowtie} \pi' =_{\beta o} \mathrm{id}_A$.

Lemma 7.8. *Given two formulas A and B, $A \trianglelefteq B$ if and only if there exist proofs π and π' such that $A \overset{\pi,\pi'}{\trianglelefteq} B$.*

Proof. The converse way follows by definition of a retraction. For the direct way, take proofs π and π' given by the definition of a retraction. One can expand all axioms, eliminate all cuts and apply all $\overset{o}{\rightsquigarrow}$ in π and π' (Proposition 2.8 and Corollary 2.43) to obtain respectively ϕ and ϕ'. Using Lemma 6.5, $\phi \overset{B}{\bowtie} \phi' =_{\beta o} \mathrm{id}_A$. Hence, $A \overset{\phi,\phi'}{\trianglelefteq} B$. □

7.2 Reduction to unit-free MLL and proof-nets

As for isomorphisms, the problem can be reduced to the unit-free case, which will allow us to use proof-nets. We do this in three times, reducing the problem from $\mathsf{MLL}^{0,2}$ (with potentially the *mix*-Rétoré transformation) to proof-nets of MLL_{uf}. First, we prove optional multiplicative rules and the *mix*-Rétoré transformation add at most one isomorphism, so that retractions in $\mathsf{MLL}^{0,2}$ can be deduced from those in MLL (Section 7.2.1). Then, we demonstrate that while units add the unitary isomorphisms, they do not add any other retractions; thence the problem of retractions in MLL is reduced to the one in MLL_{uf} (Section 7.2.2). Finally, in the unit-free case the proof-net syntax can be considered, not only for MLL_{uf} but even for MALL_{uf} (Section 7.2.3).

7.2.1 Optional multiplicative rules

We study here what happens when taking (some of) the optional multiplicative rules and (maybe) the *mix*-Rétoré transformation $\overset{\rho m}{\mapsto}$. We prove that adding both optional multiplicative rules does not add any retraction, unless one also adds the *mix*-Rétoré transformation, in which case the added retractions are exactly given by the isomorphism $1 \simeq \bot$.

Lemma 7.9. *In* $\mathsf{MLL}^{0,2}$ *with* $\overset{\rho m}{\mapsto}$, $\bot \simeq 1$.

Proof. Consider the two following proofs:

$$\pi = \dfrac{\dfrac{}{\vdash 1_2}\,(1) \qquad \dfrac{}{\vdash 1_1}\,(1)}{\vdash 1_2, 1_1}\,(mix_2) \qquad\qquad \pi' = \dfrac{\dfrac{\dfrac{}{\vdash}\,(mix_0)}{\vdash \bot_2}\,(\bot)}{\vdash \bot_1, \bot_2}\,(\bot)$$

One can check that:

$$
\begin{array}{rl}
\pi \stackrel{1_1}{\bowtie} \pi' & \xrightarrow{\overline{\beta}} \dfrac{\dfrac{}{\vdash 1_2}\,(1) \qquad \dfrac{\dfrac{}{\vdash 1_1}\,(1) \quad \dfrac{\dfrac{\dfrac{}{\vdash}\,(mix_0)}{\vdash \bot_2}\,(\bot)}{\vdash \bot_1, \bot_2}\,(\bot)}{\vdash \bot_2}\,(cut)}{\vdash 1_2, \bot_2}\,(mix_2) \\[2em]
& \xrightarrow{\overline{\beta}} \dfrac{\dfrac{}{\vdash 1_2}\,(1) \quad \dfrac{\dfrac{}{\vdash}\,(mix_0)}{\vdash \bot_2}\,(\bot)}{\vdash 1_2, \bot_2}\,(mix_2) \\[2em]
& \stackrel{r^*}{\vdash\!\dashv} \dfrac{\dfrac{\dfrac{}{\vdash 1_2}\,(1) \quad \dfrac{}{\vdash}\,(mix_0)}{\vdash 1_2}\,(mix_2)}{\vdash 1_2, \bot_2}\,(\bot) \\[2em]
& \stackrel{om}{\rightsquigarrow} \mathrm{id}_{\bot_2}
\end{array}
$$

And:

$$
\begin{array}{rl}
\pi' \stackrel{\bot_2}{\bowtie} \pi & \xrightarrow{\overline{\beta}+} \dfrac{\dfrac{\dfrac{\dfrac{}{\vdash 1_2}\,(1) \quad \dfrac{\dfrac{}{\vdash}\,(mix_0)}{\vdash \bot_2}\,(\bot)}{\vdash}\,(cut)}{\vdash \bot_1}\,(\bot) \qquad \dfrac{}{\vdash 1_1}\,(1)}{\vdash \bot_1, 1_2}\,(mix_2) \\[2em]
& \xrightarrow{\overline{\beta}} \dfrac{\dfrac{\dfrac{}{\vdash}\,(mix_0)}{\vdash \bot_1}\,(\bot) \quad \dfrac{}{\vdash 1_2}\,(1)}{\vdash \bot_1, 1_2}\,(mix_2) \\[2em]
& \stackrel{r}{\vdash\!\dashv} \dfrac{\dfrac{\dfrac{}{\vdash}\,(mix_0) \quad \dfrac{}{\vdash 1_2}\,(1)}{\vdash 1_2}\,(mix_2)}{\vdash \bot_1, 1_2}\,(\bot) \\[2em]
& \stackrel{om}{\rightsquigarrow} \mathrm{id}_{1_1}
\end{array}
$$

Using Proposition 2.46, $\bot_2 \stackrel{\pi,\pi'}{\simeq} 1_1$ follows. □

Fact 7.10. *Consider $\pi =_\beta \pi'$ in $\mathsf{MLL}^{0,2}$, without the $\stackrel{om}{\vdash\!\dashv}$ transformation. Then π has a mix_2-rule (resp. mix_0-rule) if and only if π' has such a rule.*

Proof. Because in this logical system no cut-elimination step can erase nor produce a mix_2- or mix_0-rule. □

Lemma 7.11. *In $\mathsf{MLL}^{0,2}$, with or without $\stackrel{om}{\vdash\!\dashv}$, a rule with as premise the empty sequent $\vdash$ can only be a $\bot$- or mix_2-rule.*

Proof. Such a rule cannot be an *ax*-, $\parr$-, $\otimes$-, 1- nor mix_0-rule, as they have no premise or have all premises with at least one formula. The only other rules in this system are the $\perp$- and mix_2-rules. $\square$

Lemma 7.12. *In* $\mathsf{MLL}^{0,2}$ *with* $\overset{pm}{\mapsto}$ *but without* $\perp$*, proofs in normal form for* $\overset{om}{\rightsquigarrow}$ *are* mix_0*-free or equal to* $\dfrac{}{\vdash}\,(mix_0)$.

Proof. Consider a proof with a mix_0-rule. Using Lemma 7.11, a rule below it can only be a $\perp$- or mix_2-rule. But there is no $\perp$, and a mix_2-rule below would yield a possible $\overset{om}{\rightsquigarrow}$ reduction. $\square$

Lemma 7.13. *In* MLL^2*, there is no formula* A *such that* $\vdash A$ *and* $\vdash A^\perp$ *are both provable.*

Proof. If it were the case, we could compose both proofs to get a mix_0-free proof of the empty sequent $\vdash$. Reducing cuts in this proof (Corollary 2.43), one would get a cut-free proof of $\vdash$. But the only rules of MLL^2 that can have for conclusion the empty sequent are a *cut*-rule or a mix_2-rule between two proofs of $\vdash$, which leads to a contradiction. $\square$

Proposition 7.14.

- *The retractions of* $\mathsf{MLL}^{0,2}$ *without* $\overset{pm}{\mapsto}$ *are exactly the retractions of* MLL *(and thus this also holds for* MLL^2 *and* MLL^0*).*
- *The retractions of* $\mathsf{MLL}^{0,2}$ *with* $\overset{pm}{\mapsto}$ *are the retractions of* MLL *possibly composed with* $\perp \simeq 1$.

Proof. The first item can be easily proven, and the second one can be reduced to the first. For the first item, by Lemma 7.8 take a retraction $A \overset{\pi,\pi'}{\trianglelefteq} B$ in $\mathsf{MLL}^{0,2}$ without $\overset{pm}{\mapsto}$. By Fact 7.10 one has $\pi \overset{B}{\bowtie} \pi' =_\beta \mathrm{id}_A$ where in no proof of this sequence there is a mix_2- or a mix_0-rule. Thus $A \overset{\pi,\pi'}{\trianglelefteq} B$ in MLL.

Take a retraction $A \overset{\pi,\pi'}{\trianglelefteq} B$ in $\mathsf{MLL}^{0,2}$ with $\overset{pm}{\mapsto}$ (Lemma 7.8). If A (resp. B) has a 1- or $\perp$-subformula, then one can assume $A = 1$ (resp. $B = 1$), up to isomorphism. Indeed, thanks to $\perp \simeq 1$ and the unitality isomorphisms, one can rewrite:

$$\begin{array}{rcl|rcl|rcl} 1 \otimes C & \to & C & 1 \parr C & \to & C & \perp & \to & 1 \\ C \otimes 1 & \to & C & C \parr 1 & \to & C & & & \end{array}$$

(following what we did for distributed formulas in Section 6.2.3, this system being convergent, and its equations isomorphisms by Theorem 6.3 and Lemma 7.9). Remark π and π' are mix_0-free, as there are normal forms for $\overset{om}{\rightsquigarrow}$ and those proofs are not of the empty sequent (Lemma 7.12).

If both $A = 1$ and $B = 1$, then $1 \trianglelefteq 1$ is a retraction of MLL.

If $A = 1$ and B is unit-free, then π and π' are respectively proofs of $\vdash \perp, B$ and $\vdash B^\perp, 1$. Using $\perp$-commutations, as π is cut-free, $\pi \overset{r^*}{\mapsto} \dfrac{\overset{\phi}{\vdash B}}{\vdash \perp, B}\,(\perp)$ (the $\perp$-rule is reversible). Likewise, as the only rule below the 1-rule of π' can only be a mix_2-rule (for there is no *cut*- nor $\perp$-rules as $B^\perp$ is unit-free), using mix_2-commutations one gets $\pi' \overset{r^*}{\mapsto} \dfrac{\overset{\phi'}{\vdash B^\perp} \quad \dfrac{}{\vdash 1}\,(1)}{\vdash B^\perp, 1}\,(mix_2)$. We thus have proofs of B and $B^\perp$, without mix_0-rules by hypothesis. Contradiction with Lemma 7.13.

If A is unit-free and $B = 1$, we can reason as in the previous case up to swapping A and B, because we did not use that the composition of proofs yields an identity.

Thence, we can assume A and B to be unit-free. In $\pi \overset{B}{\bowtie} \pi'$, reach a normal form ϕ using cut-elimination and $\overset{om}{\rightsquigarrow}$ steps (Proposition 2.41). By Theorem 2.49, $\phi\ (\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*\ \mathrm{id}_A$. But, for rules of $\mathsf{MLL}^{0,2}_{uf}$, $\overset{r}{\vdash\!\dashv} \cdot \overset{\rho m}{\vdash\!\dashv} \subseteq \overset{\rho m}{\vdash\!\dashv} \cdot \overset{r}{\vdash\!\dashv}$ because a rule below a mix_0-rule can only be a mix_2-rule as there cannot be any $\perp$-rule (Lemma 7.11). This allows rewriting $\phi\ (\overset{r}{\vdash\!\dashv} \cup \overset{\rho m}{\vdash\!\dashv})^*\ \mathrm{id}_A$ into $\phi\ \overset{\rho m^*}{\vdash\!\dashv}\ \tau\ \overset{r^*}{\vdash\!\dashv}\ \mathrm{id}_A$ for some proof τ. Thence, as $\overset{r^*}{\vdash\!\dashv}$ preserves the absence of mix_0-rule, τ has no mix_0-rule. But ϕ also has none (Fact 7.10), thus both proofs are normal forms for $\overset{om}{\rightsquigarrow}$, and are equal up to $\overset{\rho m}{\vdash\!\dashv}$. By confluence of $\overset{om}{\rightsquigarrow}$, $\phi = \tau$, hence $\phi\ \overset{r^*}{\vdash\!\dashv}\ \mathrm{id}_A$. We thus found $\pi \overset{B}{\bowtie} \pi' =_\beta \mathrm{id}_A$ (Proposition 2.46), so this is also a retraction in $\mathsf{MLL}^{0,2}$; meaning the second item can be deduced from the first one. □

Remark 7.15. Contrary to MLL^2, retractions of MALL^2 are not the same as retractions without the mix_2-rule. Indeed, one can prove that in MALL $X^+ \trianglelefteq X^+ \mathbin{\&} A$ if and only if $\vdash X^-, A$ is provable (Lemma 7.64), and looking at the proof of this last lemma one also gets that in MALL^2 if $\vdash X^-, A$ is provable then $X^+ \trianglelefteq X^+ \mathbin{\&} A$. For provability with the mix_2-rule is not provability without the mix_2-rule, one easily deduces an example. It suffices for instance to take $A = X^+ \parr (Y^- \parr Y^+)$ to get that $X^+ \trianglelefteq X^+ \mathbin{\&} (X^+ \parr (Y^- \parr Y^+))$ is a retraction in MALL^2 but not in MALL.

7.2.2 Units do not matter

Our goal here is to prove an analog of Theorem 6.31 for retractions in MLL, reducing the problem to the unit-free case MLL_{uf}. To this aim, we reason as done in Section 6.2, reusing results from this section as we have taken care to write them in a generic way. Recall a $1/\perp$-pattern is the following sub-proof:

$$\dfrac{\dfrac{}{\vdash 1}\ (1)}{\vdash \perp, 1}\ (\perp)$$

Also notice that a formula of MLL is distributed (Definition 6.8) if and only if it does not have any sub-formula of the form

$$A \otimes 1 \quad 1 \otimes A \quad \perp \parr A \quad A \parr \perp$$

(where A is any formula).

Definition 7.16. Denote by $n_A(B)$ the number of A-sub-formulas in B, *i.e.* the number of occurrences of A as a sub-formula of B. We extend this notion to sequents, with $n_A(\vdash \Gamma)$ the number of A-sub-formulas in the formulas of Γ, meaning $n_A(\vdash B_1, \ldots, B_n) = \sum_{i=1}^{n} n_A(B_i)$.

Proposition 7.17. *Let A and B be two* distributed *formulas, π and π'* cut-free *proofs respectively of $\vdash A^\perp, B$ and $\vdash B^\perp, A$. Then all 1 and $\perp$-rules in π and π' belong to a $1/\perp$-pattern.*

Proof. Take a 1-rule of π, proving a sequent $\vdash 1$. There must be a rule r below it in π, for $\vdash 1$ consists of a unique formula whereas the conclusion of π has two formulas. The rule r cannot be a $\parr$-rule, again as $\vdash 1$ contains a unique formula. By distributivity, r cannot be a $\otimes$-rule too, for it would give a formula of the shape $1 \otimes -$ or $- \otimes 1$. As π is cut-free, the sole remaining possibility is r being a $\perp$-rule. Thence, every 1-rule of π belongs to a $1/\perp$-pattern, and similarly for π'.

Exactly two kinds of rules can introduce a 1-formula in π: a 1-rule and an ax-rule on $\vdash \perp, 1$. Similarly, exactly two rules can introduce a $\perp$-formula: a $\perp$-rule and an ax-rule on $\vdash \perp, 1$. Our previous analysis proves there is at least as many $\perp$-rules as 1-rules in π. As an ax-rule introduces as many $\bot$-formulas as 1-formulas, and because in MLL each formula of the conclusion sequent has exactly one rule in a proof introducing it, we deduce that $n_1(\vdash A^\bot, B) \leq n_\bot(\vdash A^\bot, B)$. By symmetry, looking at π' instead of π, $n_1(\vdash B^\bot, A) \leq n_\bot(\vdash B^\bot, A)$. However:

$$\begin{aligned} n_1(\vdash A^\bot, B) &= n_1(A^\bot) + n_1(B) = n_\bot(A) + n_1(B) \\ n_\bot(\vdash A^\bot, B) &= n_\bot(A^\bot) + n_\bot(B) = n_1(A) + n_\bot(B) \\ n_1(\vdash B^\bot, A) &= n_1(B^\bot) + n_1(A) = n_\bot(B) + n_1(A) \\ n_\bot(\vdash B^\bot, A) &= n_\bot(B^\bot) + n_\bot(A) = n_1(B) + n_\bot(A) \end{aligned}$$

Thence, $n_\bot(\vdash B^\bot, A) = n_1(\vdash A^\bot, B) \leq n_\bot(\vdash A^\bot, B) = n_1(\vdash B^\bot, A) \leq n_\bot(\vdash B^\bot, A)$. Thus, we are in the equality case, which can only hold if all (1- and) $\bot$-rules in π and π' are in a $1/\bot$-pattern, because a given $\bot$ or 1-rule cannot belong to two different $1/\bot$-patterns. $\square$

Remark 7.18. Proposition 7.17 corresponds to [BD99, Theorem 19], except that the result is stated here in sequent calculus while the authors of [BD99] worked in proof-nets with $\bot$-boxes.

Lemma 7.19. *Set A and B* distributed *formulas, and X an unsigned atom. If $A \trianglelefteq B$ then $A[X^+/1, X^-/\bot] \trianglelefteq B[X^+/1, X^-/\bot]$.*

Proof. Suppose $A \overset{\pi,\pi'}{\trianglelefteq} B$ by Lemma 7.8. Exactly two kinds of rules can introduce a 1-formula (resp. $\bot$-formula): a 1-rule (resp. $\bot$-rule) and a non-expanded ax-rule on $\vdash \bot, 1$. Using Proposition 7.17, the atomic-axiom π and π' have their $\bot$ and 1-formulas introduced by $1/\bot$-patterns. We apply Lemma 6.29: the two proofs given by this result yield $\sigma(A) \trianglelefteq \sigma(B)$, with $\sigma = [X^+/1, X^-/\bot]$. $\square$

Theorem 7.20. *Take A and B formulas of* MLL. *Set A' (resp. B') a normal form of A (resp. B) for $\mathfrak{D}$ (recall Definition 6.10 on Page 205, normalizing by Lemma 6.11). Then $A \trianglelefteq B \iff A'[X^+/1, X^-/\bot] \trianglelefteq B'[X^+/1, X^-/\bot]$, where X is a fresh atom for both A and B.*

Proof. Remark that $A \trianglelefteq B \iff A' \trianglelefteq B'$ (Corollary B.7), using that the rewriting rules of $\mathfrak{D}$ are between isomorphic formulas (Theorem 6.3).

Assume first that $A' \trianglelefteq B'$, and take X any unsigned atom. According to Lemma 7.19, $A'[X^+/1, X^-/\bot] \trianglelefteq B'[X^+/1, X^-/\bot]$.

Reciprocally, suppose $A'[X^+/1, X^-/\bot] \trianglelefteq B'[X^+/1, X^-/\bot]$ with X a fresh atom (for A and B, or equivalently for A' and B'). Then $A' \trianglelefteq B'$ follows by substituting X by 1, as X was fresh, thus $[X^+/1, X^-/\bot][1/X]$ is the identity substitution, and because substitution on an atom preserves retractions (Lemma B.6). $\square$

This last result tell us that the only retractions introduced by the units 1 and $\bot$ are their respective unitality isomorphisms for $\otimes$ and $\parr$. Therefore, we assume from now on and until the end of this chapter to be in MLL_{uf}, unless explicitly stated. In particular, this allows us to work with (canonical) proof-nets! It also means implicitly that our formulas are distributed, and our proof-nets ax-unique (Lemma 6.42).

7.2.3 Retractions in proof-nets

As for isomorphisms, a notion of retraction can be defined directly on proof-nets.

Definition 7.21. We write $A \overset{\theta,\psi}{\trianglelefteq} B$ when θ and ψ are two cut-free proof-nets respectively on $A^\perp, B$ and $B^\perp, A$ such that $\theta \overset{B}{\bowtie} \psi \xrightarrow{\beta^*} \mathrm{id}_A$.

We prove here that $A \trianglelefteq B$, $A \overset{\pi,\pi'}{\trianglelefteq} B$ and $A \overset{\theta,\theta'}{\trianglelefteq} B$ are equivalent, as was the case for isomorphisms. This particular result is true not only in MLL_{uf}, but also in MALL_{uf}.

Theorem 7.22 (Type retractions in proof-nets)**.** *Set A and B two MALL_{uf} formulas. The followings are equivalent:*

(1) $A \trianglelefteq B$

(2) there exist proofs π, π' such that $A \overset{\pi,\pi'}{\trianglelefteq} B$

(3) there exist proof-nets θ, θ' such that $A \overset{\theta,\theta'}{\trianglelefteq} B$

Proof. That Items (1) and (2) are equivalent is simply Lemma 7.8. We prove that Items (2) and (3) are equivalent by double implication.

Set π, ϕ proofs such that $A \overset{\pi,\phi}{\trianglelefteq} B$. By Lemma 4.25 and as $\mathfrak{P}_f(\mathrm{id}_A)$, $\mathfrak{P}_f(\pi)$ and $\mathfrak{P}_f(\phi)$ are cut-free, $\mathcal{B}(\mathfrak{P}_f(\pi) \overset{B}{\bowtie} \mathfrak{P}_f(\phi)) = \mathfrak{P}_f(\mathrm{id}_A)$. Thus, there exist cut-free proof-nets $\mathfrak{P}_f(\pi)$ and $\mathfrak{P}_f(\phi)$ whose composition over B yields after cut-elimination the identity proof-net of A.

Assume there exist two proof-nets θ and ψ such that $A \overset{\theta,\psi}{\trianglelefteq} B$. By Theorem 4.18, there exist atomic-axiom cut-free proofs π and ϕ such that $\mathfrak{P}_f(\pi) = \theta$ and $\mathfrak{P}_f(\phi) = \psi$ (they are cut-free by Fact 4.17, so $\mathfrak{P}$ is the singleton containing the image of $\mathfrak{P}_f$). Using Lemma 6.35, $\pi \overset{B}{\bowtie} \phi =_\beta \mathrm{id}_A$. Hence $A \overset{\pi,\phi}{\trianglelefteq} B$. $\square$

7.3 Reduction to a non-ambiguous formula

We show that in $A \trianglelefteq B$, one can assume A to be non-ambiguous. This is done thanks to half-bipartiteness of proof-nets of a retraction, leveraging results from Chapter 6.

7.3.1 Retractions are half-bipartite

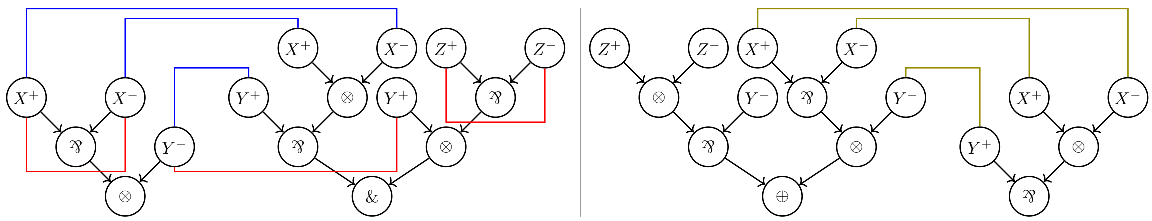
Proposition 7.23. *Assuming $A \overset{\theta,\psi}{\trianglelefteq} B$, θ (resp. ψ) is half-bipartite in $A^\perp$ (resp. A).*

Proof. We proceed by contradiction: *w.l.o.g.* there is an axiom link a in θ which is between leaves of $A^\perp$. Whence a, which does not involve leaves of B, belongs to the composition where cuts have been eliminated – there is no erasure of linkings as there is no additive. But this reduction yields the identity proof-net of A, which is bipartite by Corollary 6.45, so there cannot be an axiom link between leaves of $A^\perp$ inside: contradiction. $\square$

Remark 7.24. The extension of Proposition 7.23 to MALL_{uf} does not hold. The underlying reason is that links breaking the half-bipartite property can be erased during cut-elimination, when all linkings they belong to do not match. As a counter-example, consider the following retraction – which is a strict one, as we know the equational theory $\mathcal{L}^\dagger$ corresponds to isomorphisms in MALL_{uf}:

$$Y^+ \parr (X^+ \otimes X^-) \lhd (Y^+ \parr (X^+ \otimes X^-)) \mathbin{\&} (Y^+ \otimes (Z^+ \parr Z^-))$$

The only pair of proof-structures on the considered sequents are the following connected proof-nets, with the left one not half-bipartite in neither of its conclusions.



Also remark there is no renaming α of the left proof-net making even one of its conclusion non-ambiguous. This means that when considering a retraction $A \trianglelefteq B$ in MALL_{uf}, we cannot assume A to be non-ambiguous. Neither can one assume the proof-nets to be half-bipartite, whereas it can be done in both MLL_{uf} and ALL_{uf} (Lemma 6.43)! See also Figure 6.1 on Page 221 for another example.

Lemma 7.25. *Take θ a proof-net on A, B which is half-bipartite in A. Then $s(A) \leq s(B)$, with equality if and only if θ is bipartite.*

Proof. As θ is half-bipartite in A, as well as ax-unique (Lemma 6.42), all leaves of A are associated, by means of axiom links, to distinct leaves of B. Hence the number of leaves of A is at most the number of leaves of $B^\perp$, so $s(A) \leq s(B)$ (using Fact 1.3).

In the equality case, there cannot be a link between leaves of B, for otherwise the number of leaves of B would be strictly greater than the one of A. Therefore θ is bipartite.

Reciprocally, if θ is bipartite, then the number of leaves of A is equal to the number of leaves of B, thus these two formulas have the same size (using again Fact 1.3). ☐

Lemma 7.26. *Assuming $A \overset{\theta,\psi}{\trianglelefteq} B$, θ is bipartite if and only if ψ is.*

Proof. Remember an MLL_{uf} proof-net is always ax-unique (Lemma 6.42). We know that θ and ψ are half-bipartite respectively in $A^\perp$ and A (Proposition 7.23). Then θ bipartite if and only if $s(A^\perp) = s(B)$ (Lemma 7.25), *i.e.* $s(A) = s(B^\perp)$ (Fact 1.2), if and only if ψ is bipartite (Lemma 7.25). ☐

7.3.2 Reduction to a non-ambiguous formula

Proposition 7.27. *Take A and B some formulas and assume $A \overset{\theta,\theta'}{\trianglelefteq} B$. There exist renamings α and α', respectively of θ and θ', such that $\alpha'(A) \overset{\alpha(\theta),\alpha'(\theta')}{\trianglelefteq} \alpha(B)$, with $\alpha'(A)$ non-ambiguous. (In particular, $\alpha(A^\perp) = \alpha'(A)^\perp$ and $\alpha'(B^\perp) = \alpha(B)^\perp$.)*

Proof. This is a corollary of Theorem 6.81, not using its Item (5). The needed hypotheses come from previous results, namely Lemma 6.42 and Proposition 7.23. ☐

Lemma 7.28 (Non-ambiguous retractions)**.** *Let A and B be formulas, such that $A \overset{\theta,\theta'}{\trianglelefteq} B$. There exists an atomic-substitution σ and formulas A' and B', with A'* non-ambiguous*, such that $A = \sigma(A')$, $B = \sigma(B')$ and $A' \overset{\psi,\psi'}{\trianglelefteq} B'$ for some proof-nets ψ and ψ'.*

Proof. By Proposition 7.27, there exists renamings α and α', respectively of θ and θ', such that $\alpha(A^\perp)$ and $\alpha'(A)$ are non-ambiguous, and $\alpha'(A) \overset{\alpha(\theta),\alpha'(\theta')}{\trianglelefteq} \alpha(B)$, with $\alpha(A^\perp) = \alpha'(A)^\perp$ and $\alpha'(B^\perp) = \alpha(B)^\perp$.

Pose $A' \coloneqq \alpha'(A)$ and $B' \coloneqq \alpha(B)$, hence $A' \overset{\alpha(\theta),\alpha'(\theta')}{\trianglelefteq} B'$ and $\alpha(\theta)$ is on the sequent ${A'}^\perp, B'$. Remark A' is non-ambiguous, for $\alpha(A)$ is non-ambiguous (Lemma 6.64).

On $\alpha(\theta)$ one can define a renaming α^{-1} such that $\theta = \alpha^{-1}(\alpha(\theta))$. We use Lemma 6.68 for this, which proves that α^{-1} is a renaming, with $\alpha^{-1}(l) = m$ where m is the unique leaf of θ such that $\alpha(m) = l$.

Recall that all leaves of ${A'}^\perp$ are on distinct unsigned atoms by definition of non-ambiguous. Hence, the definition of α^{-1} on ${A'}^\perp$ can be seen as an atomic-substitution σ on atoms of ${A'}^\perp$ and of B', as follows. Take $\alpha(l)$ a leaf of ${A'}^\perp, B'$, of label a signed atom $\alpha(X)$; *w.l.o.g.* $\alpha(X)$ is a positive occurrence of an unsigned atom. The leaf l belongs to some class $\mathcal{C}$ for $B^\perp * B$ of $\theta \overset{B}{\bowtie} \theta'$. Remember that in the proof of Theorem 6.81, we defined $\alpha(X) = f(\mathcal{C})$, with f any injective function from the classes for $B^\perp * B$ of $\theta \overset{B}{\bowtie} \theta'$ to the set of positive atoms of $\mathcal{X}$. Then, α^{-1} can be seen as the atomic-substitution σ on atoms of ${A'}^\perp$ and B', defined by $\sigma(Y^+) = \alpha^{-1}(l(X))$ where X is a leaf of ${A'}^\perp, B'$ of label Y^+ (existing as by *ax*-uniqueness each negative leaf is associated to a positive leaf of dual label). This is well-defined, as two leaves of ${A'}^\perp, B'$ with the same label belong to the same class $\mathcal{C}$ by definition of α, and by injectivity of f in this definition.

Thus, $\theta = \alpha^{-1}(\alpha(\theta)) = \sigma(\alpha(\theta))$: in particular, $\sigma(B') = B$ and $\sigma({A'}^\perp) = A^\perp$, so $\sigma(A') = A$ (by Fact 1.5). Finally, A' is a non-ambiguous formula such that $A' \overset{\alpha(\theta),\alpha'(\theta')}{\trianglelefteq} B'$, $A = \sigma(A')$ and $B = \sigma(B')$. ☐

Corollary 7.29 (Reduction to non-ambiguous formulas)**.** *The set of couples of formulas A and B such that $A \trianglelefteq B$ is the set of instances, by an atomic-substitution, of couples of formulas A' and B' such that A' is* non-ambiguous *and $A' \trianglelefteq B'$. It is also the set of instances, by a substitution, of couples of formulas A' and B' such that A' is* non-ambiguous *and $A' \trianglelefteq B'$.*

Proof. By Theorem 7.22 and Lemma 7.28, given a retraction $A \trianglelefteq B$, one gets a non-ambiguous formula A and a formula B' such that $A' \trianglelefteq B'$ and A (resp. B) is an instance by an atomic substitution of A' (resp. B'). This atomic substitution can also be seen as a "standard" substitution.

Reciprocally, a substitution (thus an atomic-substitution) preserves retractions (Lemma B.6), so that given a non-ambiguous formula A' and some formula B' as well as a substitution σ, $A' \trianglelefteq B' \implies \sigma(A') \trianglelefteq \sigma(B')$. ☐

Therefore, when considering a retraction $A \trianglelefteq B$, we can always assume A to be non-ambiguous. In this case, retraction can be reduced to provability only, as was done for isomorphisms with Theorem 6.91.

Theorem 7.30 (Proof-nets for non-ambiguous retractions)**.** *Let A and B be formulas such that A is non-ambiguous. If there exist proof-nets θ and ψ respectively on the sequents $A^\perp, B$ and $B^\perp, A$, then $A \overset{\theta,\psi}{\trianglelefteq} B$.*

Proof. By Theorem 6.90, $\theta \overset{B}{\bowtie} \psi$ reduces to id_A, whence $A \overset{\theta,\psi}{\trianglelefteq} B$. □

7.4 Cantor-Bernstein-Schröeder property

In the case where one of the proof-nets of the retraction is bipartite, we can prove there is an isomorphism.

Theorem 7.31. *For A and B formulas, assume $A \overset{\theta,\theta'}{\trianglelefteq} B$, with θ and θ' proof-nets respectively on the sequents $A^\perp, B$ and $B^\perp, A$, such that θ or θ' is bipartite. Then $A \overset{\theta,\theta'}{\simeq} B$.*

Proof. By Proposition 7.27, there exists renamings α and α', respectively of θ and θ', such that $\alpha'(A) \overset{\alpha(\theta),\alpha'(\theta')}{\trianglelefteq} \alpha(B)$, with $\alpha'(A)$ non-ambiguous. As a renaming only modify leaves, $\alpha(\theta)$ or $\alpha'(\theta')$ is bipartite. In fact, using Lemma 7.26, both are. By Lemma 6.71, as $\alpha'(\theta')$ is bipartite and $\alpha'(A)$ non-ambiguous, it follows $\alpha'(B^\perp) = \alpha(B)^\perp$ is non-ambiguous, and so is $\alpha(B)$ (Lemma 6.64).

Using Theorem 6.91, $\alpha'(A) \overset{\alpha(\theta),\alpha'(\theta')}{\simeq} \alpha(B)$. But cut-elimination does not depend on labels, so $\theta \overset{A}{\bowtie} \theta'$ reduces to a proof-net whose links are those of an identity proof-net: by Lemmas 6.42 and 6.44, this proof-net and the identity proof-net of B have each exactly one linking, and their links are the same, hence they are equal. Thus, $A \overset{\theta,\theta'}{\simeq} B$. □

This last result corresponds to the strong Cantor-Bernstein-Schröeder property: proof-nets corresponding to a retraction between unit-free formulas of the same size are also proof-nets corresponding to an isomorphism between these formulas.

Proposition 7.32. *Let A and B be MLL_{uf} formulas such that $A \trianglelefteq B$. Then $s(A) \leq s(B)$, with equality if and only if $A \simeq B$. We can be even more precise: $s(B) = s(A) + 4 \times n$ for some $n \in \mathbb{N}$.*

Proof. By Theorem 7.22 and Proposition 7.23, $A \overset{\theta,\theta'}{\trianglelefteq} B$ with θ (resp. θ') half-bipartite in $A^\perp$ (resp. A). By Fact 1.2 and Lemmas 7.25 and 6.42, $s(A) = s(A^\perp)$ and the number of leaves of B is equal to the number of leaves of $A^\perp$ plus $2 \times n$ for some $n \in \mathbb{N}$: by ax-uniqueness and half-bipartiteness in $A^\perp$, each leaf of $A^\perp$ is associated to one of B by means of an axiom link; the other n links, between leaves of B, along with Fact 1.3, yield the result. In the equality case, $s(A) = s(B)$, and θ is bipartite. By Theorem 7.31, $A \overset{\theta,\theta'}{\simeq} B$ follows. Then, according to Theorem 6.36, $A \simeq B$.

Reciprocally, if $A \simeq B$, then $s(A) = s(B)$ as isomorphisms in MLL_{uf} arise from associativity and commutativity of $\otimes$ and $⅋$ (Remark 6.97). □

Remark 7.33. In Proposition 7.32, we need the unit-freeness hypothesis. This is because isomorphisms in the unit-free sub-system do not modify the size of a formula, whereas neutrality of units does. As an example, look at $X^+ \otimes (\perp ⅋ 1) \trianglelefteq X^+ \otimes (X^- ⅋ X^+)$. One can check that $X^+ \vartriangleleft X^+ \otimes (X^- ⅋ X^+)$ (this is done later, in Lemma 7.50 on Page 269). But $X^+ \simeq$

$X^+ \otimes (\bot \parr 1)$ by Theorem 6.3, and retractions can be considered up to isomorphisms (Corollary B.7), so $X^+ \otimes (\bot \parr 1) \trianglelefteq X^+ \otimes (X^- \parr X^+)$. These formulas share the same size, but are not isomorphic: $X^+ \otimes (\bot \parr 1) \not\simeq X^+ \otimes (X^- \parr X^+)$ using Theorem 6.96.

Corollary 7.34. *For A and B two MLL_{uf} formulas, if $A \triangleleft B$ then $s(A) < s(B)$.*

Proof. We apply Proposition 7.32: as $A \trianglelefteq B$ it follows $s(A) \leq s(B)$. And if $s(A) = s(B)$, then $A \simeq B$, a contradiction. □

The result on Proposition 7.32 is a really strong one: we have on one side strict retractions $A \triangleleft B$ with $s(A) < s(B)$, and on the other side isomorphisms $A \simeq B$. As a corollary, we know for instance the followings cannot be retractions, as they are not isomorphisms (according to Theorem 6.96):

$$X \otimes Y \ntrianglelefteq X \parr Y$$
$$X \otimes (Y \parr Z) \ntrianglelefteq Y \parr (X \otimes Z)$$

Therefore, the problem of finding retractions in MLL reduces to finding strict retractions $A \triangleleft B$ in MLL_{uf}, with $s(A) < s(B)$ (remember that units have been removed in Section 7.2). We can also deduce the Cantor-Bernstein-Schröeder property for MLL from the previous result.

Lemma 7.35 (Cantor-Bernstein-Schröeder for MLL_{uf})**.** *Let A and B be MLL_{uf} formulas such that $A \trianglelefteq B$ and $B \trianglelefteq A$. Then $A \simeq B$.*

Proof. By Proposition 7.32, $s(A) \leq s(B)$ and $s(B) \leq s(A)$. Whence we are in the equality case, and $A \simeq B$. □

Theorem 7.36 (Cantor-Bernstein-Schröeder for MLL)**.** *Let A and B be MLL formulas such that $A \trianglelefteq B$ and $B \trianglelefteq A$. Then $A \simeq B$.*

Proof. Let A_1 (resp. B_1) be a normal form of A (resp. B) for $\mathfrak{D}$. Set $A_2 = A_1[X^+/1, X^-/\bot]$ and $B_2 = B_1[X^+/1, X^-/\bot]$, for a fresh atom X. By Theorem 7.20, $A_2 \trianglelefteq B_2$ and $B_2 \trianglelefteq A_2$. But A_2 and B_2 are unit-free and therefore, by Lemma 7.35, $A_2 \simeq B_2$. We obtain $A \simeq B$ by Theorem 7.20. □

7.5 General results

We give here some general results about retractions in MLL_{uf}. The main idea from this section and the next one is to find particular sub-graphs in proof-nets of a retraction, for instance a shape (nearly) correspond to a composition with a Beffara retraction. Thence, removing one of these sub-graphs allows to conclude by induction that a retraction is made of the "basic" retractions whose shape we found. We prove here results on a shape corresponding to the Beffara retraction, even if we can only prove such a shape appears in retractions to an atom (in the next section).

Recall that, when $A \trianglelefteq B$, we can assume A (but not B) to be non-ambiguous, with each atom occurring positively – no negated atom in A. Also recall cut-elimination is confluent and strongly normalizing (Theorem 4.12).

Given a formula A, set $\mathcal{V}(A)$ the set of unsigned atoms occurring in A. For instance, one has $\mathcal{V}((X^+ \parr Y^-) \otimes (X^- \parr X^+)) = \{X; Y\}$.

Lemma 7.37. *If $A \trianglelefteq B$ then $\mathcal{V}(A) = \mathcal{V}(B)$.*

Proof. By Theorem 7.22, we have proof-nets θ and ψ such that $A \overset{\theta,\psi}{\trianglelefteq} B$. According to Lemma 6.79, each class for $B * B^\perp$ of $\theta \overset{B}{\bowtie} \psi$ is a simple path in GOI projection graph. Thence, it contains two leaves of $A^\perp$ or A. As θ and ψ are half-bipartite respectively in $A^\perp$ and A, each class cannot be reduced to two leaves of $A^\perp$ or A, *i.e.* it must contain a leaf in B or $B^\perp$. Thus, a class contains at least one leaf of $A^\perp$ or A, and at least one of B or $B^\perp$.

Furthermore, a class contains leaves on the same unsigned atom (Lemma 6.77). Consider an unsigned atom X, assume it appears in A. Then its class contains a leaf of $B^\perp$ or B, hence X also appears in B. This gives $\mathcal{V}(A) \subseteq \mathcal{V}(B)$, and the same reasoning applied on an atom of B gives the other inclusion. □

We have a general necessary condition on B when it is a retraction to any formula.

Lemma 7.38. *If $A \trianglelefteq B$ with A and B MLL_{uf} formulas, and A non-ambiguous, then for any leaf X, for any sub-formula B' of B (*i.e. *a sub-tree), the number $n_X(B')$ of occurrences of X in B' – counting separately its positive and negated versions – differ from $n_{X^\perp}(B')$, the one of $X^\perp$, by at most one. In other words, $|n_X(B') - n_{X^\perp}(B')| \leq 1$.*

Proof. Take a sub-formula B' of B. By symmetry, we prove only $n_X(B') \leq n_{X^\perp}(B') + 1$.

We have proof-nets θ and ψ such that $A \overset{\theta,\psi}{\trianglelefteq} B$ (Theorem 7.22). In $\theta \overset{B}{\bowtie} \psi$, we reduce all cuts except the one of root B'. We obtain a proof-net on the cut sequent $[B' * B'^\perp]\ A^\perp, A$.

The leaves labeled X in B' can only be linked to leaves labeled $X^\perp$ in B' or to the unique $X^\perp$ of A or $A^\perp$. In particular, they cannot be linked to any $X^\perp$ in $B'^\perp$ by Lemma 4.53. Therefore, in this proof-net, axiom links must link $n_X(B')$ leaves labeled X to at most $n_{X^\perp}(B') + 1$ leaves labeled $X^\perp$, with all links being disjoint. This implies $n_X(B') \leq n_{X^\perp}(B') + 1$. □

The condition in Lemma 7.38 is necessary, but not sufficient. For instance, consider $A = X$ and $B = X \otimes (X^\perp \otimes X)$. Then B indeed respects the conclusion of Lemma 7.38, but we do not have $X \trianglelefteq X \otimes (X^\perp \otimes X)$, simply because $\vdash X^\perp, X \otimes (X^\perp \otimes X)$ is not provable. Also, the non-ambiguity hypothesis is necessary: with $A = X^+ \otimes X^+$ and $B = X^+ \otimes (X^- \parr (X^+ \otimes X^+))$, one has $A \triangleleft B$ (see Section 7.7) but A is a sub-formula of B which does not respect the conclusion of Lemma 7.38.

We say B is a **sub-formula up to isomorphism** of A if there exists a formula A' such that $A \simeq A'$ and B is a sub-formula of A'.

Corollary 7.39. *If $A \trianglelefteq B$ with A non-ambiguous, then for any signed atom X, neither $X \parr X$ nor $X \otimes X$ is a sub-formula up to isomorphism of B.*

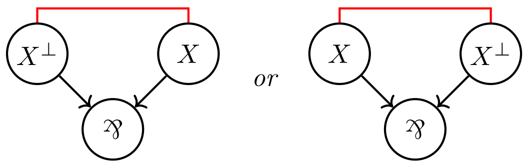
Proof. By contradiction, say $X \parr X$ is a sub-formula of B' for $B \simeq B'$. Then $A \trianglelefteq B'$ by Corollary B.7, and we contradict Lemma 7.38:

$$n_X(X \parr X) - n_{X^\perp}(X \parr X) = 2 - 0 = 2 > 1$$

The result on $X \otimes X$ follows the same way. □

We now aim to prove that retractions contain special sub-graphs. The sub-graph we are looking for is an axiom link between the two premises of a $\parr$-vertex. Proceding by contradiction, we look at what a proof-net without such a sub-graph look like. As a short-hand, when a vertex v has for child a vertex u in the syntactic tree of a formula, we say v is **above** u.

Lemma 7.40. *Take $A \overset{\theta,\theta'}{\lhd} B$ and X an atom of A, such that the connective just below X in A is not a $\parr$ – thus, either such a connective does not exist, meaning $A = X$, or it is a $\otimes$. Assume neither θ nor θ' contains a sub-graph of the following shape:*



Then every X (resp. $X^\perp$) of B and $B^\perp$ is above a $\otimes$ (resp. $\parr$).

Proof. We follow the construction of the class of X for $B^\perp * B$ (Definition 6.75), starting from this leaf X, and prove that, while not seeing a forbidden sub-graph, all visited leaves have the wished property. This will be enough as this class contains all leaves of the proof-net, thanks to Lemma 6.79 (because there are only two leaves outside of $B^\perp * B$). Remark this class can be constructed the following way:

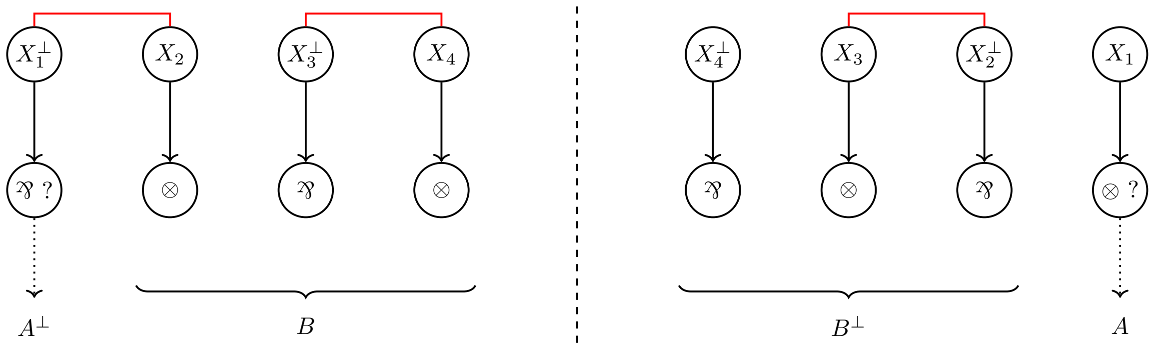
1. we start from X;

2. with l the last leaf added, we add the unique leaf l' such that there is an axiom link $l — l'$ in $\theta \overset{B}{\bowtie} \theta'$;

3. If l' belongs to A, we stop here; otherwise, l' belongs either to B or $B^\perp$, and we add $l'^\perp$, belonging respectively to $B^\perp$ or B. We then go back to Item 2.

This is true vacuously at the beginning of the construction for no leaf of B nor of $B^\perp$ is crossed. Assume the result holds up to some point in the construction. We are considering a leaf l labeled $X^\perp$, which is either in $A^\perp$ (in the first step) or in B or $B^\perp$ (in the following steps). By hypothesis and induction hypothesis respectively, l is above no connective or above a $\parr$. It is linked to another leaf l', labeled X. If l' is in A, we are done. Otherwise, it is in B or $B^\perp$, so above either a $\parr$ or a $\otimes$ – notice we use here that B is not a single leaf, which is true because of the strict retraction $A \lhd B$, thus $s(B) > s(A) \geq 1$ using Corollary 7.34. If it were above a $\parr$, then we would contradict either Lemma 4.54 (if l and l' are not above a same $\parr$) or the hypothesis on sub-graphs (if l and l' are above a same $\parr$). Therefore, l' must be above a $\otimes$. Thus, its dual $l'^\perp$, labeled $X^\perp$, is above a $\parr$, which concludes the inductive case.

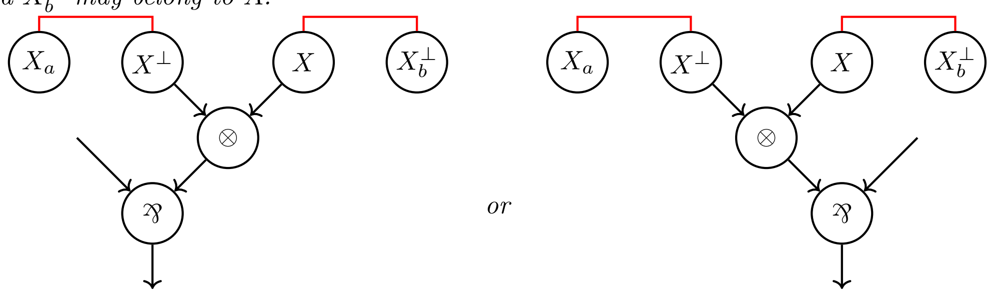
Here are some drawings and additional explanations to give intuitions. We start from $X_1^\perp$ in $A^\perp$, which is above a $\parr$ or nothing. It is linked to some X_2, necessarily in B, which must be above a $\otimes$ (Lemma 4.54 and $s(B) > 1$). We then look at the dual $X_2^\perp$ in $B^\perp$, which must be above a $\parr$ by duality. It is linked to some X: if it is X_1 in A then we are done, otherwise it is a non-previously seen X_3 in $B^\perp$. This leaf X_3 cannot be above the $\parr$ below $X_2^\perp$ by hypothesis on sub-graphs, nor above a different $\parr$ by Lemma 4.54. Therefore, X_3 is above a $\otimes$. We then keep going, with $X_3^\perp$ above a $\parr$ in B, which can only be linked to some X_4 which is above a $\otimes$ by the previous reasoning, …



On the previous drawing, notice that some ⊗-vertices (resp. ⅋-vertices) drawn separately may in fact represent the same vertex. □

Remark 7.41. In the proof of Lemma 7.40, we used connectivity a lot. Nonetheless, retractions with and without the mix_2-rules should be the same according to the work done in Section 7.2.1, thus it is surprising we have to use it here. Still, I did not manage to find a proof not using connectivity.

Lemma 7.42. *Take θ a proof-net whose cut sequent has a sub-formula of the shape $(X^\perp \otimes X) \parr A$ or $A \parr (X^\perp \otimes X)$ (*i.e.* has $(X^\perp \otimes X) \parr A$ as a sub-formula up to isomorphism), where X is a signed atom and A a formula. Then, on this sub-formula, θ has the following shape, where signed atoms X_a and $X_b^\perp$ may belong to A:*

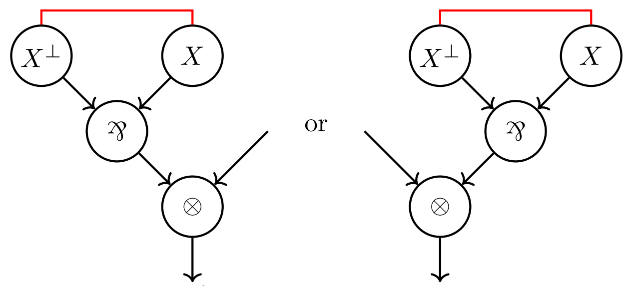


Proof. As the formula tree is given by the cut sequent θ is on, it suffices to prove the axiom links are on the required shapes. The only way for these links not to be of this form would be for there to be a link $X^\perp — X$. But this would lead to a switching cycle, forbidden by the correctness criterion (P2): the cycle passing through this link, going to the ⊗-vertex using its right premise, and going back to X. □

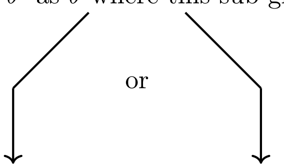
We can now proceed with our main idea: removing a sub-graph corresponding to a retraction, while still keeping the resulting proof-nets associated to a retraction. As the sub-graphs we study are not exactly those of the Beffara retraction, but are closely alike, we call our rewriting procedure Quasi-Beffara.

Definition 7.43 (Quasi-Beffara on proof-structures)**.** We define the **Quasi-Beffara** rewriting, noted $\xrightarrow{\text{qBeffara}}$, as the union of two transformations $\xrightarrow{\text{qBeffara}_1}$ and $\xrightarrow{\text{qBeffara}_2}$ on proof-structures as follows.

Consider a *cut-free* proof-structure θ containing a sub-graph of the shape:

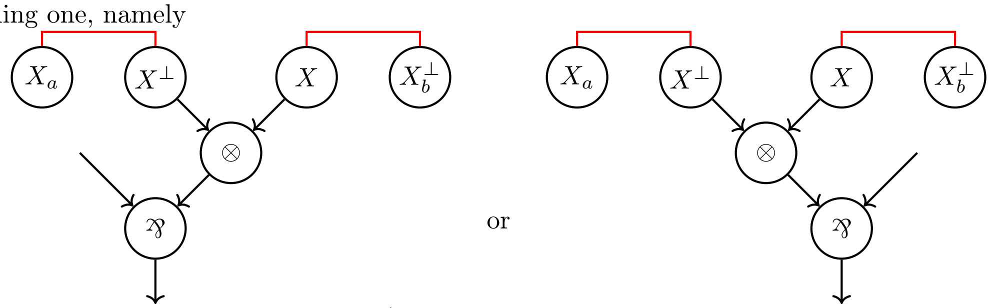


where X is a signed atom. We define θ' as θ where this sub-graph is replaced respectively by

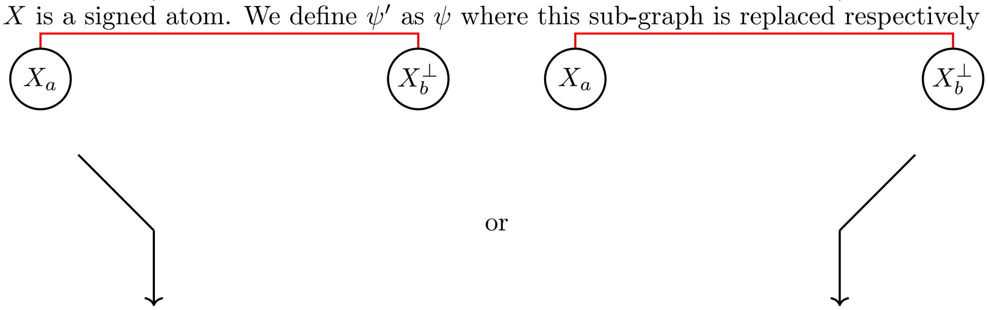


and call $\theta \xrightarrow{\text{qBeffara}_1} \theta'$ this rewriting rule.

Similarly, take a cut-free proof-structure ψ containing a sub-graph of the dual shape of the preceding one, namely



where X is a signed atom. We define ψ' as ψ where this sub-graph is replaced respectively by



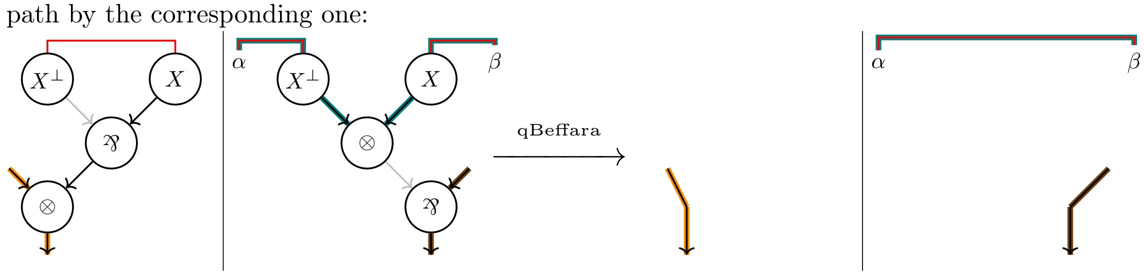
and call $\psi \xrightarrow{\text{qBeffara}_2} \psi'$ this rewriting rule.

Lemma 7.44. *Consider a proof-net θ and a rewriting step $\theta \xrightarrow{\text{qBeffara}} \theta'$. Then θ' is a proof-net, with the same number of connected components $\#cc$ as θ.*

Proof. It is easy to check that if θ is a proof-structure with $\theta \xrightarrow{\text{qBeffara}} \theta'$, then θ' is a proof-structure too. Remember a MLL proof-structure is composed of a single linking, whose additive resolution is the full sequent (Lemma 6.42). The considered rewriting rule preserves this property, hence what we get after is still a linking on the resulting cut sequent. That (P0) and (P1) are preserved is trivial. Thus, we only need to prove (P2) as well as the result about connected components –

remember an MLL proof-net trivially respects (P3) as it is made of a single linking (Lemma 6.42).

Preservation of (P2) follows as we only remove paths. More precisely, one can prove that a switching cycle in θ' would yield a switching cycle in θ, a contradiction. For this, transport a path from θ' to θ in the trivial way outside of the rewritten zone, while inside it one replaces each colored path by the corresponding one:



The number $\#cc$ of connective components is preserved as we remove 4 vertices including one $\parr$-vertex, and 5 edges, according to Lemma 4.52. □

Definition 7.45 (Quasi-Beffara on formulas)**.** Notice that Quasi-Beffara modifies the cut sequent of a proof-net it is applied on. Therefore, we extend these rewriting rules to (sub-)formulas through:

$$A \otimes (X^\perp \parr X) \xrightarrow{\text{qBeffara}_1} A \qquad (X^\perp \parr X) \otimes A \xrightarrow{\text{qBeffara}_1} A$$
$$(X^\perp \otimes X) \parr A \xrightarrow{\text{qBeffara}_2} A \qquad A \parr (X^\perp \otimes X) \xrightarrow{\text{qBeffara}_2} A$$

By abuse of notations, we still denote $\xrightarrow{\text{qBeffara}}$ this rewriting on formulas.

Lemma 7.46. *Consider a rewriting step $\theta \xrightarrow{\text{qBeffara}_i} \theta'$, for $i \in \{0;1\}$. Then the cut sequents of θ and θ' are the same, except a $\xrightarrow{\text{qBeffara}_i}$ has been applied to a formula of the sequent of θ to get the one of θ'.*

Proof. This follows from Definitions 7.43 and 7.45. □

We can now extend Quasi-Beffara to retractions.

Lemma 7.47. *Consider a retraction $A \overset{\theta,\psi}{\trianglelefteq} B$ where a $\xrightarrow{\text{qBeffara}_1}$ rule can be applied on θ (resp. ψ), yielding θ' on $\vdash A^\perp, B'$ (resp. ψ' on $\vdash {B'}^\perp, A$) with $B \xrightarrow{\text{qBeffara}} B'$. Then, a $\xrightarrow{\text{qBeffara}_2}$ step can be applied on the corresponding dual sub-graph of ψ (resp. θ) by, giving ψ' on $\vdash {B'}^\perp, A$ (resp. θ' on $\vdash A^\perp, B'$). Furthermore, $A \overset{\theta',\psi'}{\trianglelefteq} B'$.*

Proof. Using Proposition 7.23, the $\xrightarrow{\text{qBeffara}_1}$ step can only be applied on a sub-graph using the syntactic tree of B (resp. $B^\perp$). That a $\xrightarrow{\text{qBeffara}_2}$ step can be applied on the corresponding dual sub-graph of the other proof-net follows by Lemma 7.42.

It remains to prove the obtained graphs are proof-nets of a retraction. They are proof-nets by Lemma 7.44, on the wished cut sequents. Finally, the normal form of $\theta' \overset{B'}{\bowtie} \psi'$ is the same as the normal form of $\theta \overset{B}{\bowtie} \psi$: we can reduce cuts in $\theta \overset{B}{\bowtie} \psi$ following what we would do in $\theta' \overset{B'}{\bowtie} \psi'$, except for the removed vertices. But reducing a cut on a removed vertex as soon as it appears, doing as soon as possible the $\parr - \otimes$ case, followed by the *ax* one, permits to reduce cuts in the same way in both compositions. The results of eliminating cuts in both proof-nets are one and the same, using Theorem 4.12 and that as soon as the cut-elimination as been applied in the removed vertices of $\theta \overset{B}{\bowtie} \psi$, then the reduced proof-nets obtained in both cases are equal. ☐

Definition 7.48 (Quasi-Beffara on retractions)**.** Consider a retraction $A \overset{\theta,\psi}{\trianglelefteq} B$ where a $\xrightarrow{\text{qBeffara}_1}$ rule can be applied on θ (resp. ψ), yielding θ' on $\vdash A^\perp, B'$ (resp. ψ' on $\vdash B'^\perp, A$) with $B \xrightarrow{\text{qBeffara}} B'$. Using Lemma 7.47, a corresponding $\xrightarrow{\text{qBeffara}_2}$ step can be applied on the other proof-net $\psi \xrightarrow{\text{qBeffara}_2} \psi'$ (resp. $\theta \xrightarrow{\text{qBeffara}_2} \theta'$), yielding a retraction $A \overset{\theta',\psi'}{\trianglelefteq} B'$. We denote in this case $A \overset{\theta,\psi}{\trianglelefteq} B \xrightarrow{\text{qBeffara}} A \overset{\theta',\psi'}{\trianglelefteq} B'$.

7.6 Retractions to an atom

We now restrict ourselves to retractions of the shape $X \trianglelefteq B$ for X a signed atom. In this restricted setting, we can characterize all retractions.

Let us remark that $X \trianglelefteq B$ if and only if there are proofs of $\vdash X^\perp, B$ and of $\vdash B^\perp, X$ (Theorem 7.30, with X non-ambiguous). The retraction $X \trianglelefteq B$ is the retraction on one atom, as we can rename to have the retract formula (on the left of $\trianglelefteq$) non-ambiguous, thus for instance deducing retractions to $X \otimes X$ from those to $X \otimes Y$. Furthermore, retractions B of X corresponds to universal super-types: if we know nothing about a type X, here are the types B such that X is a sub-type of B.

7.6.1 Beffara generates all retractions to an atom

In all this section, X is a fixed signed atom.

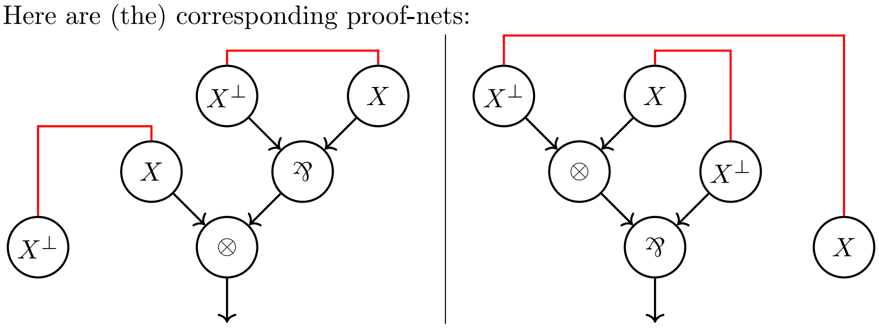
Definition 7.49 (Beffara)**.** The Beffara's type retraction is $X \lhd X \otimes (X^\perp \parr X)$.

This is indeed a strict retraction.

Lemma 7.50.

$$X \lhd X \otimes (X^\perp \parr X)$$

Proof. That this retraction is strict follows from Proposition 7.32, once we have proved it is a retraction. Here are (the) corresponding proof-nets:



One could check they indeed are connected proof-nets, and that their composition on $X \otimes (X^\perp \parr X)$ reduces to the identity proof-net – even if this last step is not necessary according to Theorem 7.30. $\square$

Remark 7.51. Another way of writing the Beffara retraction is the following:

$$X \lhd (X \multimap X) \multimap X$$

We want to prove that any type retraction of the shape $X \trianglelefteq A$ is generated by a (finite) sequence of Beffara's type retractions.

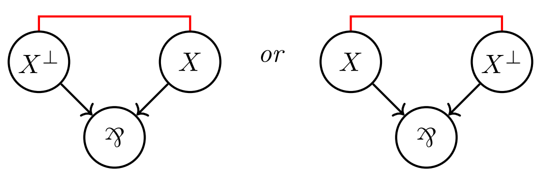
Theorem 7.52. *Take A a formula of* MLL. *Then $X \trianglelefteq A$ if and only if A is obtained from X by a sequence of isomorphisms and Beffara's retractions.*

To prove the above theorem, most of the work has been done in the previous section. The lacking result is finding our sub-graph corresponding to an axiom above a $\parr$, which we manage to when looking at retractions to an atom.

Corollary 7.53. *If $X \lhd B$, then $X \parr X^\perp$ or $X \otimes X^\perp$ is a sub-formula up to isomorphism of B.*

Proof. As $X \not\simeq B$, we have $B \neq X$. Moreover, $\mathcal{V}(B) = \{X\}$ (Lemma 7.37). Take a "highest" connective in B, meaning one whose both premises are leaves. Such a connective exists as B cannot be reduced to a signed atom, so has at least one connective $\parr$ or $\otimes$. By Corollary 7.39, a "highest" connective cannot be of the shape $X \parr X$, $X \otimes X$, $X^\perp \parr X^\perp$ nor $X^\perp \otimes X^\perp$. It must therefore be of the shape $X \parr X^\perp$, $X \otimes X^\perp$, $X^\perp \parr X$ nor $X^\perp \otimes X$, so $X \parr X^\perp$ or $X \otimes X^\perp$ up to isomorphism. $\square$

Corollary 7.54. *If $X \overset{\theta,\theta'}{\lhd} B$, then θ or θ' contain a sub-graph of the following shape:*



Proof. By contradiction, if none of the two contains such a sub-graph, then by Lemma 7.40 every X (resp. $X^\perp$) of B and $B^\perp$ is above a $\otimes$ (resp. $\parr$). This cannot be according to Corollary 7.53. $\square$

Remark 7.55. Another possible proof of Corollary 7.54, not using Corollary 7.53, is the following. Still by contradiction, if none of the two proof-nets contains such a sub-graph, then by Lemma 7.40 every X (resp. $X^\perp$) of B and $B^\perp$ is above a $\otimes$ (resp. $\parr$). Then X in $A = X$, which is above nothing, is linked to some $X^\perp$ in $B^\perp$, which is thus above a $\parr$: contradiction with Lemma 4.54.

Proposition 7.56. *Assume $X \overset{\theta,\psi}{\trianglelefteq} B$. Then $X \overset{\theta,\psi}{\trianglelefteq} B \xrightarrow{\text{qBeffara}^*} X \overset{\mathrm{id}_X,\mathrm{id}_X}{\trianglelefteq} X$. In particular, if $X \trianglelefteq B$, then $B \xrightarrow{\text{qBeffara}^*} X$.*

Proof. We reason by strong recurrence on $s(B)$.

If $s(B) = 1$, then B is an atom Y such that $\vdash X^\perp, Y$ is provable. This ensures $B = Y = X$.

If $s(B) > 1$, then $X \lhd B$ because isomorphisms of MLL_{uf} preserve the size of formulas (Theorem 6.96). Remark we cannot have $B = X \parr X^\perp$ or dually $B = X^\perp \otimes X$, up to isomorphism, because the sizes do not match using Proposition 7.32. Using Corollary 7.54 followed by Lemma 4.55 (with $s(B) \geq 5$), a $\xrightarrow{\text{qBeffara}_1}$ step can be applied on θ or ψ. Therefore, by Definition 7.48, $A \overset{\theta,\psi}{\trianglelefteq} B \xrightarrow{\text{qBeffara}} A \overset{\theta',\psi'}{\trianglelefteq} B'$. In particular, $s(B') = s(B) - 4$. By induction hypothesis, $X \overset{\theta',\psi'}{\trianglelefteq} B' \xrightarrow{\text{qBeffara}^*} X \overset{\mathrm{id}_X,\mathrm{id}_X}{\trianglelefteq} X$, allowing us to conclude. □

We thus have proved that all (proofs of) retractions are obtained through a succession of Quasi-Beffara steps. However, remember we wanted Beffara steps. We will bridge the gap between Quasi-Beffara and Beffara by giving a grammar characterizing formulas B such that $X \trianglelefteq B$.

Definition 7.57. We define the two following grammars, where X is a fixed signed atom.

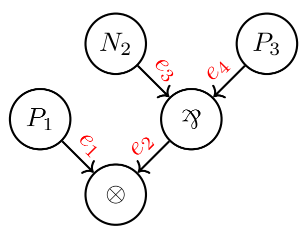
$$\begin{aligned} P &::= X \quad \mid P \otimes (N \parr P) \mid P \parr (N \otimes P) \\ N &::= X^\perp \mid N \otimes (P \parr N) \mid N \parr (P \otimes N) \end{aligned}$$

Lemma 7.58. *If $A \xrightarrow{\text{qBeffara}} B$ and $B \in P$ (resp. N) then* up to isomorphism *$A \in P$ (resp. N).*

Proof. By duality, we only prove it for P with $B = X$ or $B = P_1 \otimes (N_2 \parr P_3)$ – we can do so because the grammar N contains the dual of the formulas generated by P, that the $P \parr (N \otimes P)$ will be symmetric to this one, and $\xrightarrow{\text{qBeffara}}$ is "closed under dual", namely that applying $\xrightarrow{\text{qBeffara}_1}$ then taking the dual is the same as taking the dual then applying $\xrightarrow{\text{qBeffara}_2}$, and vice-versa. By induction on the grammar of P, we consider only the case where the Quasi-Beffara step is applied in the "pattern" of B, with as final edge e.

If $B = X$, then A is, up to isomorphism, either $X \otimes (X^\perp \parr X)$ or $X \parr (X^\perp \otimes X)$. In both cases, $A \in P$ up to isomorphism.

Otherwise, $B = P_1 \otimes (N_2 \parr P_3)$. Call e_1 the edge between (the root of) P_1 and the $\otimes$, e_2 the one between the $\parr$ and the $\otimes$, e_3 the one from N_2 to the $\parr$ and e_4 the one between P_3 and the $\parr$. Graphically:



By hypothesis, $e \in \{e_1; e_2; e_3; e_4\}$. Notice that for any C in P (resp. N), $C \otimes (X^\perp \parr X)$ and $C \parr (X^\perp \otimes X)$ belong to P (resp. N). Thus, if $e \in \{e_1; e_3; e_4\}$, then A is in P as $A = P_1' \otimes (N_2' \parr P_3')$ by the previous argument (with two formulas among P_1', N_2' and P_3' equal to their non prime versions).

Finally, in case $e = e_2$, then up to isomorphism $e = e_1$ (in case of a $\xrightarrow{\text{qBeffara}_1}$ step) or $e = e_4$ (in case of a $\xrightarrow{\text{qBeffara}_2}$ step). □

Proposition 7.59. *If $B \xrightarrow{\text{qBeffara}^*} X$, then $B \in P$* up to isomorphism. *If $B \xrightarrow{\text{qBeffara}^*} X^\perp$, then $B \in N$* up to isomorphism.

Proof. This follows from $X \in P$, $X^\perp \in N$ and Lemma 7.58. □

Definition 7.60. Set $\xrightarrow{\text{Beffara}}$ the union of the following rewriting rules on (sub-)formulas, where X is a signed atom:

- $X \otimes (X^\perp \parr X) \xrightarrow{\text{Beffara}_1} X$
- $(X^\perp \parr X) \otimes X \xrightarrow{\text{Beffara}_1} X$
- $(X^\perp \otimes X) \parr X \xrightarrow{\text{Beffara}_2} X$
- $X \parr (X^\perp \otimes X) \xrightarrow{\text{Beffara}_2} X$

Similarly, we set $\xrightarrow[\simeq]{\text{Beffara}}$ the previous rewriting rules applied on (sub-)formulas *up to isomorphism.*

Lemma 7.61. *If $B \in P$, then $B \xrightarrow{\text{Beffara}^*} X$. If $B \in N$, then $B \xrightarrow{\text{Beffara}^*} X^\perp$.*

In particular, if $B \in P$ (resp. N) up to isomorphism, then $B \xrightarrow[\simeq]{\text{Beffara}^*} X$ (resp. $X^\perp$).

Proof. We prove the wished property by induction on the grammars P and N. By duality, we show only the result for P.

If $B = X$, then the result holds trivially.

If $B = P_1 \otimes (N_2 \parr P_3)$, then by induction hypothesis:

$$P_1 \xrightarrow{\text{Beffara}^*} X$$
$$N_2 \xrightarrow{\text{Beffara}^*} X^\perp$$
$$P_3 \xrightarrow{\text{Beffara}^*} X$$

Therefore, $B \xrightarrow{\text{Beffara}^*} X \otimes (X^\perp \parr X) \xrightarrow{\text{Beffara}_1} X$.

If $B = P_1 \parr (N_2 \otimes P_3)$, then by induction hypothesis:

$$P_1 \xrightarrow{\text{Beffara}^*} X$$
$$N_2 \xrightarrow{\text{Beffara}^*} X^\perp$$
$$P_3 \xrightarrow{\text{Beffara}^*} X$$

Therefore, $B \xrightarrow{\text{Beffara}^*} X \parr (X^\perp \otimes X) \xrightarrow{\text{Beffara}_2} X$. ☐

Lemma 7.62. *If $B \xrightarrow{\text{Beffara}} A$, then $A \lhd B$.*

Proof. Because the Beffara retraction is a strict retraction (Lemma 7.50), and by contextuality (Lemma B.4). ☐

Our main result on retractions to an atom then follows.

Theorem 7.63. *The followings are equivalent, where X is a signed atom and B an MLL_{uf} formula:*

1. $X \trianglelefteq B$

2. $B \xrightarrow{\text{qBeffara}^*} X$

3. $B \in P$ *up to isomorphism*

4. $B \xrightarrow[\simeq]{\text{Beffara}^*} X$

Proof. Using Proposition 7.56, it follows Item 1 implies Item 2. By Proposition 7.59, Item 2 implies Item 3. Through Lemma 7.61, Item 3 implies Item 4. Finally, Item 4 implies Item 1 according to Lemma 7.62 and because retractions can be considered up to isomorphisms (Corollary B.7). ☐

From our main theorem, one can deduce the result claimed at the beginning of this section, namely that if $X \trianglelefteq A$ then A can be obtained from X by a sequence of Beffara's retractions and isomorphisms.

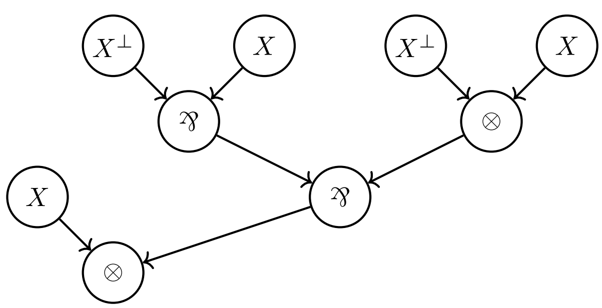
Proof of Theorem 7.52. Up to isomorphism, one can assume A to be distributed (Theorem 7.20), and then unit-free (Theorem 7.20). Then one simply use that Item 1 and Item 4 are equivalent in Theorem 7.63. ☐

7.6.2 Subtleties on the result

We give here some remarks on Theorem 7.63, explaining why we stated it in terms of formulas and not in terms of proof(-net)s, *i.e.* why it stands for type retractions and not the retractions themselves.

7.6.2.1 Beffara generates all retracts to an atom only up to isomorphism

A first remark is that the "up to isomorphisms", whether in Item 3 or in Item 4, are necessary. For instance, consider $X \lhd X \otimes ((X^\perp \parr X) \parr (X^\perp \otimes X))$ – which is indeed a strict retraction by Proposition 7.32 and Theorem 7.63. However, the retracting formula has the following syntactic tree:

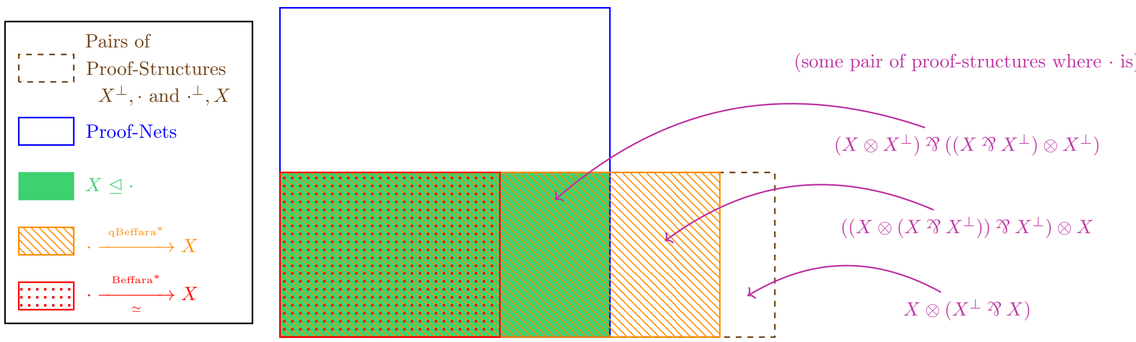


It is generated by Beffara only up to isomorphism, *i.e.* by $\cdot \xrightarrow[\simeq]{\text{Beffara}} X$ but not by $\cdot \xrightarrow{\text{Beffara}} X$. Indeed, it has no sub-formula of the shape $X\otimes(X^\perp \parr X)$ nor $X \parr (X^\perp \otimes X)$, even up to commutativity of the connectives. We need associativity of the $\parr$, which yields:

$$X \otimes ((X^\perp \parr X) \parr (X^\perp \otimes X)) \simeq X \otimes (X^\perp \parr (X \parr (X^\perp \otimes X)))$$

A Beffara rewriting can then be applied. Similarly, $X \otimes ((X^\perp \parr X) \parr (X^\perp \otimes X)) \notin P$, because this would imply $(X^\perp \parr X)$ in P or N, which is not. However, $X \otimes (X^\perp \parr (X \parr (X^\perp \otimes X))) \in P$.

Notice that our main result, Theorem 7.63, holds only on type retractions, *i.e. on formulas but not on proofs*. As we will see, we are in the situation depicted below:



With a thousand words instead of a picture:

- Beffara generates all formulas retracting to an atom, but not all proof-nets that these formulas are retracting to an atom, even up to isomorphism;

- Quasi-Beffara does generate all formulas retracting to an atom, as well as all proof-nets that these formulas are retracting to an atom (not using any isomorphism), but it also generates incorrect proof-structures which do not respect (P2) but compose to the identity proof-net of the considered atom;

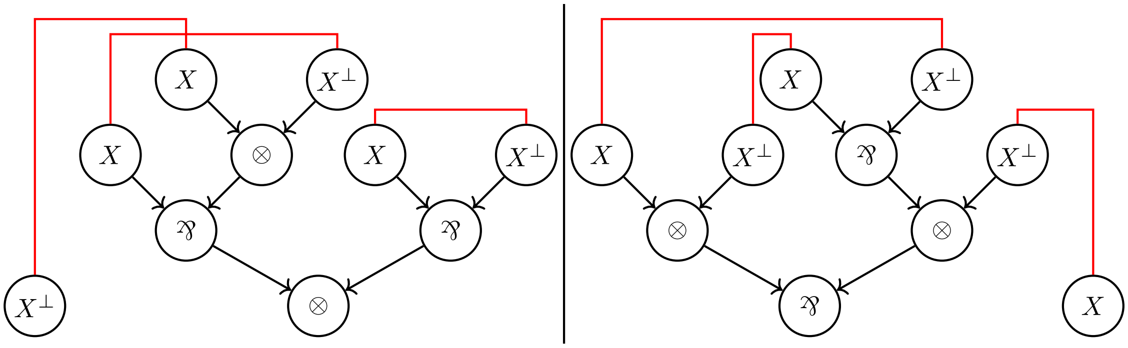
- nevertheless, Quasi-Beffara does not generate all proof-structures respectively on the sequents $X^\perp, B$ and $B^\perp, X$, with B a retracting of X, whose composition yields the identity proof-net of X.

As the two transformations Beffara and Quasi-Beffara are really close to one another – these transformations on proof-structures differ only by the presence of an X in Beffara that is not in Quasi-Beffara – it seems hard to get an intermediate transformation generating exactly proof-nets of retractions to an atom. The only way I have found to get exactly these proof-nets is to generate all proof-structures given by Quasi-Beffara, and then to forget the pairs where one of the proof-structures is not correct.

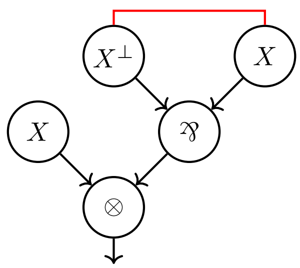
The following sections give examples for the above claims.

7.6.2.2 Proof-nets of a retraction not generated by Beffara

The following pair of proofs of $X \lhd (X \otimes X^\perp) \parr ((X \parr X^\perp) \otimes X^\perp)$ is not generated by Beffara, even up to isomorphism.

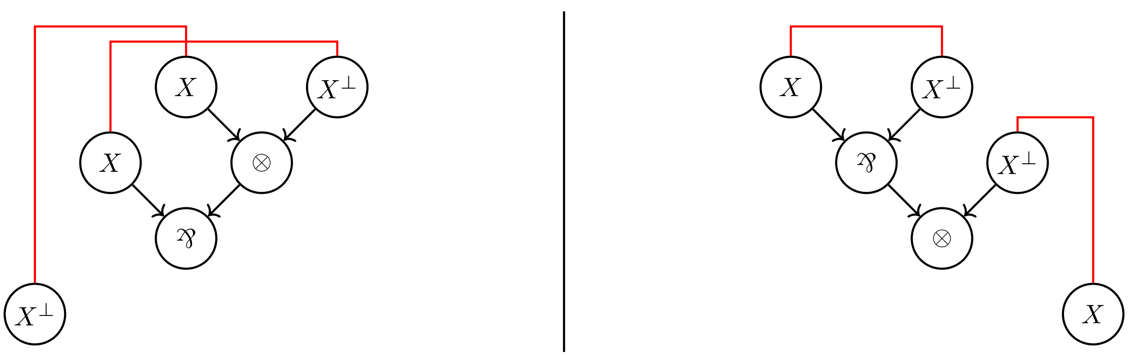


Indeed, both of these proof-nets do not contain the following sub-graph, present in any result of applying a Beffara reverted step:



The only (non-reflexive) isomorphisms that can be applied are commutativity of the connectives, and they cannot generate this shape.

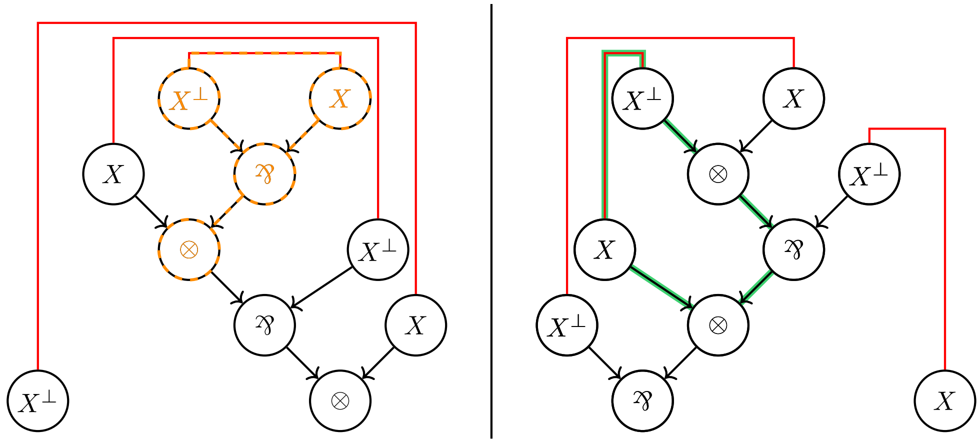
Still, these proof-nets are generated by Quasi-Beffara steps. Indeed, one can apply such a step, yielding the following proof-nets.



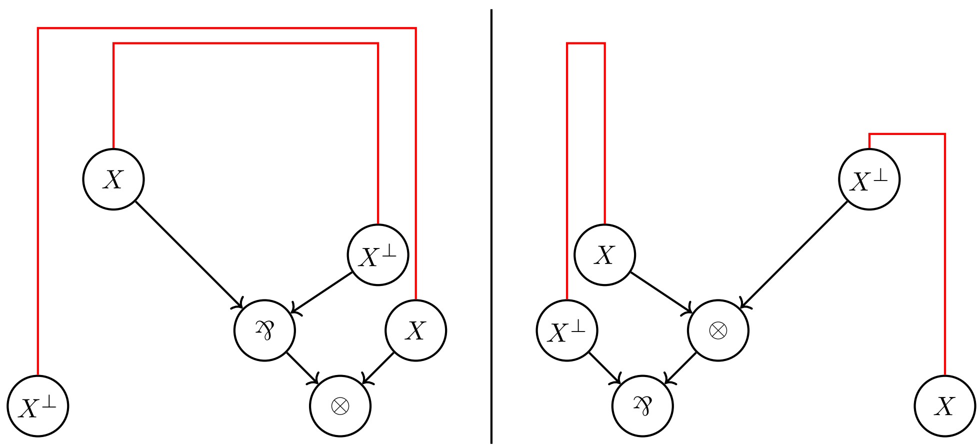
We then recognize the proof-nets of the Beffara retraction. We can apply another Quasi-Beffara on those, yielding the identity proof-net of X.

7.6.2.3 Incorrect proof-structures generated by Quasi-Beffara

The retraction $X \lhd ((X \otimes (X ⅋ X^\perp)) ⅋ X^\perp) \otimes X$ has some proof-structures generated by Quasi-Beffara that are incorrect, and as such not proof-nets. Such an example is the following pair:



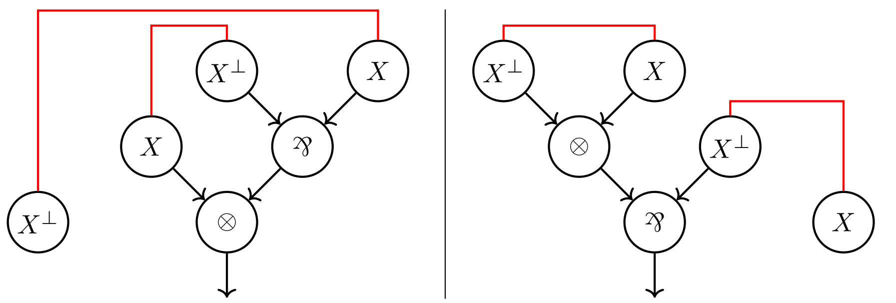
It is incorrect, as can be seen from the green switching cycle contradicting (P2). Nonetheless, one can apply a Quasi-Beffara step on it (recognizing the sub-graph dashed in yellow), resulting on the following proof-structures. They can be recognized as the proof-nets for the Beffara retraction, and as such are attainable from X by one step of Quasi-Beffara.



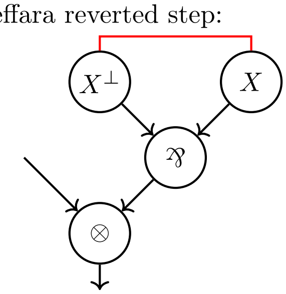
This counter-example proves the following fact: when $\theta \xrightarrow{\text{qBeffara}} \theta'$, whereas the correctness of θ implies the one of θ' (Lemma 7.44), the converse does not hold. In our example, we have θ' correct but not θ, even without considering connectivity. Still, a property holding in both directions is that applying a Quasi-Beffara step on proof-structures preserve the result of their composition, as seen in the proof of Lemma 7.47.

7.6.2.4 Incorrect proof-structures not generated by Quasi-Beffara

Consider the following proof-structures which both do not respect (P2) but on the same sequents as the proof-nets associated to the Beffara retraction, namely with $X \otimes (X^\perp ⅋ X)$ as a retracting formula.



These proof-structures cannot be generated by a succession of Quasi-Beffara, even up to isomorphism. Indeed, both of these proof-structures do not contain the following sub-graph, present in any result of applying a Quasi-Beffara reverted step:



Nonetheless, composing these two proof-structures on the retracting formula gives id_X after cut-elimination.

7.7 Retractions to an arbitrary formula

Let us now go back to the general case: the problem to solve is finding an equational theory for strict retractions $A \lhd B$ in MLL_{uf}. We know $s(A) < s(B)$ (Corollary 7.34).

Here are some general form of retractions I have found, generalizing Beffara's retraction:

- $A \otimes B \lhd A \otimes (A^\perp ⅋ (A \otimes B))$, whose proof-nets when A and B are atoms (which implies the general case) are depicted on Figure 7.1;

- if B or $B^\perp$ is a sub-formula of A, then $A \lhd A \otimes (B^\perp ⅋ B)$, whose proof-nets when B is an atom are depicted on Figure 7.2;

Notice the side-condition in the second retraction, which is not what we would prefer in an inequational theory. In both cases, taking $A = B$ yield instances of Beffara's retractions.

Looking at these two generalizations, one could conjecture that if $A \lhd B$ then A is a strict sub-formula up to isomorphism of B. I have no idea whether it is true or false, but if it holds it would be a really remarkable property.

The main difficulty compared to the atomic case, is that there are more patterns. Remember the pattern we identified when looking at retractions to an atom, which was the crux for solving this case:

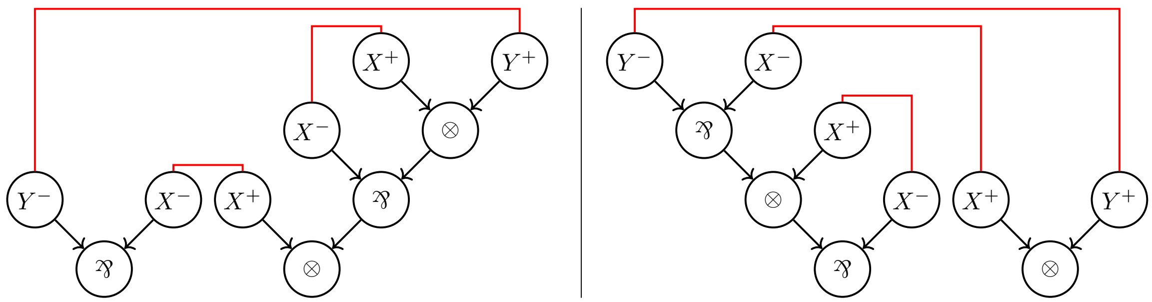
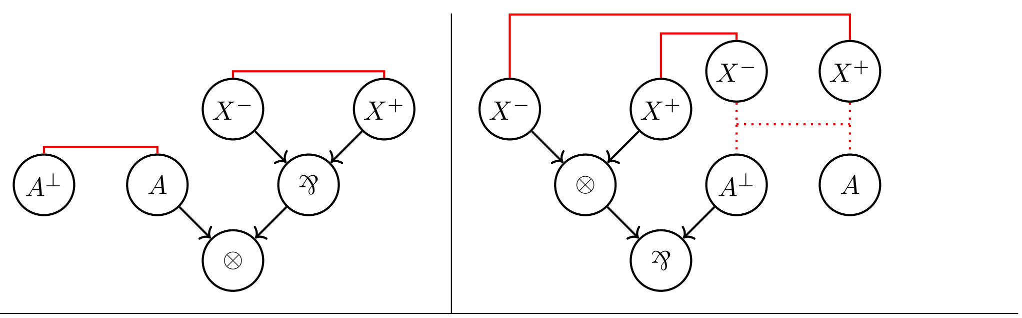
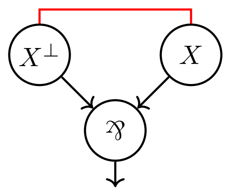


Figure 7.1: Proof-nets for $X^+ \otimes Y^+ \lhd X^+ \otimes (X^- ⅋ (X^+ \otimes Y^+))$

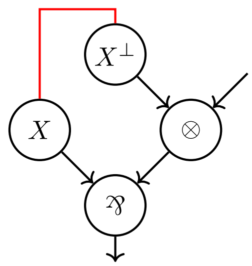


Dotted lines here mean that between $A^\perp$ and A there are the same links as in an identity proof-net, except for the link between X^- and X^+.

Figure 7.2: Proof-nets for $A \lhd A \otimes (X^- ⅋ X^+)$ when X^+ is a sub-formula of A

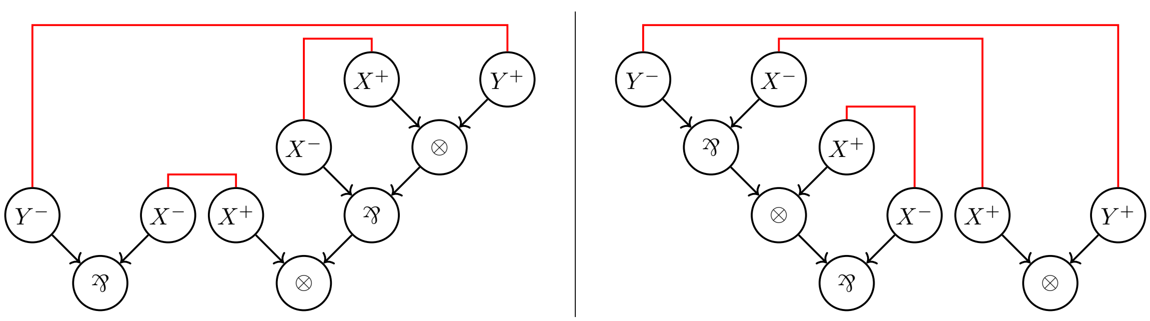


In the general case, we also have the following pattern in retractions (and maybe others?):



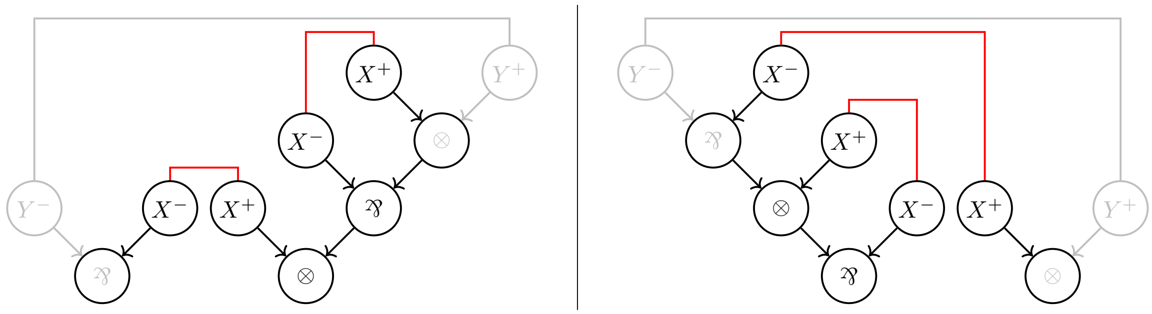
As an example, consider the following retraction, and the associated proof-nets below which are without the first pattern but contain the second one:

$$X^+ \otimes Y^+ \trianglelefteq X^+ \otimes (X^- \parr (X^+ \otimes Y^+))$$



This new pattern is less nice than the first one because it is no more on a full sub-formula $X^- \parr X^+$, but on a part of a sub-formula only.

Nonetheless, the study of the atomic case gives an over-approximation of retractions in the general case. Indeed, given a retraction $A \trianglelefteq B$, one can take (an occurrence of) a signed atom X of A and consider the proof-nets obtained by taking those of $A \trianglelefteq B$ and keeping only links and leaves in the class containing X (recall Definition 6.75). These proof-nets gives a retraction $X \trianglelefteq B'$, because the original proof-nets compose to the identity! For instance, looking at the class of X^+ in the proof-nets of $X^+ \otimes Y^+ \trianglelefteq X^+ \otimes (X^- \parr (X^+ \otimes Y^+))$, we get the followings:



The difficulty for this approach is that it gives that every such "projection" is a retraction to an atom, but unfortunately all proof-structures which such projections are not correct.

Lastly, maybe retractions are not finitely axiomatizable? A problematic example is the following family of retractions, for any natural number n:

$$\left(\otimes(X_i^+)_{i\in[\![1;n]\!]}\right) \lhd \left(\otimes(X_i^+)_{i\in[\![1;n]\!]}\right) \parr (X_1^+ \otimes (X_1^- \parr (\ldots (X_{n-1}^+ \otimes (X_{n-1}^- \parr (X_n^+ \otimes X_n^-)) \ldots)))$$

This family seems hard to obtain, and each retraction also looks difficult to obtain only from previous ones, because the following is not a retraction:

$$(X^+ \otimes Y^+) \parr Z^+ \not\trianglelefteq (X^+ \otimes Y^+) \parr (Y^+ \otimes (Y^- \parr Z^+))$$

7.8 Provability & Decidability

We consider the retraction problem $\mathcal{R}$: given a pair of formulas (A, B), does $A \trianglelefteq B$ hold? We want to know whether $\mathcal{R}$ is decidable or not, and if yes in which complexity class is this problem. In other words: are retractions decidable, and if yes what is their complexity? This problem can be declined in the various sub-systems of linear logic. In all this section, we consider linear logic without any of the optional rules.

We show that provability in some sub-systems reduces to this problem on retractions, giving lower-bounds for the complexity of $\mathcal{R}$ in Section 7.8.1. We then give in Section 7.8.2 an algorithm proving that retractions are decidable in MALL.

7.8.1 Retractions and Provability

While retractions have not been well-studied in linear logic, provability has been extensively considered (at least without the optional rules). In particular, the complexity of provability for various sub-systems have been found. We prove here that solving retractions in some sub-systems would allow to solve provability in these sub-systems, giving lower-bounds for the complexity of retractions. As a remarkable result, we get undecidability in the case of full (propositional) linear logic, and the high lower-bounds in MELL and MALL show that the retraction problem $\mathcal{R}$ is hard in these sub-systems.

We start by giving two couples that are retractions if and only if some sequents are provable.

Lemma 7.64. *In any sub-system with the $\oplus_i$- and &-rules, given two formulas A and B there is a retraction $A \trianglelefteq A \oplus B$ if and only if $\vdash B^\perp, A$ is provable.*

Proof. Let us start with the easier converse way. Assume to have a proof π of $\vdash B^\perp, A$. Set:

$$\phi \quad = \quad \cfrac{\cfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash A^\perp, A \oplus B}\,(\oplus_1) \qquad\qquad \tau \quad = \quad \cfrac{\overset{\pi}{\vdash B^\perp, A} \qquad \cfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash B^\perp \,\&\, A^\perp, A}\,(\&)$$

One can check that $\phi \overset{A \oplus B}{\bowtie} \tau \xrightarrow{\beta} ax_A$:

$$\begin{aligned}
\phi \overset{A \oplus B}{\bowtie} \tau \quad &= \quad \cfrac{\cfrac{\cfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash A^\perp, A \oplus B}\,(\oplus_1) \qquad \cfrac{\overset{\pi}{\vdash B^\perp, A} \qquad \cfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash B^\perp \,\&\, A^\perp, A}\,(\&)}{\vdash A^\perp, A}\,(cut) \\
&\xrightarrow{\beta} \quad \cfrac{\cfrac{}{\vdash A^\perp, A}\,(ax) \qquad \cfrac{}{\vdash A^\perp, A}\,(ax)}{\vdash A^\perp, A}\,(cut) \\
&\xrightarrow{\beta} \quad \cfrac{}{\vdash A^\perp, A}\,(ax)
\end{aligned}$$

Therefore, $A \trianglelefteq A \oplus B$.

For the direct way, suppose $A \trianglelefteq A \oplus B$. In particular, there is a proof π of $\vdash B^\perp \,\&\, A^\perp, A$. We can assume it to be cut-free (Theorem 2.15). Up to rule commutation, one can assume the last rule of π to be a &-rule (this rule is reversible). We thus have:

$$\pi \quad = \quad \cfrac{\overset{\phi}{\vdash B^\perp, A} \qquad \overset{\tau}{\vdash A^\perp, A}}{\vdash B^\perp \,\&\, A^\perp, A}\,(\&)$$

In particular, $\vdash B^\perp, A$ is provable. ☐

Lemma 7.65. *In any sub-system with the $\otimes$-, $⅋$, !- and $?_w$-rules, given a signed atom X and a formula A there is a retraction $X \trianglelefteq X \otimes !A$ if and only if $\vdash A$ is provable.*

Proof. As in the previous proof, let us start by the converse way. Assume to have a proof π of $\vdash A$, and set:

$$\phi \quad = \quad \cfrac{\cfrac{}{\vdash X^\perp, X}\,(ax) \qquad \cfrac{\overset{\pi}{\vdash A}}{\vdash !A}\,(!)}{\vdash X^\perp, X \otimes !A}\,(\otimes) \qquad\qquad \tau \quad = \quad \cfrac{\cfrac{\cfrac{}{\vdash X^\perp, X}\,(ax)}{\vdash ?A^\perp, X^\perp, X}\,(?_w)}{\vdash ?A^\perp ⅋ X^\perp, X}\,(⅋)$$

One can check that $\phi \overset{X \otimes !A}{\bowtie} \tau \xrightarrow{\beta} ax_X$:

$$\phi \overset{B}{\bowtie} \tau \quad = \quad \cfrac{\cfrac{\cfrac{}{\vdash X^\perp, X}\,(ax) \qquad \cfrac{\overset{\pi}{\vdash A}}{\vdash !A}\,(!)}{\vdash X^\perp, X \otimes !A}\,(\otimes) \qquad \cfrac{\cfrac{\cfrac{}{\vdash X^\perp, X}\,(ax)}{\vdash ?A^\perp, X^\perp, X}\,(?_w)}{\vdash ?A^\perp ⅋ X^\perp, X}\,(⅋)}{\vdash X^\perp, X}\,(cut)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{}{\vdash X^\perp, X}\,(ax) \qquad \dfrac{\dfrac{\overset{\pi}{\vdash A}}{\vdash\, !A}\,(!) \qquad \dfrac{\dfrac{}{\vdash X^\perp, X}\,(ax)}{\vdash\, ?A^\perp, X^\perp, X}\,(?_w)}{\vdash X^\perp, X}\,(cut)}{\vdash X^\perp, X}\,(cut)$$

$$\xrightarrow{\beta} \quad \dfrac{\dfrac{}{\vdash X^\perp, X}\,(ax) \qquad \dfrac{}{\vdash X^\perp, X}\,(ax)}{\vdash X^\perp, X}\,(cut) \quad \xrightarrow{\beta} \quad \dfrac{}{\vdash X^\perp, X}\,(ax)$$

Therefore, $X \trianglelefteq X \otimes\, !A$.

For the direct way, suppose $X \trianglelefteq X \otimes\, !A$. In particular, there is a proof π of $\vdash X^\perp, X \otimes\, !A$, that can be assumed cut-free (Theorem 2.15). The only non-*cut*-rule that can be applied on $\vdash X^\perp, X \otimes !A$ is a $\otimes$-rule. As $\vdash$ and $\vdash X$ are not provable (simply because no non-*cut*-rule can be applied on these sequents), we must have the following:

$$\pi \quad = \quad \dfrac{\overset{\tau}{\vdash X^\perp, X} \qquad \overset{\phi}{\vdash\, !A}}{\vdash X^\perp, X \otimes\, !A}\,(\otimes)$$

But the last rule of the cut-free ϕ must be a !-rule, for it is the sole applicable rule on $\vdash\, !A$. Thus, $\vdash A$ is provable. □

These two couples allow us to reduce the problem of provability to the one of retractions $\mathcal{R}$. The problem of provability is the following: given a sequent $\vdash \Gamma$, is it provable, and if yes what is the complexity of deciding this? The complexity of provability is known in many sub-systems; an overview of such results can be found in [Lin95; Chu21], with the additive case in [HH15, Theorem 17]. The following table introduces the complexity of provability in the main sub-systems.

Sub-system	Complexity of Provability
LL	Undecidable
MELL	TOWER-hard (decidability is open)
MALL	PSPACE-complete
MLL	NP-complete
ALL	in P (linear in the product of the size of the formulas)
ELL	in P

As a corollary of Lemmas 7.64 and 7.65, the above table also gives lower-bounds for the complexity of retractions in these sub-systems (excepted MLL and ELL). Indeed, in ALL provable sequents have exactly two formulas, so that Lemma 7.64 reduces provability to retraction in this system. In a system with at least the $\otimes$-, $⅋$, !- and $!_w$-rules, such as MELL and LL, deciding if a sequent is provable is the same as deciding if a formula is (using $⅋$ to concatenate the formulas of the sequent), which can be solved by looking if a couple is a retraction according to Lemma 7.65. Finally, in MALL, provability of a sequent, which is equivalent to provability of a formula, is also equivalent to provability of a sequent with two formulas. Indeed, $\vdash A$ is provable if and only if $\vdash X^\perp, X \otimes A$ is: for the direct way, given a proof π of $\vdash A$, $\dfrac{\dfrac{}{\vdash X^\perp, X}\,(ax) \qquad \overset{\pi}{\vdash A}}{\vdash X^\perp, X \otimes A}\,(\otimes)$ is a proof of

$\vdash X^\perp, X \otimes A$; reciprocally, any cut-free proof of $\vdash X^\perp, X \otimes A$ must end with a $\otimes$-rule of the shape $\dfrac{\vdash X^\perp, X \quad \vdash A}{\vdash X^\perp, X \otimes A}\ (\otimes)$ which yields a proof of $\vdash A$. Remark in particular that deciding if a couple is a retraction in the general case is undecidable!

As a last remark, $\mathcal{R}$ for MLL_{uf} is in NP. Indeed, one can reduce the problem to provability thanks to Corollary 7.29 and Theorem 7.30. To sum up, here is a table grouping the results of this section and the next one together.

Sub-system	Complexity of Retraction
LL	Undecidable
MELL	TOWER-hard (undecidable if provability is)
MALL	PSPACE-hard & Decidable
MLL	in NP
ALL	at least P

7.8.2 Decidability of retractions

We just saw that $\mathcal{R}$ in the general case is undecidable. In MELL, if $\mathcal{R}$ is decidable then provability in this sub-system would also be decidable (using Lemma 7.65). On the other hand, if $\mathcal{R}$ is undecidable then we cannot say anything about decidability of provability.

Algorithm 1: Deciding if a pair of formulas is a retraction in MALL

Data: A, B formulas
Result: boolean asserting $A \trianglelefteq B$

$\Pi \leftarrow$ the set of all atomic-axiom cut-free proofs of $\vdash A^\perp, A$ in the equivalence class of id_A for $\overset{r^*}{\vdash\!\dashv}$;
for π *atomic-axiom cut-free proof of* $\vdash A^\perp, B$ **do**
  for ϕ *atomic-axiom cut-free proof of* $\vdash B^\perp, A$ **do**
    $\tau \leftarrow$ a result of eliminating all cuts in $\pi \overset{B}{\bowtie} \phi$;
    if $\tau \in \Pi$ **then**
      return True;
return False;

We prove here that $\mathcal{R}$ is decidable in MALL, by giving a (terminating) algorithm solving it: Algorithm 1. This algorithm halts: there is a finite number of cut-free proofs of a given sequent in MALL, because all rules of cut-free MALL have the strict sub-sequent property (Fact 1.7). Thence, computing all needed proofs can be done, and checking if a proof belongs to the equivalence class for $\overset{r^*}{\vdash\!\dashv}$ of another can also be done in finite time, by computing iteratively the proofs that can be reached in n commutations. Lastly, reducing all cuts can be done in finite time (Corollary 2.43).

This algorithm is also correct. Indeed, $A \trianglelefteq B$ if and only if there are atomic-axiom cut-free proofs whose composition is β-equal to id_A (Lemma 7.8). By Theorem 2.49, this is the case if and only if any normal form of the composition is in the equivalence class of id_A for $\overset{r^*}{\vdash\!\dashv}$.

Inasmuch as it allows deciding $\mathcal{R}$ in MALL, Algorithm 1 does not apply with the exponential connectives, because there is no longer a finite number of atomic-axiom cut-free proofs of a given sequent due to the $?_c$-rule.

7.9 Retractions up to cut-elimination only

We argue in this section that retractions in $\mathsf{MLL}^{0,2}$ when proofs are not considered up to $\beta\eta$-equality, but only β-equality, are given by retractions to an atom, that we solved. As done for isomorphisms (in Section 6.5), this means that in the definition of retractions (Definition 7.1) one replaces $=_{\beta\eta o}$ by $=_\beta$.

Lemma 7.66. *Let A and B be two formulas in $\mathsf{MLL}^{0,2}$ along cut-free proofs π of $\vdash A^\perp, B$ and ϕ of $\vdash B^\perp, A$ such that $\pi \overset{B}{\bowtie} \phi =_\beta ax_A$. Then some occurrences of A in B can be replaced by any atom X^+, in a substitution on occurrences we call α, such that α applied to π and ϕ yield proofs, $\alpha(B)^\perp = \alpha(B^\perp)$ and $X^+ \overset{\alpha(\pi),\alpha(\phi)}{\trianglelefteq} \alpha(B)$.*

Proof. As no rule commutation can be applied on ax_A, we know cut-elimination in $\pi \overset{B}{\bowtie} \phi$ leads to a unique normal form, ax_A (Theorem 2.49). Call $A_r^\perp$ and A_r the occurrences corresponding to the retract formulas $A^\perp$ and A. We first prove there is an ax-rule on $A_r^\perp$ (resp. A_r) in π (resp. ϕ). If it were not the case, then applying cut-elimination steps can never create an ax-rule on this occurrence, using that in MLL there is a unique rule applied on each occurrence in a proof.

Now, we show that there is a sequence of ax-rules between $A_r^\perp$ in π and A_r in ϕ, by following ax-rules and dual occurrences as done in the proof of Lemma 7.40. Say the ax-rule in π on $A_r^\perp$ is applied on an occurrence A_1, a sub-formula of B (because π is cut-free, applying Fact 1.7). Consider the dual occurrence $A_1^\perp$ in ϕ. If no ax-rule is applied on it, then when reducing cuts in $\pi \overset{B}{\bowtie} \phi$ one ends up with a non-ax-rule on $A_r^\perp$. Indeed, apply all possible cut-elimination steps which do not involve the ax-rule on $\vdash A_r^\perp, A_1$ nor the rule r on $A_1^\perp$. This gives a cut-rule of the form $\dfrac{\dfrac{}{\vdash A_r^\perp, A_1}(ax) \quad \dfrac{}{\vdash A_1^\perp, \Gamma}(r)}{\vdash A_r^\perp, \Gamma}(cut)$ in the resulting proof, thanks to strong normalization of $\xrightarrow{\overline{\beta}} \cdot \overset{c^*}{\mapsto}$ (Proposition 2.41) allowing to move a cut-rule up to this point and to eliminate all cut-rules above. Applying an ax key case, one gets a non-ax-rule r of main formula $A_r^\perp$, which cannot be erased by cut-elimination: absurd as we get an ax-rule at the end. Remark this also means that at any point during any cut-elimination procedure applied in $\pi \overset{B}{\bowtie} \phi$, while $A_1^\perp$ is in this proof it must have an ax-rule on it.

Therefore, there is an ax-rule in ϕ on $\vdash A_1^\perp, A_2$, for A_2 some occurrence of A. If $A_2 = A_r$ (as occurrences), then we are done: we rename all these occurrences into X^+ and X^- and obtain the wished result, as we took care to rename B and $B^\perp$ into two dual formulas. Otherwise, we repeat the above reasoning by replacing A_1 with A_2: we get an ax-rule on $\vdash A_2^\perp, A_3$ in π, and so on. This process does not loop, because when finding A_i we have at disposition ax-rules on all observed occurrences excepted for $A_i^\perp$ and A_r, so new ax-rules can only yield $A_{i+1} \neq A_j$ for $j \in [\![1; i]\!]$. $\square$

As one can always assume proofs associated to a retraction to be cut-free (Corollary 2.43), this means retractions up to cut-elimination only are the same as retractions to an atom – the reciprocal being Lemma 7.8.

7.10 Perspectives

We proved in this chapter that the usual main difficulties when considering $\mathsf{MLL}^{0,2}$ – namely the units and the mix_2-rule – can be easily taken care of for the study of retractions. Thus, the problem reduces to the MLL_{uf} case, which is one of the simplest sub-systems of linear logic. Still, retractions even in this setting are complicated. Looking at our analysis on complexity, it is even worse for bigger sub-systems such as MALL or MELL.

One could consider the other two simplest sub-systems of linear logic: ALL and ELL. For the *additive* formulas, the units also brings nothing to the table, and there seems to be only one "basic" strict retraction:

$$A \trianglelefteq A \,\&\, B \iff \vdash A^{\perp}, B \qquad \text{or dually} \qquad A \trianglelefteq A \oplus B \iff \vdash A, B^{\perp}$$

Here again, retraction to an atom are manageable, this time because provability of $\vdash X, A$ behaves well. The general case is not as trivial, for composition of the two above retractions is bad due to the side condition: $X \oplus Y \triangleleft ((X \oplus Z) \,\&\, (X \oplus Y)) \oplus Y$ comes from $X \oplus Y \triangleleft (X \oplus Y) \oplus Y$ but we dot not have $\vdash X \oplus Z, (X \oplus Y)^{\perp}$. It can also be proved that the Cantor-Bernstein-Schröeder property holds in ALL. But even just proving Cantor-Bernstein-Schröeder for MALL seems hard. It would also not be surprising if there was the following result for distributed formulas of MALL_{uf}: if $A \trianglelefteq B$, then $s(A) \leq s(B)$ with equality if and only if it is an isomorphism.

Considering the *exponential* formulas, even if it is conjectured there is no (non-reflexive) isomorphism, there are retractions:

$$?A \trianglelefteq ??A \qquad\qquad ?!A \trianglelefteq ?!?!A$$

Again, these look like the one corresponding to an inequational theory characterizing retractions of ELL, but proving it is not trivial. This is because this system is quite strange, with the $?_w$- and $?_c$-rules that should not matter in the case of retractions.

Conclusion

In this thesis, we considered equality of proofs and of formulas in linear logic, identifying proofs equal up to cut-elimination and axiom-expansion as it is the minimum assimilation one can wish for. We focused on $\mathsf{MALL}^{0,2}$ and worked in both the syntaxes of sequent calculus and of proof-nets to achieve this goal, using tools from various fields such as abstract rewriting systems or graph theory.

After recalling the necessary definitions from sequent calculus in Chapter 1, we first considered equality of proofs. We demonstrated in Chapter 2 that in $\mathsf{MALL}^{0,2}$ two proofs are identified up to cut-elimination and axiom-expansion – and possibly *mix*-Rétoré – if and only if their normal forms are related by rule commutations – and with *mix*-Rétoré also up to *mix*-Rétoré (see Propositions 2.10 and 2.46 and Theorems 2.47 and 2.48). This proof was based on a key result about strong normalization of $\xrightarrow{\overline{\beta}} \cdot (\overset{r\setminus\top}{\rightsquigarrow} \cup \overset{c}{\vdash\!\dashv} \cup \overset{om}{\rightsquigarrow})^*$ – *i.e.* of cut-elimination without the $cut - cut$ commutative case, modulo rule commutation and $cut - cut$ commutation (Proposition 2.41). We conjecture these two results on strong normalization and Church-Rosser modulo extend to the full system, with a very similar proof for Church-Rosser modulo and a quite different one for strong normalization. For this last proof, we gave an extended sketch based on a paper about strong normalization in proof-nets [PT10]. Formalizing this sketch into a proper proof should be a matter of verifying the steps from [PT10] could be adapted as stated, a work that may be tedious but not overly complicated. While our results on cut-elimination, and even their generalizations to the full logic, are in no way unexpected, the absence of a proof is a big blind spot of the literature, especially as these results underlay motivations and results from many papers. A striking corollary is that proof-nets are a canonical representation of proofs up to cut-elimination and axiom-expansion, because they represent proofs exactly up to rule commutations. This explains a lot of their good properties, *e.g.* convergence of cut-elimination, and was claimed without proof for decades. Furthermore, that rule commutation (defined as the minimum relation to have cut-elimination Church-Rosser modulo it) should not contain any commutation with an !-rule is noteworthy.

Going to the proof-net syntax, we then looked at usual graph theory in Chapter 3, where we proved a generalization of Yeo's theorem [Yeo97] (Theorem 3.19). This generalization allows for the presence of very particular bridge-free cycles, as well as a more general notion of edge-coloring. Whereas the conditions on these cycles may not be usual for a graph theoretician, they allow us to extend the links between standard graph theory and proof-nets from $\mathsf{MLL}_{uf}^{0,2}$ to $\mathsf{MALL}_{uf}^{0,2}$. Then, we recalled in Chapter 4 the definition of proof-nets for $\mathsf{MALL}_{uf}^{0,2}$ from [HG05]. We gave a new simpler proof of the sequentialization theorem, a usual hard theorem allowing to recover a proof of sequent calculus from a proof-net. It is obtained mainly as a corollary of our generalization of Yeo's theorem from the preceding chapter. Not only is it possible to obtain such a proof by splitting terminal vertex, modifying the parameter from our graph theorem allows to recover other kinds of sequentialization: we recover all proofs by means of a splitting vertex in one go, for plenty of variants in the definition of proof-nets – *e.g.* by splitting $\parr\backslash\&$-vertices for $\mathsf{MALL}^{0,2}_{uf}$ proof-nets. When restricted to $\mathsf{MLL}^{0,2}_{uf}$ proof-nets, this sequentialization proof is the simplest we are aware of, simple enough to be given in an undergraduate lecture. Such an approach seems applicable to other notions of proof-nets, giving a kind of unifying framework for proofs of sequentialization by splitting vertices. Thus, this method advocates for more links between proof-nets and the more usual theory of graphs, in the same vein as previous papers such as [Ret03; Ngu20].

We also formalized proof-nets for MLL_{uf} in the proof assistant Coq in Chapter 5. More precisely, we proved on machine the sequentialization theorem, and a normalization of cut-elimination. Despite many difficulties arising from the manipulation of graphs on a computer, this successful implementation indicates this is possible with the key points on paper presenting no real problem. Many concepts and results about proof-nets have not been formalized, *e.g.* axiom-expansion, simulation of cut-elimination from sequent calculus to proof-nets, $\mathsf{MELL}^{0,2}$ and $\mathsf{MALL}^{0,2}_{uf}$ proof-nets, etc.

The proof-net syntax was a key ingredient in the rest of this thesis, which was about the equality of formulas. We characterized isomorphisms of MALL^2 (as well as those in $\star$-autonomous categories with finite products) in Chapter 6, extending the result of Balat and Di Cosmo for MLL (and $\star$-autonomous categories) in [BD99]. While the equational theory associated to isomorphisms is the expected one (Table 6.1 for MALL^2 and Table 6.2 for $\star$-autonomous categories), proving it was by no mean easy. Even if the base idea of distributing formulas was a simple one, so that only associativity and commutativity isomorphisms are left, exploiting the global information that a formula is distributed to deduce local properties on the associated proofs necessitated thorough studies. The previous work on sequent calculus was needed to reduce the study to the unit-free setting, the first step of the proof. Whereas working on equivalence classes of rule commutation was source of many technicalities, it was still manageable, contrary to considering proofs up to cut-elimination directly. For solving the unit-free case, proof-nets were instrumental. Not only did we recover canonicity of the representation of proofs, no more working in equivalence classes, but the geometrical shapes found (*e.g.* fullness and ax-uniqueness) and reasoning applied (in particular through the correctness criterion) seem hardly mimickable in sequent calculus without a gap in complexity. The extension of this characterization to the exponentials, so as to consider the full propositional linear logic, is challenging as there is no associated notion of proof-nets. The main difficulty could be the $\overset{oe}{\leadsto}$ transformation, which is needed to get the exponential isomorphisms but not expected to be included in $\beta\eta$-equality. Still, there is a well-known conjecture on what isomorphisms are (see Conjecture 6.103), giving a directing line.

Meanwhile, the situation is way more complex for retractions. We studied retractions of $\mathsf{MLL}^{0,2}$ in Chapter 7 and proved that all type-retractions (but not all retractions!) from an atom are obtained from the Beffara's retraction. While we have some general results such as a proof of the Cantor-Bernstein-Schröeder property for $\mathsf{MLL}^{0,2}$, and generally that a retraction in MLL_{uf} between two formulas of the same size is an isomorphism, we were not successful in solving the problem. What is most particular is that the usual main difficulties from $\mathsf{MLL}^{0,2}$, the units and the mix_2-rule, are here irrelevant. Thence, all difficulties come from MLL_{uf}, one of the simplest sub-systems of linear logic, and surely the most studied. Nonetheless, even there retractions are complicated. Looking at bigger sub-systems, complexity bounds obtained from provability indicate the problem cannot be easy, and is even indecidable in the case of propositional lineal logic.

Appendices

Appendix A

Case studies for Sequent Calculus

We give here the complete proofs of case studies from Chapter 2. They were not given in the main part for clarity and brevity sake, hence the proofs here are quite long, tedious, repetitive and boring. These proofs are mostly given for completeness sake. No sane reader is expected to peruse all proofs here, that in the end are mostly computational content, better suited to be computer-checked in a proof assistant.

Lemma 2.27. *For a proof π of a sequent $\vdash \Gamma$, $w(\pi) \geq w(\vdash \Gamma)$. Furthermore, if π is cut-free, then $w(\pi) = w(\vdash \Gamma)$.*

Proof. By induction on the proof π.

If $\pi = \dfrac{}{\vdash A^\perp, A}\,(ax)$. Then π is cut-free and $w(\pi) = w(A^\perp) + w(A) = w(\vdash A^\perp, A)$.

If $\pi = \dfrac{\begin{array}{c}\pi_1\\ \vdash A, \Gamma\end{array} \quad \begin{array}{c}\pi_2\\ \vdash A^\perp, \Delta\end{array}}{\vdash \Gamma, \Delta}\,(cut)$. Then π is not cut-free and

$$\begin{aligned}
w(\pi) &= w(\pi_1) + w(\pi_2)\\
&\geq w(\vdash A, \Gamma) + w(\vdash A^\perp, \Delta) \text{ by induction hypothesis}\\
&= w(A) + w(\vdash \Gamma) + w(A^\perp) + w(\vdash \Delta)\\
&> w(\vdash \Gamma) + w(\vdash \Delta) \text{ by Fact 2.25}\\
&= w(\vdash \Gamma, \Delta)
\end{aligned}$$

If $\pi = \dfrac{\begin{array}{c}\pi_1\\ \vdash A, \Gamma\end{array} \quad \begin{array}{c}\pi_2\\ \vdash B, \Delta\end{array}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes)$. Remark π is cut-free if and only if π_1 and π_2 are. Then

$$\begin{aligned}
w(\pi) &= w(\pi_1) + w(\pi_2) + 1\\
&\geq w(\vdash A, \Gamma) + w(\vdash B, \Delta) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free}\\
&= w(A) + w(\vdash \Gamma) + w(B) + w(\vdash \Delta) + 1\\
&= w(A \otimes B) + w(\vdash \Gamma) + w(\vdash \Delta)\\
&= w(\vdash A \otimes B, \Gamma, \Delta)
\end{aligned}$$

If $\pi = \dfrac{\begin{array}{c}\pi_1 \\ \vdash A, B, \Gamma\end{array}}{\vdash A \parr B, \Gamma}\ (\parr)$. Remark that π is cut-free if and only if π_1 is. Then

$$\begin{aligned}
w(\pi) &= w(\pi_1) + 1 \\
&\geq w(\vdash A, B, \Gamma) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\
&= w(A) + w(B) + w(\vdash \Gamma) + 1 \\
&= w(A \parr B) + w(\vdash \Gamma) \\
&= w(\vdash A \parr B, \Gamma)
\end{aligned}$$

If $\pi = \dfrac{}{\vdash 1}\ (1)$. Then π is cut-free and $w(\pi) = 1 = w(1) = w(\vdash 1)$.

If $\pi = \dfrac{\begin{array}{c}\pi_1 \\ \vdash \Gamma\end{array}}{\vdash \bot, \Gamma}\ (\bot)$. Remark that π is cut-free if and only if π_1 is. Then

$$\begin{aligned}
w(\pi) &= w(\pi_1) + 1 \\
&\geq w(\vdash \Gamma) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\
&= w(\vdash \Gamma) + w(\bot) \\
&= w(\vdash \bot, \Gamma)
\end{aligned}$$

If $\pi = \dfrac{\begin{array}{c}\pi_1 \\ \vdash A, \Gamma\end{array} \quad \begin{array}{c}\pi_2 \\ \vdash B, \Gamma\end{array}}{\vdash A \,\&\, B, \Gamma}\ (\&)$. Remark that π is cut-free if and only if π_1 and π_2 are. Then

$$\begin{aligned}
w(\pi) &= \max(w(\pi_1), w(\pi_2)) + 1 \\
&\geq \max(w(\vdash A, \Gamma), w(\vdash B, \Gamma)) + 1 \text{ by induction, with equality if } \pi \text{ is cut-free} \\
&= \max(w(A) + w(\vdash \Gamma), w(B) + w(\vdash \Gamma)) + 1 \\
&= \max(w(A), w(B)) + w(\vdash \Gamma) + 1 \\
&= w(A \,\&\, B) + w(\vdash \Gamma) \\
&= w(\vdash A \,\&\, B, \Gamma)
\end{aligned}$$

If $\pi = \dfrac{\begin{array}{c}\pi_1 \\ \vdash A_1, \Gamma\end{array}}{\vdash A_1 \oplus A_2, \Gamma}\ (\oplus_1)$. Remark that π is cut-free if and only if π_1 is. Then

$$\begin{aligned}
w(\pi) &= w(\pi_1) + w(A_2) + 1 \\
&\geq w(\vdash A_1, \Gamma) + w(A_2) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\
&= w(\vdash \Gamma) + w(A_1) + w(A_2) + 1 \\
&= w(\vdash \Gamma) + w(A_1 \oplus A_2) \\
&= w(\vdash A_1 \oplus A_2, \Gamma)
\end{aligned}$$

If $\pi = \dfrac{\begin{array}{c}\pi_1 \\ \vdash A_2, \Gamma\end{array}}{\vdash A_1 \oplus A_2, \Gamma}\ (\oplus_2)$. Remark that π is cut-free if and only if π_1 is. Then

$$\begin{aligned} w(\pi) &= w(\pi_1) + w(A_1) + 1 \\ &\geq w(\vdash A_2, \Gamma) + w(A_1) + 1 \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\ &= w(\vdash \Gamma) + w(A_1) + w(A_2) + 1 \\ &= w(\vdash \Gamma) + w(A_1 \oplus A_2) \\ &= w(\vdash A_1 \oplus A_2, \Gamma) \end{aligned}$$

If $\pi = \dfrac{}{\vdash \top, \Gamma}\ (\top)$. Then π is cut-free and $w(\pi) = 1 + w(\vdash \Gamma) = w(\vdash \top, \Gamma)$.

If $\pi = \dfrac{\begin{array}{cc}\pi_1 & \pi_2 \\ \vdash \Gamma & \vdash \Delta\end{array}}{\vdash \Gamma, \Delta}\ (mix_2)$. Remark π is cut-free if and only if π_1 and π_2 are. Then

$$\begin{aligned} w(\pi) &= w(\pi_1) + w(\pi_2) \\ &\geq w(\vdash \Gamma) + w(\vdash \Delta) \text{ by induction hypothesis, with equality if } \pi \text{ is cut-free} \\ &= w(\vdash \Gamma, \Delta) \end{aligned}$$

If $\pi = \dfrac{}{\vdash}\ (mix_0)$. Then π is cut-free and $w(\pi) = 0 = w(\vdash\ )$. ☐

Proposition 2.46 ($\overset{rc}{\longmapsto\!\dashv} \subseteq =_\beta$)**.** *Given proofs π and τ in $\mathsf{MALL}^{0,2}$, if $\pi \overset{rc}{\longmapsto\!\dashv} \tau$ then $\pi \overset{\beta^*}{\longleftarrow} \cdot \overset{\beta^*}{\longrightarrow} \tau$. In particular, if $\pi \overset{r}{\longmapsto\!\dashv} \tau$ then $\pi =_\beta \tau$.*

Proof. It suffices, for each commutation, to give a proof that can be reduced by cut-elimination to both sides of the commutation – using that both rule commutation and cut-elimination are contextual, so that one can ignore what happens below the rule commutation. Remark that this trivially holds for $\overset{c}{\longmapsto\!\dashv}$, as well as for other commutations of $\overset{rc}{\longmapsto\!\dashv}$ involving a *cut*-rule: two such proofs are related by $\overset{\overline{\beta}}{\longrightarrow}$ or $\overset{\overline{\beta}}{\longleftarrow}$. As there are 152 cases in general, we treat here only the rule commutations of $\mathsf{MALL}^{0,2}$. In all cases here, we give a proof with a single *cut*-rule above two others proofs, such that the above proof reduces to the left one or to the right one according respectively to whether the *cut*-rule is first commuted with the rule on its left or on its right – once this first commutation is done, there will always be a single result of cut-elimination, that can be reached using only $\overset{\overline{\beta}}{\longrightarrow}$ steps.

- $C_{\parr}^{\parr}$ *commutation*

$$\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, D, \Gamma}}{\vdash A \parr B, C, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr) \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^{\perp}, B}\,(ax) \qquad \dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash B^{\perp} \otimes A^{\perp}, A, B}\,(\otimes)}{\vdash B^{\perp} \otimes A^{\perp}, A \parr B}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, D, \Gamma}}{\vdash A \parr B, C, D, \Gamma}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, D, \Gamma}}{\vdash A, B, C \parr D, \Gamma}\,(\parr)}{\vdash A \parr B, C \parr D, \Gamma}\,(\parr)$$

- $C_{\otimes}^{\parr}$ *and* $C_{\parr}^{\otimes}$ *commutations*

$$\dfrac{\dfrac{\dfrac{}{\vdash C^{\perp}, C}\,(ax) \qquad \overset{\phi}{\vdash D, \Delta}}{\vdash C^{\perp}, C \otimes D, \Delta}\,(\otimes) \qquad \dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash C, A \parr B, \Gamma}\,(\parr)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma}}{\vdash A \parr B, C, \Gamma}\,(\parr) \qquad \overset{\phi}{\vdash D, \Delta}}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\otimes) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, B, C, \Gamma} \qquad \overset{\phi}{\vdash D, \Delta}}{\vdash A, B, C \otimes D, \Gamma, \Delta}\,(\otimes)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\parr)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \dfrac{}{\vdash D^{\perp}, D}\,(ax)}{\vdash D^{\perp}, C \otimes D, \Gamma}\,(\otimes) \qquad \dfrac{\overset{\phi}{\vdash A, B, D, \Delta}}{\vdash D, A \parr B, \Delta}\,(\parr)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, B, D, \Delta}}{\vdash A \parr B, D, \Delta}\,(\parr)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\otimes) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash C, \Gamma} \qquad \overset{\phi}{\vdash A, B, D, \Delta}}{\vdash A, B, C \otimes D, \Gamma, \Delta}\,(\otimes)}{\vdash A \parr B, C \otimes D, \Gamma, \Delta}\,(\parr)$$

- $C_{\perp}^{\parr}$ *and* $C_{\parr}^{\perp}$ *commutations*

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1,\bot}\,(\bot) \qquad \dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,\Gamma\end{array}}{\vdash A,B,\bot,\Gamma}\,(\bot)}{\vdash A \parr B,\bot,\Gamma}\,(\parr)}{\vdash A \parr B,\bot,\Gamma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,\Gamma\end{array}}{\vdash A \parr B,\Gamma}\,(\parr)}{\vdash A \parr B,\bot,\Gamma}\,(\bot) \qquad\qquad\qquad \dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,\Gamma\end{array}}{\vdash A,B,\bot,\Gamma}\,(\bot)}{\vdash A \parr B,\bot,\Gamma}\,(\parr)$$

- $C_{\&}^{\parr}$ *and* $C_{\parr}^{\&}$ *commutations.*

$$\dfrac{\dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array}}{\vdash A \parr B,C,\Gamma}\,(\parr) \quad \dfrac{\begin{array}{c}\phi\\ \vdash A,B,D,\Gamma\end{array}}{\vdash A \parr B,D,\Gamma}\,(\parr)}{\vdash A \parr B,C \& D,\Gamma}\,(\&) \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^{\bot},B}\,(ax) \quad \dfrac{}{\vdash A^{\bot},A}\,(ax)}{\vdash A,B,B^{\bot}\otimes A^{\bot}}\,(\otimes)}{\vdash A \parr B,B^{\bot}\otimes A^{\bot}}\,(\parr)}{\vdash A \parr B,C \& D,\Gamma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array}}{\vdash A \parr B,C,\Gamma}\,(\parr) \quad \dfrac{\begin{array}{c}\phi\\ \vdash A,B,D,\Gamma\end{array}}{\vdash A \parr B,D,\Gamma}\,(\parr)}{\vdash A \parr B,C \& D,\Gamma}\,(\&) \qquad\qquad \dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array} \quad \begin{array}{c}\phi\\ \vdash A,B,D,\Gamma\end{array}}{\vdash A,B,C \& D,\Gamma}\,(\&)}{\vdash A \parr B,C \& D,\Gamma}\,(\parr)$$

- $C_{\oplus_i}^{\parr}$ *and* $C_{\parr}^{\oplus_i}$ *commutations* We consider by symmetry only the $\oplus_1$-rule.

$$\dfrac{\dfrac{\dfrac{}{\vdash C^{\bot},C}\,(ax)}{\vdash C^{\bot},C\oplus D}\,(\oplus_1) \qquad \dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array}}{\vdash A \parr B,C,\Gamma}\,(\parr)}{\vdash A \parr B,C\oplus D,\Gamma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array}}{\vdash A \parr B,C,\Gamma}\,(\parr)}{\vdash A \parr B,C\oplus D,\Gamma}\,(\oplus_1) \qquad\qquad\qquad \dfrac{\dfrac{\begin{array}{c}\pi\\ \vdash A,B,C,\Gamma\end{array}}{\vdash A,B,C\oplus D,\Gamma}\,(\oplus_1)}{\vdash A \parr B,C\oplus D,\Gamma}\,(\parr)$$

- $C_{\top}^{\parr}$ *and* $C_{\parr}^{\top}$ *commutations*

$$\dfrac{\dfrac{}{\vdash \top, 0}\,{\scriptstyle(\top)} \qquad \dfrac{\dfrac{}{\vdash A, B, \top, \Gamma}\,{\scriptstyle(\top)}}{\vdash A \parr B, \top, \Gamma}\,{\scriptstyle(\parr)}}{\vdash A \parr B, \top, \Gamma}\,{\scriptstyle(cut)}$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{}{\vdash A \parr B, \top, \Gamma}\,{\scriptstyle(\top)} \qquad\qquad\qquad \dfrac{\dfrac{}{\vdash A, B, \top, \Gamma}\,{\scriptstyle(\top)}}{\vdash A \parr B, \top, \Gamma}\,{\scriptstyle(\parr)}$$

- *$C_{\parr}^{mix_2}$ and $C_{mix_2}^{\parr}$ commutations*

$$\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}\,{\scriptstyle(\parr)} \quad \overset{\phi}{\vdash \Delta}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(mix_2)} \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^{\perp}, B}\,{\scriptstyle(ax)} \quad \dfrac{}{\vdash A^{\perp}, A}\,{\scriptstyle(ax)}}{\vdash A, B, B^{\perp} \otimes A^{\perp}}\,{\scriptstyle(\otimes)}}{\vdash A \parr B, B^{\perp} \otimes A^{\perp}}\,{\scriptstyle(\parr)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(cut)}$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma}}{\vdash A \parr B, \Gamma}\,{\scriptstyle(\parr)} \quad \overset{\phi}{\vdash \Delta}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(mix_2)} \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, B, \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash A, B, \Gamma, \Delta}\,{\scriptstyle(\otimes)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(\parr)}$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, B, \Delta}}{\vdash A \parr B, \Delta}\,{\scriptstyle(\parr)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(mix_2)} \qquad \dfrac{\dfrac{\dfrac{}{\vdash B^{\perp}, B}\,{\scriptstyle(ax)} \quad \dfrac{}{\vdash A^{\perp}, A}\,{\scriptstyle(ax)}}{\vdash A, B, B^{\perp} \otimes A^{\perp}}\,{\scriptstyle(\otimes)}}{\vdash A \parr B, B^{\perp} \otimes A^{\perp}}\,{\scriptstyle(\parr)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(cut)}$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, B, \Delta}}{\vdash A \parr B, \Delta}\,{\scriptstyle(\parr)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(mix_2)} \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, B, \Delta}}{\vdash A, B, \Gamma, \Delta}\,{\scriptstyle(mix_2)}}{\vdash A \parr B, \Gamma, \Delta}\,{\scriptstyle(\parr)}$$

- *$C_{\otimes}^{\otimes}$ commutations.*

$$\dfrac{\dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \overset{\phi}{\vdash A,D,\Delta}}{\vdash A, C\otimes D,\Gamma,\Delta}\,(\otimes) \qquad \dfrac{\dfrac{}{\vdash A^{\perp},A}\,(ax) \qquad \overset{\tau}{\vdash B,\Sigma}}{\vdash A^{\perp}, A\otimes B,\Sigma}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(cut)$$

$$\overline{\beta}\swarrow \qquad\qquad \searrow\overline{\beta}$$

$$\dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \dfrac{\overset{\phi}{\vdash A,D,\Delta} \qquad \overset{\tau}{\vdash B,\Sigma}}{\vdash A\otimes B, D,\Delta,\Sigma}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \overset{\phi}{\vdash A,D,\Delta}}{\vdash A, C\otimes D,\Gamma,\Delta}\,(\otimes) \qquad \overset{\tau}{\vdash B,\Sigma}}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A,C,\Gamma} \qquad \overset{\tau}{\vdash D,\Sigma}}{\vdash A, C\otimes D,\Gamma,\Sigma}\,(\otimes) \qquad \dfrac{\dfrac{}{\vdash A^{\perp},A}\,(ax) \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A^{\perp}, A\otimes B,\Delta}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(cut)$$

$$\overline{\beta}\swarrow \qquad\qquad \searrow\overline{\beta}$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A,C,\Gamma} \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B, C,\Gamma,\Delta}\,(\otimes) \qquad \overset{\tau}{\vdash D,\Sigma}}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A,C,\Gamma} \qquad \overset{\tau}{\vdash D,\Sigma}}{\vdash A, C\otimes D,\Gamma,\Sigma}\,(\otimes) \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \dfrac{}{\vdash D^{\perp},D}\,(ax)}{\vdash D^{\perp}, C\otimes D,\Gamma}\,(\otimes) \qquad \dfrac{\overset{\phi}{\vdash A,\Delta} \qquad \overset{\tau}{\vdash B,D,\Sigma}}{\vdash A\otimes B, D,\Delta,\Sigma}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(cut)$$

$$\overline{\beta}\swarrow \qquad\qquad \searrow\overline{\beta}$$

$$\dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \dfrac{\overset{\phi}{\vdash A,\Delta} \qquad \overset{\tau}{\vdash B,D,\Sigma}}{\vdash A\otimes B, D,\Gamma,\Sigma}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes) \qquad\qquad \dfrac{\overset{\phi}{\vdash A,\Delta} \qquad \dfrac{\overset{\pi}{\vdash C,\Gamma} \qquad \overset{\tau}{\vdash B,D,\Sigma}}{\vdash B, C\otimes D,\Delta,\Sigma}\,(\otimes)}{\vdash A\otimes B, C\otimes D,\Gamma,\Delta,\Sigma}\,(\otimes)$$

- $C^{\otimes}_{\perp}$ *and* $C^{\perp}_{\otimes}$ *commutations*

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1,\bot}\,(\bot) \qquad \dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A,\bot,\Gamma}\,(\bot) \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\otimes)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(cut)$$

$$\swarrow \beta \qquad\qquad \beta \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B,\Gamma,\Delta}\,(\otimes)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\bot) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma}}{\vdash A,\bot,\Gamma}\,(\bot) \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\otimes)$$

$$\dfrac{\dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1,\bot}\,(\bot) \qquad \dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \dfrac{\overset{\phi}{\vdash B,\Delta}}{\vdash B,\bot,\Delta}\,(\bot)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\otimes)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(cut)$$

$$\swarrow \beta \qquad\qquad \beta \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \overset{\phi}{\vdash B,\Delta}}{\vdash A\otimes B,\Gamma,\Delta}\,(\otimes)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\bot) \qquad\qquad \dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \dfrac{\overset{\phi}{\vdash B,\Delta}}{\vdash B,\bot,\Delta}\,(\bot)}{\vdash A\otimes B,\bot,\Gamma,\Delta}\,(\otimes)$$

- $C^{\otimes}_{\&}$ *and* $C^{\&}_{\otimes}$ *commutations*

$$\dfrac{\dfrac{\overset{\phi}{\vdash B,C,\Delta} \qquad \overset{\tau}{\vdash B,D,\Delta}}{\vdash B,C\,\&\,D,\Delta}\,(\&) \qquad \dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \dfrac{}{\vdash B^{\perp},B}\,(ax)}{\vdash B^{\perp},A\otimes B,\Gamma}\,(\otimes)}{\vdash A\otimes B,C\,\&\,D,\Gamma,\Delta}\,(cut)$$

$$\swarrow \beta \qquad\qquad \beta \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \overset{\phi}{\vdash B,C,\Delta}}{\vdash A\otimes B,C,\Gamma,\Delta}\,(\otimes) \qquad \dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \overset{\tau}{\vdash B,D,\Delta}}{\vdash A\otimes B,D,\Gamma,\Delta}\,(\otimes)}{\vdash A\otimes B,C\,\&\,D,\Gamma,\Delta}\,(\&) \qquad\qquad \dfrac{\overset{\pi}{\vdash A,\Gamma} \qquad \dfrac{\overset{\phi}{\vdash B,C,\Delta} \qquad \overset{\tau}{\vdash B,D,\Delta}}{\vdash B,C\,\&\,D,\Delta}\,(\&)}{\vdash A\otimes B,C\,\&\,D,\Gamma,\Delta}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\tau}{\vdash A, D, \Gamma}}{\vdash A, C \mathbin{\&} D, \Gamma}\ (\&) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\ (ax) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A^{\perp}, A \otimes B, \Delta}\ (\otimes)}{\vdash A \otimes B, C \mathbin{\&} D, \Gamma, \Delta}\ (cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\ (\otimes) \qquad \dfrac{\overset{\tau}{\vdash A, D, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, D, \Gamma, \Delta}\ (\otimes)}{\vdash A \otimes B, C \mathbin{\&} D, \Gamma, \Delta}\ (\&) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\tau}{\vdash A, D, \Gamma}}{\vdash A, C \mathbin{\&} D, \Gamma}\ (\&) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \mathbin{\&} D, \Gamma, \Delta}\ (\otimes)$$

- $C^{\otimes}_{\oplus_i}$ *and* $C^{\oplus_i}_{\otimes}$ *commutations* We consider by symmetry only the $\oplus_1$-rule.

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\ (\oplus_1) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\ (ax) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A^{\perp}, A \otimes B, \Delta}\ (\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\ (\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (\oplus_1) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\ (\oplus_1) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (\otimes)$$

$$\dfrac{\dfrac{\overset{\phi}{\vdash B, C, \Delta}}{\vdash B, C \oplus D, \Delta}\ (\oplus_1) \qquad \dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{}{\vdash B^{\perp}, B}\ (ax)}{\vdash B^{\perp}, A \otimes B, \Gamma}\ (\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, C, \Delta}}{\vdash A \otimes B, C, \Gamma, \Delta}\ (\otimes)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (\oplus_1) \qquad\qquad \dfrac{\overset{\pi}{\vdash A, C, \Gamma} \qquad \dfrac{\overset{\phi}{\vdash B, \Delta}}{\vdash B, C \oplus D, \Delta}\ (\oplus_1)}{\vdash A \otimes B, C \oplus D, \Gamma, \Delta}\ (\otimes)$$

- $C^{\otimes}_{\top}$ *and* $C^{\top}_{\otimes}$ *commutations*

$$\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \overset{\pi}{\vdash B, \Delta}}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\top) \qquad\qquad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \qquad \overset{\pi}{\vdash B, \Delta}}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)$$

$$\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{}{\vdash B, \top, \Delta}\,(\top)}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\top) \qquad\qquad \dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \dfrac{}{\vdash B, \top, \Delta}\,(\top)}{\vdash A \otimes B, \top, \Gamma, \Delta}\,(\otimes)$$

- $C_{\otimes}^{mix_2}$ *and* $C_{mix_2}^{\otimes}$ *commutations*

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\tau}{\vdash \Sigma}}{\vdash A, \Gamma, \Sigma}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A^{\perp}, A \otimes B, \Delta}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \qquad \overset{\tau}{\vdash \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\tau}{\vdash \Sigma}}{\vdash A, \Gamma, \Sigma}\,(mix_2) \qquad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax) \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A^{\perp}, A \otimes B, \Sigma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(cut)$$

$$\swarrow_{\overline{\beta}} \qquad\qquad \searrow^{\overline{\beta}}$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\phi}{\vdash B, \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash B, \Delta, \Sigma}\,(mix_2) \qquad \dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{}{\vdash B^{\perp}, B}\,(ax)}{\vdash B^{\perp}, A \otimes B, \Gamma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(cut)$$

$$\swarrow_{\overline{\beta}} \qquad\qquad \searrow^{\overline{\beta}}$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \overset{\phi}{\vdash B, \Delta}}{\vdash A \otimes B, \Gamma, \Delta}\,(\otimes) \quad \overset{\tau}{\vdash \Sigma}}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2) \qquad\qquad \dfrac{\overset{\pi}{\vdash A, \Gamma} \quad \dfrac{\overset{\phi}{\vdash B, \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash B, \Delta, \Sigma}\,(mix_2)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash B, \Gamma, \Sigma}\,(mix_2) \qquad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \dfrac{}{\vdash B^{\perp}, B}\,(ax)}{\vdash B^{\perp}, A \otimes B, \Delta}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(cut)$$

$$\swarrow_{\overline{\beta}} \qquad\qquad \searrow^{\overline{\beta}}$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash A \otimes B, \Delta, \Sigma}\,(\otimes)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(mix_2) \qquad\qquad \dfrac{\overset{\phi}{\vdash A, \Delta} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash B, \Sigma}}{\vdash B, \Gamma, \Sigma}\,(mix_2)}{\vdash A \otimes B, \Gamma, \Delta, \Sigma}\,(\otimes)$$

- $C^{\perp}_{\perp}$ *commutation*

$$
\begin{array}{ccc}
& \dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,(\bot)}{\vdash \bot, \bot_2, \Gamma}\,(\bot_2) \qquad \dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1, \bot_1}\,(\bot_1)}{\vdash \bot_1, \bot_2, \Gamma}\,(cut) & \\
\overline{\beta}\swarrow & & \searrow\overline{\beta} \\
\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot_1, \Gamma}\,(\bot_1)}{\vdash \bot_1, \bot_2, \Gamma}\,(\bot_2) & & \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot_2, \Gamma}\,(\bot_2)}{\vdash \bot_1, \bot_2, \Gamma}\,(\bot_1)
\end{array}
$$

- $C_{\bot}^{\&}$ *and* $C_{\&}^{\bot}$ *commutations*

$$
\begin{array}{ccc}
& \dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot) \qquad \dfrac{\overset{\phi}{\vdash B, \Gamma}}{\vdash B, \bot, \Gamma}\,(\bot)}{\vdash A \,\&\, B, \bot, \Gamma}\,(\&) \qquad \dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1, \bot}\,(\bot)}{\vdash A \,\&\, B, \bot, \Gamma}\,(cut) & \\
\overline{\beta}\swarrow & & \searrow\overline{\beta} \\
\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot) \qquad \dfrac{\overset{\phi}{\vdash B, \Gamma}}{\vdash B, \bot, \Gamma}\,(\bot)}{\vdash A \,\&\, B, \bot, \Gamma}\,(\&) & & \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash B, \Gamma}}{\vdash A \,\&\, B, \Gamma}\,(\&)}{\vdash A \,\&\, B, \bot, \Gamma}\,(\bot)
\end{array}
$$

- $C_{\bot}^{\oplus_i}$ *and* $C_{\oplus_i}^{\bot}$ *commutations.* We consider by symmetry only the $\oplus_1$-rule.

$$
\begin{array}{ccc}
& \dfrac{\dfrac{\dfrac{}{\vdash A^{\bot}, A}\,(ax)}{\vdash A^{\bot}, A \oplus B}\,(\oplus_1) \qquad \dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(cut) & \\
\overline{\beta}\swarrow & & \searrow\overline{\beta} \\
\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A, \bot, \Gamma}\,(\bot)}{\vdash A \oplus B, \bot, \Gamma}\,(\oplus_1) & & \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, \bot, \Gamma}\,(\bot)
\end{array}
$$

- $C_{\top}^{\bot}$ *and* $C_{\bot}^{\top}$ *commutations*

$$\begin{array}{ccc} & \dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\dfrac{}{\vdash \top,\Gamma}\,(\top)}{\vdash \top,\bot,\Gamma}\,(\bot)}{\vdash \top,\bot,\Gamma}\,(cut) & \\ \swarrow{\scriptstyle\overline{\beta}} & & \searrow{\scriptstyle\overline{\beta}} \\ \dfrac{}{\vdash \top,\bot,\Gamma}\,(\top) & & \dfrac{\dfrac{}{\vdash \top,\Gamma}\,(\top)}{\vdash \top,\bot,\Gamma}\,(\bot) \end{array}$$

- $C_{\bot}^{mix_2}$ *and* $C_{mix_2}^{\bot}$ *commutations*

$$\begin{array}{ccc} & \dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot,\Gamma}\,(\bot) \qquad \overset{\phi}{\vdash \Delta}}{\vdash \bot,\Gamma,\Delta}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1,\bot}\,(\bot)}{\vdash \bot,\Gamma,\Delta}\,(cut) & \\ \swarrow{\scriptstyle\overline{\beta}} & & \searrow{\scriptstyle\overline{\beta}} \\ \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot,\Gamma}\,(\bot) \qquad \overset{\phi}{\vdash \Delta}}{\vdash \bot,\Gamma,\Delta}\,(mix_2) & & \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma,\Delta}\,(mix_2)}{\vdash \bot,\Gamma,\Delta}\,(\bot) \end{array}$$

$$\begin{array}{ccc} & \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash \bot,\Delta}\,(\bot)}{\vdash \bot,\Gamma,\Delta}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash 1}\,(1)}{\vdash 1,\bot}\,(\bot)}{\vdash \bot,\Gamma,\Delta}\,(cut) & \\ \swarrow{\scriptstyle\overline{\beta}} & & \searrow{\scriptstyle\overline{\beta}} \\ \dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash \bot,\Delta}\,(\bot)}{\vdash \bot,\Gamma,\Delta}\,(mix_2) & & \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma,\Delta}\,(mix_2)}{\vdash \bot,\Gamma,\Delta}\,(\bot) \end{array}$$

- $C_{\&}^{\&}$ *commutation.*

$$
\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \,\&\, B, C, \Gamma}\,(\&) \quad \dfrac{\overset{\tau}{\vdash A, D, \Gamma} \quad \overset{\mu}{\vdash B, D, \Gamma}}{\vdash A \,\&\, B, D, \Gamma}\,(\&)}{\vdash A \,\&\, B, C \,\&\, D, \Gamma}\,(\&) \quad \dfrac{\dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash B \oplus A^{\perp}, A}\,(\oplus_2) \quad \dfrac{\dfrac{}{\vdash B^{\perp}, B}\,(ax)}{\vdash B \oplus A^{\perp}, B}\,(\oplus_1)}{\vdash B^{\perp} \oplus A^{\perp}, A \,\&\, B}\,(\&)}{\vdash A \,\&\, B, C \,\&\, D, \Gamma}\,(cut)
$$

$$
\xrightarrow{\overline{\beta}}
\quad
\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \,\&\, B, C, \Gamma}\,(\&) \quad \dfrac{\overset{\tau}{\vdash A, D, \Gamma} \quad \overset{\mu}{\vdash B, D, \Gamma}}{\vdash A \,\&\, B, D, \Gamma}\,(\&)}{\vdash A \,\&\, B, C \,\&\, D, \Gamma}\,(\&)
$$

$$
\xrightarrow{\overline{\beta}}
\quad
\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\tau}{\vdash A, D, \Gamma}}{\vdash A, C \,\&\, D, \Gamma}\,(\&) \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma} \quad \overset{\mu}{\vdash B, D, \Gamma}}{\vdash B, C \,\&\, D, \Gamma}\,(\&)}{\vdash A \,\&\, B, C \,\&\, D, \Gamma}\,(\&)
$$

- $C^{\&}_{\oplus_i}$ *and* $C^{\oplus_i}_{\&}$ *commutations* We consider by symmetry only the $\oplus_1$-rule.

$$
\dfrac{\dfrac{\dfrac{}{\vdash C^{\perp}, C}\,(ax)}{\vdash C^{\perp}, C \oplus D}\,(\oplus_1) \quad \dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \,\&\, B, C, \Gamma}\,(\&)}{\vdash A \,\&\, B, C \oplus D, \Gamma}\,(cut)
$$

$$
\xrightarrow{\overline{\beta}}
\quad
\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma} \quad \overset{\phi}{\vdash B, C, \Gamma}}{\vdash A \,\&\, B, C, \Gamma}\,(\&)}{\vdash A \,\&\, B, C \oplus D, \Gamma}\,(\oplus_1)
\qquad
\xrightarrow{\overline{\beta}}
\quad
\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_1) \quad \dfrac{\overset{\phi}{\vdash B, C, \Gamma}}{\vdash B, C \oplus D, \Gamma}\,(\oplus_1)}{\vdash A \,\&\, B, C \oplus D, \Gamma}\,(\&)
$$

- $C^{\&}_{\top}$ *and* $C^{\top}_{\&}$ *commutations.*

$$
\dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \quad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \quad \dfrac{}{\vdash B, \top, \Gamma}\,(\top)}{\vdash A \,\&\, B, \top, \Gamma}\,(\&)}{\vdash A \,\&\, B, \top, \Gamma}\,(cut)
$$

$$
\xrightarrow{\overline{\beta}}
\quad
\dfrac{}{\vdash A \,\&\, B, \top, \Gamma}\,(\top)
\qquad
\xrightarrow{\overline{\beta}}
\quad
\dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top) \quad \dfrac{}{\vdash B, \top, \Gamma}\,(\top)}{\vdash A \,\&\, B, \top, \Gamma}\,(\&)
$$

- $C^{mix_2}_{\&}$ *and* $C^{\&}_{mix_2}$ *commutations*

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, \Delta} \qquad \overset{\tau}{\vdash B, \Delta}}{\vdash A \,\&\, B, \Delta}\,(\&)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\dfrac{\dfrac{}{\vdash A, A^{\perp}}\,(ax)}{\vdash A, B^{\perp} \oplus A^{\perp}}\,(\oplus_2) \qquad \dfrac{\dfrac{}{\vdash B, B^{\perp}}\,(ax)}{\vdash B, B^{\perp} \oplus A^{\perp}}\,(\oplus_1)}{\vdash A \,\&\, B, B^{\perp} \oplus A^{\perp}}\,(\&)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, \Delta} \qquad \overset{\tau}{\vdash B, \Delta}}{\vdash A \,\&\, B, \Delta}\,(\&)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(mix_2) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\tau}{\vdash B, \Delta}}{\vdash B, \Gamma, \Delta}\,(mix_2)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(\&)$$

$$\dfrac{\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\tau}{\vdash B, \Gamma}}{\vdash A \,\&\, B, \Gamma}\,(\&) \qquad \overset{\phi}{\vdash \Delta}}{\vdash A \,\&\, B, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\dfrac{\dfrac{}{\vdash A, A^{\perp}}\,(ax)}{\vdash A, B^{\perp} \oplus A^{\perp}}\,(\oplus_2) \qquad \dfrac{\dfrac{}{\vdash B, B^{\perp}}\,(ax)}{\vdash B, B^{\perp} \oplus A^{\perp}}\,(\oplus_1)}{\vdash A \,\&\, B, B^{\perp} \oplus A^{\perp}}\,(\&)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\tau}{\vdash B, \Gamma}}{\vdash A \,\&\, B, \Gamma}\,(\&) \qquad \overset{\phi}{\vdash \Delta}}{\vdash A \,\&\, B, \Gamma, \Delta}\,(mix_2) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\overset{\tau}{\vdash B, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash B, \Gamma, \Delta}\,(mix_2)}{\vdash A \,\&\, B, \Gamma, \Delta}\,(\&)$$

- $C^{\oplus_i}_{\oplus_j}$ *commutation* We consider by symmetry only the $\oplus_1$-rule.

$$\dfrac{\dfrac{\dfrac{}{\vdash C^{\perp}, C}\,(ax)}{\vdash C^{\perp}, C \oplus D}\,(\oplus_1) \qquad \dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A \oplus B, C, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}\,(cut)$$

$$\swarrow \overline{\beta} \qquad\qquad \overline{\beta} \searrow$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A \oplus B, C, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}\,(\oplus_1) \qquad\qquad \dfrac{\dfrac{\overset{\pi}{\vdash A, C, \Gamma}}{\vdash A, C \oplus D, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, C \oplus D, \Gamma}\,(\oplus_1)$$

- $C^{\oplus_i}_{\top}$ *and* $C^{\top}_{\oplus_i}$ *commutations* We consider by symmetry only the $\oplus_1$-rule.

$$\dfrac{\dfrac{}{\vdash A \oplus B, \top, \Gamma}\,(\top)}{} \;\xleftarrow{\overline{\beta}}\; \dfrac{\dfrac{}{\vdash \top, 0}\,(\top) \qquad \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top)}{\vdash A \oplus B, \top, \Gamma}\,(\oplus_1)}{\vdash A \oplus B, \top, \Gamma}\,(cut) \;\xrightarrow{\overline{\beta}}\; \dfrac{\dfrac{}{\vdash A, \top, \Gamma}\,(\top)}{\vdash A \oplus B, \top, \Gamma}\,(\oplus_1)$$

- $C_{\oplus_i}^{mix_2}$ *and* $C_{mix_2}^{\oplus_i}$ *commutations* We consider only $\oplus_1$ by symmetry.

$$\dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma}}{\vdash A \oplus B, \Gamma}\,(\oplus_1) \qquad \overset{\phi}{\vdash \Delta}}{\vdash A \oplus B, \Gamma, \Delta}\,(mix_2) \;\xleftarrow{\overline{\beta}}\; \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash A^{\perp}, A \oplus B}\,(\oplus_1)}{\vdash A \oplus B, \Gamma, \Delta}\,(cut) \;\xrightarrow{\overline{\beta}}\; \dfrac{\dfrac{\overset{\pi}{\vdash A, \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash A \oplus B, \Gamma, \Delta}\,(\oplus_1)$$

$$\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash A, \Delta}}{\vdash A \oplus B, \Delta}\,(\oplus_1)}{\vdash A \oplus B, \Gamma, \Delta}\,(mix_2) \;\xleftarrow{\overline{\beta}}\; \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2) \qquad \dfrac{\dfrac{}{\vdash A^{\perp}, A}\,(ax)}{\vdash A^{\perp}, A \oplus B}\,(\oplus_1)}{\vdash A \oplus B, \Gamma, \Delta}\,(cut) \;\xrightarrow{\overline{\beta}}\; \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash A, \Delta}}{\vdash A, \Gamma, \Delta}\,(mix_2)}{\vdash A \oplus B, \Gamma, \Delta}\,(\oplus_1)$$

- $C_{\top}^{\top}$ *commutation*

$$\dfrac{}{\vdash \top_1, \top_2, \Gamma}\,(\top_1) \;\xleftarrow{\overline{\beta}}\; \dfrac{\dfrac{}{\vdash \top_1, 0}\,(\top_1) \qquad \dfrac{}{\vdash \top, \top_2, \Gamma}\,(\top_2)}{\vdash \top_1, \top_2, \Gamma}\,(cut) \;\xrightarrow{\overline{\beta}}\; \dfrac{}{\vdash \top_1, \top_2, \Gamma}\,(\top_2)$$

- $C_{\top}^{mix_2}$ *and* $C_{mix_2}^{\top}$ *commutations*

$$\dfrac{\dfrac{\dfrac{}{\vdash \top, \Gamma}(\top) \qquad \overset{\pi}{\vdash \Delta}}{\vdash \top, \Gamma, \Delta}(mix_2) \qquad \dfrac{}{\vdash 0, \top}(\top)}{\vdash \top, \Gamma, \Delta}(cut)$$

$$\xrightarrow{\overline{\beta}} \quad \dfrac{\dfrac{}{\vdash \top, \Gamma}(\top) \qquad \overset{\pi}{\vdash \Delta}}{\vdash \top, \Gamma, \Delta}(mix_2) \qquad\qquad \xrightarrow{\overline{\beta}} \quad \dfrac{}{\vdash \top, \Gamma, \Delta}(\top)$$

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{}{\vdash \top, \Delta}(\top)}{\vdash \top, \Gamma, \Delta}(mix_2) \qquad \dfrac{}{\vdash 0, \top}(\top)}{\vdash \top, \Gamma, \Delta}(cut)$$

$$\xrightarrow{\overline{\beta}} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{}{\vdash \top, \Delta}(\top)}{\vdash \top, \Gamma, \Delta}(mix_2) \qquad\qquad \xrightarrow{\overline{\beta}} \quad \dfrac{}{\vdash \top, \Gamma, \Delta}(\top)$$

- $C_{mix_2}^{mix_2}$ *commutations*

$$\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash \Delta}}{\vdash \bot, \Delta}(\bot)}{\vdash \bot, \Gamma, \Delta}(mix_2) \qquad \dfrac{\dfrac{}{\vdash 1}(1) \qquad \overset{\tau}{\vdash \Sigma}}{\vdash 1, \Sigma}(mix_2)}{\vdash \Gamma, \Delta, \Sigma}(cut)$$

$$\xrightarrow{\overline{\beta}} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \qquad \dfrac{\overset{\phi}{\vdash \Delta} \qquad \overset{\tau}{\vdash \Sigma}}{\vdash \Delta, \Sigma}(mix_2)}{\vdash \Gamma, \Delta, \Sigma}(mix_2) \qquad\qquad \xrightarrow{\overline{\beta}} \quad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \qquad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}(mix_2) \qquad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Delta, \Sigma}(mix_2)$$

$$
\begin{tikzcd}[column sep=tiny]
& \dfrac{\dfrac{\dfrac{}{\vdash 1}\,{\scriptstyle (1)} \quad \overset{\tau}{\vdash \Sigma}}{\vdash 1, \Sigma}\,{\scriptstyle (mix_2)} \qquad \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma}}{\vdash \bot, \Gamma}\,{\scriptstyle (\bot)} \quad \overset{\phi}{\vdash \Delta}}{\vdash \bot, \Gamma, \Delta}\,{\scriptstyle (mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (cut)} \arrow[dl, "\beta"'] \arrow[dr, "\beta"] & \\
\dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta}\,{\scriptstyle (mix_2)} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (mix_2)} & & \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Sigma}\,{\scriptstyle (mix_2)} \quad \overset{\phi}{\vdash \Delta}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (mix_2)}
\end{tikzcd}
$$

$$
\begin{tikzcd}[column sep=tiny]
& \dfrac{\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{}{\vdash 1}\,{\scriptstyle (1)}}{\vdash 1, \Gamma}\,{\scriptstyle (mix_2)} \qquad \dfrac{\overset{\phi}{\vdash \Delta} \quad \dfrac{\overset{\tau}{\vdash \Sigma}}{\vdash \bot, \Sigma}\,{\scriptstyle (\bot)}}{\vdash \bot, \Delta, \Sigma}\,{\scriptstyle (mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (cut)} \arrow[dl, "\beta"'] \arrow[dr, "\beta"] & \\
\dfrac{\overset{\pi}{\vdash \Gamma} \quad \dfrac{\overset{\phi}{\vdash \Delta} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Delta, \Sigma}\,{\scriptstyle (mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (mix_2)} & & \dfrac{\overset{\phi}{\vdash \Delta} \quad \dfrac{\overset{\pi}{\vdash \Gamma} \quad \overset{\tau}{\vdash \Sigma}}{\vdash \Gamma, \Sigma}\,{\scriptstyle (mix_2)}}{\vdash \Gamma, \Delta, \Sigma}\,{\scriptstyle (mix_2)}
\end{tikzcd}
$$

□

Concerning rule commutations for the full system, a devilishly motivated reader could prove their inclusion in β-equality following the same schemes as for the above rule commutations, *e.g.*:

$$
\begin{tikzcd}[column sep=tiny]
& \dfrac{\dfrac{\dfrac{}{\vdash \Gamma}\,{\scriptstyle (\emptyset)}}{\vdash \bot, \Gamma}\,{\scriptstyle (\bot)} \qquad \dfrac{\dfrac{}{\vdash 1}\,{\scriptstyle (1)} \quad \dfrac{}{\vdash 1}\,{\scriptstyle (1)}}{\vdash 1}\,{\scriptstyle (\cup)}}{\vdash \Gamma}\,{\scriptstyle (cut)} \arrow[dl, "\beta"'] \arrow[dr, "\beta"] & \\
\dfrac{}{\vdash \Gamma}\,{\scriptstyle (\emptyset)} & & \dfrac{\dfrac{}{\vdash \Gamma}\,{\scriptstyle (\emptyset)} \quad \dfrac{}{\vdash \Gamma}\,{\scriptstyle (\emptyset)}}{\vdash \Gamma}\,{\scriptstyle (\cup)}
\end{tikzcd}
$$

Appendix B

Basic properties of Isomorphisms and Retractions

We prove in this appendix properties of isomorphisms and retractions used implicitly in Part III. In particular, that these relations are transitive, respect substitution and contextualization, so that an (in)equational theory is a legal tool for their characterization.

We recall the definition of isomorphisms and retractions, from Chapters 6 and 7

Definition 6.1 (Isomorphism)**.** Consider (a sub-system of) linear logic with $\overset{o}{\vdash\!\dashv}$ some (possibly none, or all) of the Rétoré transformations. Two formulas A and B are **isomorphic** (in this sub-system), denoted $A \simeq B$, if there exist proofs π of $\vdash A^\perp, B$ and π' of $\vdash B^\perp, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A$ and $\pi' \overset{A}{\bowtie} \pi =_{\beta\eta o} ax_B$:

$$\pi \overset{B}{\bowtie} \pi' \;=\; \dfrac{\overset{\pi}{\vdash A^\perp, B} \qquad \overset{\pi'}{\vdash B^\perp, A}}{\vdash A^\perp, A}\,{\scriptstyle (cut)} \;=_{\beta\eta o}\; \dfrac{}{\vdash A^\perp, A}\,{\scriptstyle (ax)} \;=\; ax_A$$

and

$$\pi' \overset{A}{\bowtie} \pi \;=\; \dfrac{\overset{\pi'}{\vdash B^\perp, A} \qquad \overset{\pi}{\vdash A^\perp, B}}{\vdash B^\perp, B}\,{\scriptstyle (cut)} \;=_{\beta\eta o}\; \dfrac{}{\vdash B^\perp, B}\,{\scriptstyle (ax)} \;=\; ax_B$$

Definition 7.1 (Retraction)**.** Consider (a sub-system of) linear logic with $\overset{o}{\vdash\!\dashv}$ some (possibly none, or all) of the Rétoré transformations. There is a **retraction** between formulas A and B, denoted $A \trianglelefteq B$, if there exist proofs π of $\vdash A^\perp, B$ and π' of $\vdash B^\perp, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta o} ax_A$:

$$\pi \overset{B}{\bowtie} \pi' \;=\; \dfrac{\overset{\pi}{\vdash A^\perp, B} \qquad \overset{\pi'}{\vdash B^\perp, A}}{\vdash A^\perp, A}\,{\scriptstyle (cut)} \;=_{\beta\eta o}\; \dfrac{}{\vdash A^\perp, A}\,{\scriptstyle (ax)} \;=\; ax_A$$

The two previous definitions correspond to instantiations of the categorical concepts, in the category corresponding to Linear Logic, whose objects are formulas and whose morphisms are proofs up to $=_{\beta\eta}$ [See89]. In both of these definitions, the focus is on objects, not on the underlying morphisms.

The relation "being a retraction of", namely $\trianglelefteq$, is reflexive and transitive. The last property allows in particular to consider retractions up to isomorphisms. Meanwhile, the relation "being isomorphic to", $\simeq$, is an equivalence relation.

These properties hold not only in linear logic, but more generally in any logic whose retractions and isomorphisms are defined as in Definitions 7.1 and 6.1 and having a $cut - cut$ commutative case and an ax key case in its cut-elimination procedure.

Lemma B.1. *Take $\pi_\leftarrow \vdash A^\perp, B$, $\pi_\rightarrow \vdash B^\perp, A$, $\phi_\leftarrow \vdash B^\perp, C$ and $\phi_\rightarrow \vdash C^\perp, B$ proofs such that $\pi_\leftarrow \overset{B}{\bowtie} \pi_\rightarrow =_{\beta\eta} ax_A$ and $\phi_\leftarrow \overset{C}{\bowtie} \phi_\rightarrow =_{\beta\eta} ax_B$.*

Set $\tau_\leftarrow = \pi_\leftarrow \overset{B}{\bowtie} \phi \leftarrow$ and $\tau_\rightarrow = \phi_\rightarrow \overset{B}{\bowtie} \pi_\rightarrow$, respectively proofs of $\vdash A^\perp, C$ and $\vdash C^\perp, A$.

Then $\tau_\leftarrow \overset{C}{\bowtie} \tau_\rightarrow =_{\beta\eta} ax_A$.

Proof.

$$\dfrac{\overset{\tau_\leftarrow}{\vdash A^\perp, C} \qquad \overset{\tau_\rightarrow}{\vdash C^\perp, A}}{\vdash A^\perp, A}\,{\scriptstyle (cut)} \quad = \quad \dfrac{\dfrac{\overset{\pi_\leftarrow}{\vdash A^\perp, B} \qquad \overset{\phi_\leftarrow}{\vdash B^\perp, C}}{\vdash A^\perp, C}\,{\scriptstyle (cut)} \qquad \dfrac{\overset{\phi_\rightarrow}{\vdash C^\perp, B} \qquad \overset{\pi_\rightarrow}{\vdash B^\perp, A}}{\vdash C^\perp, A}\,{\scriptstyle (cut)}}{\vdash A^\perp, A}\,{\scriptstyle (cut)}$$

$\xrightarrow{\beta}$ (using a $cut - cut$ commutative case)

$$\dfrac{\overset{\pi_\leftarrow}{\vdash A^\perp, B} \qquad \dfrac{\overset{\phi_\leftarrow}{\vdash B^\perp, C} \qquad \dfrac{\overset{\phi_\rightarrow}{\vdash C^\perp, B} \qquad \overset{\pi_\rightarrow}{\vdash B^\perp, A}}{\vdash C^\perp, A}\,{\scriptstyle (cut)}}{\vdash B^\perp, A}\,{\scriptstyle (cut)}}{\vdash A^\perp, A}\,{\scriptstyle (cut)}$$

$\xrightarrow{\beta}$ (using a $cut - cut$ commutative case)

$$\dfrac{\overset{\pi_\leftarrow}{\vdash A^\perp, B} \qquad \dfrac{\dfrac{\overset{\phi_\leftarrow}{\vdash B^\perp, C} \qquad \overset{\phi_\rightarrow}{\vdash C^\perp, B}}{\vdash B^\perp, B}\,{\scriptstyle (cut)} \qquad \overset{\pi_\rightarrow}{\vdash B^\perp, A}}{\vdash B^\perp, A}\,{\scriptstyle (cut)}}{\vdash A^\perp, A}\,{\scriptstyle (cut)}$$

$$=_{\beta\eta} \quad \dfrac{\overset{\pi_\leftarrow}{\vdash A^\perp, B} \qquad \dfrac{\dfrac{}{\vdash B^\perp, B}\,{\scriptstyle (ax)} \qquad \overset{\pi_\rightarrow}{\vdash B^\perp, A}}{\vdash B^\perp, A}\,{\scriptstyle (cut)}}{\vdash A^\perp, A}\,{\scriptstyle (cut)}$$

$\xrightarrow{\beta}$ (using an ax key case)

$$\dfrac{\overset{\pi_\leftarrow}{\vdash A^\perp, B} \qquad \overset{\pi_\rightarrow}{\vdash B^\perp, A}}{\vdash A^\perp, A}\,{\scriptstyle (cut)} \quad =_{\beta\eta} \quad \dfrac{}{\vdash A^\perp, A}\,{\scriptstyle (ax)}$$

$\square$

Lemma B.2. *The relation $\trianglelefteq$ is reflexive and transitive.*

Proof. Reflexivity is the easiest to prove: just consider $\pi = ax_A = \pi'$, given a formula A. Then $ax_A \overset{A}{\bowtie} ax_A \xrightarrow{\beta} ax_A$ using an ax key case. This proves $A \trianglelefteq A$.

Now, let us prove transitivity, *i.e.* assuming $A \trianglelefteq B \trianglelefteq C$, let us show $A \trianglelefteq C$. Take $\pi_{\leftarrow} \vdash A^{\perp}, B$, $\pi_{\rightarrow} \vdash B^{\perp}, A$, $\phi_{\leftarrow} \vdash B^{\perp}, C$ and $\phi_{\rightarrow} \vdash C^{\perp}, B$ proofs such that $\pi_{\leftarrow} \overset{B}{\bowtie} \pi_{\rightarrow} =_{\beta\eta} ax_A$ and $\phi_{\leftarrow} \overset{C}{\bowtie} \phi_{\rightarrow} =_{\beta\eta} ax_B$.

By Lemma B.1, $\tau_{\leftarrow} = \pi_{\leftarrow} \overset{B}{\bowtie} \phi_{\leftarrow}$ and $\tau_{\rightarrow} = \phi_{\rightarrow} \overset{B}{\bowtie} \pi_{\rightarrow}$ yield $A \trianglelefteq C$. □

Lemma B.3. *The relation $\simeq$ is an equivalence relation.*

Proof. Reflexivity follows as in the proof of Lemma B.2: given a formula A, $ax_A \overset{A}{\bowtie} ax_A \xrightarrow{\beta} ax_A$ using an ax key case. This proves $A \simeq A$.

For symmetry, take π and π' respectively of $\vdash A^{\perp}, B$ and $\vdash B^{\perp}, A$ such that $\pi \overset{B}{\bowtie} \pi' =_{\beta\eta} ax_A$ and $\pi' \overset{A}{\bowtie} \pi =_{\beta\eta} ax_B$. Then, these proofs not only yield $A \simeq B$ but also $B \simeq A$.

Lastly, transitivity follows from Lemma B.1, as in the proof of Lemma B.2. Indeed, take $\pi_{\leftarrow} \vdash A^{\perp}, B$, $\pi_{\rightarrow} \vdash B^{\perp}, A$, $\phi_{\leftarrow} \vdash B^{\perp}, C$ and $\phi_{\rightarrow} \vdash C^{\perp}, B$ proofs such that $\pi_{\leftarrow} \overset{B}{\bowtie} \pi_{\rightarrow} =_{\beta\eta} ax_A$, $\pi_{\rightarrow} \overset{A}{\bowtie} \pi_{\leftarrow} =_{\beta\eta} ax_B$, $\phi_{\leftarrow} \overset{C}{\bowtie} \phi_{\rightarrow} =_{\beta\eta} ax_B$ and $\phi_{\rightarrow} \overset{B}{\bowtie} \phi_{\leftarrow} =_{\beta\eta} ax_C$. According to Lemma B.1, with $\tau_{\leftarrow} = \pi_{\leftarrow} \overset{B}{\bowtie} \phi_{\leftarrow}$ and $\tau_{\rightarrow} = \phi_{\rightarrow} \overset{B}{\bowtie} \pi_{\rightarrow}$, we have $\tau_{\leftarrow} \overset{C}{\bowtie} \tau_{\rightarrow} =_{\beta\eta} ax_A$ and $\tau_{\rightarrow} \overset{A}{\bowtie} \tau_{\leftarrow} =_{\beta\eta} ax_C$. □

Lemma B.4.

- *The relation $\trianglelefteq$ is contextual: if $A \trianglelefteq B$ then for any formula C, $C[A/X] \trianglelefteq C[B/X]$.*
- *The relation $\simeq$ is contextual: if $A \simeq B$ then for any formula C, $C[A/X] \simeq C[B/X]$ (in other words, $\simeq$ is a congruence).*

Proof. By induction on the formula C. The idea is to do as in the identity on C until reaching X, where one replaces the ax-rules on X by proofs associated to the retraction or isomorphism. Then, one has to check that reducing cuts can be done as wished. □

Lemma B.5. *Let σ be a substitution acting only on atoms and A a formula. Then $\sigma(ax_A) = ax_{\sigma(A)}$ and $\sigma(\mathrm{id}_A) =_{\eta} \mathrm{id}_{\sigma(A)}$. Moreover, if σ is a* atomic-substitution*, then $\sigma(\mathrm{id}_A) = \mathrm{id}_{\sigma(A)}$.*

Proof. The first statement is easy:

$$\sigma(ax_A) = \sigma\left(\cfrac{}{\vdash A^{\perp}, A}\ (ax)\right) = \cfrac{}{\vdash \sigma(A)^{\perp}, \sigma(A)}\ (ax) = ax_{\sigma(A)}$$

using that $\sigma(A^{\perp}) = \sigma(A)^{\perp}$ (Fact 1.5).

For the second statement, in the case of an atomic-substitution, remark that $\sigma(\mathrm{id}_A)$ has the same rules than id_A, and as we replaced some signed atoms by other signed atoms, all rules can still be applied except maybe ax-rules. But, again using (Fact 1.5), there is no problem for ax-rules. We prove the wished result by induction on A. For the general case, the obtained proof may not be atomic-axiom, so we need some $\xrightarrow{\eta}$ steps to recover $\mathrm{id}_{\sigma(A)}$. □

Lemma B.6. *Take σ a substitution acting only on atoms and two formulas A and B.*

- $A \trianglelefteq B \implies \sigma(A) \trianglelefteq \sigma(B)$

- $A \simeq B \implies \sigma(A) \simeq \sigma(B)$

Proof. Because such a substitution can be applied on proofs, and the same steps of axiom-expansion and cut-elimination can be applied on the obtained proofs, for any formulas can be made to act as an atom. We conclude by Lemma B.5, to get that the proof(s) obtained by the image of the $\xrightarrow{\beta}$ and $\xrightarrow{\eta}$ steps through σ is (are) still η-equal to an identity. $\square$

Corollary B.7 (Retraction up to isomorphisms)**.**

$$A \simeq B \trianglelefteq C \simeq D \implies A \trianglelefteq D$$
$$A \simeq B \triangleleft C \simeq D \implies A \triangleleft D$$

Proof. Assuming $A \simeq B \trianglelefteq C \simeq D$, we have $A \trianglelefteq B \trianglelefteq C \trianglelefteq D$ for $\simeq \subseteq \trianglelefteq$ (Fact 7.3). By Lemma B.2, $A \trianglelefteq D$.

If $B \triangleleft C$, then by contradiction assume $A \simeq D$. Then $B \simeq A \simeq D \simeq C$ while $B \triangleleft C$, a contradiction. Hence $A \trianglelefteq D$ but $A \not\simeq D$, thus $A \triangleleft D$. $\square$

Appendix C

A long computation in Coq

Here is given the full Coq program taken as an example in Section 5.2.3 with a too long computational time, excepted in the case where type annotations are given. This is a transcript of the file bug_report_1_bis.v of the code available on https://github.com/RemiDiG/proofnet_mll (at least at the time this thesis is written).

```
Set Warnings "-notation-overridden".
From mathcomp Require Import all_ssreflect.
Set Warnings "notation-overridden".
From GraphTheory Require Import mgraph.

Set Implicit Arguments.
Unset Strict Implicit.
Set Default Timeout 60.

Section Test.

Variables (Lv Le : Type) (cut : Lv).

Definition extend_edge_graph (G : graph Lv Le) (e : edge G) (R : Lv) (As At : Le) :
  graph Lv Le :=
  remove_edges [set e : edge G] ∔ R ∔ [inl (source e), As, inr tt] ∔ [inr tt, At, inl (target e)].

Definition new_graph (G : graph Lv Le) (e : edge G) :=
  (@extend_edge_graph
    (@extend_edge_graph G e cut (elabel e) (elabel e))
    None cut (elabel e) (elabel e)).

Fail Time Definition transport_to_new (G : graph Lv Le) (e : edge G) :
  edge G → edge (new_graph e) :=
  fun a ⇒
    if @boolP (a \notin [set e]) is AltTrue p1 then
      if @boolP (Some (Some (inl (Sub a p1))) \notin [set None]) is AltTrue p2 then
        Some (Some (inl (Sub (Some (Some (inl (Sub a p1)))) p2)))
    else None else None.
```

```
Time Definition transport_to_new (G : graph Lv Le) (e : edge G) :
  edge G → edge (new_graph e) :=
  fun a ⇒
    if @boolP (a \notin [set e]) is AltTrue p1 then
      if @boolP
        ((Some (Some (inl (Sub a p1 : edge (remove_edges [set e : edge G])))) :
          edge (@extend_edge_graph G e cut (elabel e) (elabel e)))
          \notin [set None]) is AltTrue p2 then
            Some (Some (inl (Sub (Some (Some (inl (Sub a p1 :
              edge (remove_edges [set e : edge G])))) :
                edge (@extend_edge_graph G e cut (elabel e) (elabel e))) p2 :
                  edge (remove_edges [set None :
                    edge (@extend_edge_graph G e cut (elabel e) (elabel e))])))))
    else None else None.
End Test.
```

Bibliography

[Abr95] Michele Abrusci. “Noncommutative Proof Nets”. In: *Advances in Linear Logic*. Ed. by Jean-Yves Girard, Yves Lafont, and Laurent Regnier. Vol. 222. London Mathematical Society Lecture Note Series. Cambridge University Press, 1995, pp. 271–296.

[Acc13] Beniamino Accattoli. “Linear Logic and Strong Normalization”. In: *Rewriting Techniques and Applications*. Ed. by Femke van Raamsdonk. Vol. 21. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2013, pp. 39–54. DOI: 10.4230/LIPIcs.RTA.2013.39. URL: https://drops.dagstuhl.de/entities/document/10.4230/LIPIcs.RTA.2013.39.

[AJM00] Samson Abramsky, Radha Jagadeesan, and Pasquale Malacaria. “Full Abstraction for PCF”. In: *Information and Computation* 163.2 (Dec. 2000), pp. 409–470.

[AM19] Michele Abrusci and Roberto Maieli. “Proof nets for multiplicative cyclic linear logic and Lambek calculus”. In: *Mathematical Structures in Computer Science* 29 (2019), pp. 733–762. DOI: 10.1017/S0960129518000300.

[AM98] Michele Abrusci and Elena Maringelli. “A New Correctness Criterion for Cyclic Proof Nets”. In: *Journal of Logic, Language and Information* 7 (1998), pp. 449–459. DOI: 10.1023/A:1008354130493.

[And92] Jean-Marc Andreoli. “Logic Programming with Focusing Proofs in Linear Logic”. In: *Journal of Logic and Computation* 2.3 (1992), pp. 297–347.

[ARG21] Clément Allain, Gabriel Radanne, and Laure Gonnord. “Isomorphisms are back!” In: *ML 2021 - ML Workshop*. Aug. 2021. URL: https://hal.science/hal-03355381.

[AT12] Takahito Aoto and Yoshihito Toyama. “A Reduction-Preserving Completion for Proving Confluence of Non-Terminating Term Rewriting Systems”. In: *Logical Methods in Computer Science* 8.1 (Mar. 2012), pp. 1–29. DOI: 10.2168/LMCS-8(1:31)2012.

[BD99] Vincent Balat and Roberto Di Cosmo. “A Linear Logical View of Linear Type Isomorphisms”. In: *Computer Science Logic*. Ed. by Jörg Flum and Mario Rodríguez-Artalejo. Vol. 1683. Lecture Notes in Computer Science. Springer, 1999, pp. 250–265.

[Ber57] Claude Berge. “Two Theorems in Graph Theory”. In: *Proceedings of the National Academy of Sciences of the United States of America* 43.9 (Sept. 1957), pp. 842–844. ISSN: 00278424, 10916490. URL: https://www.jstor.org/stable/89875.

[BHS05] Richard Blute, Masahiro Hamano, and Philip Scott. "Softness of hypercoherences and MALL full completeness". In: *Annals of Pure and Applied Logic* 131.1 (2005), pp. 1–63. ISSN: 0168-0072. DOI: 10.1016/j.apal.2004.05.002. URL: https://www.sciencedirect.com/science/article/pii/S0168007204000855.

[Bie95] Gavin Bierman. "What is a Categorical Model of Intuitionistic Linear Logic?" In: *Typed Lambda Calculi and Applications '95*. Ed. by Mariangiola Dezani and Gordon Plotkin. Vol. 902. Lecture Notes in Computer Science. Springer, Apr. 1995. DOI: 10.1007/BFb0014046.

[BL85] Kim Bruce and Giuseppe Longo. "Provable isomorphisms and domain equations in models of typed languages". In: Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing, Providence, Rhode Island, USA. Association for Computing Machinery, 1985, pp. 263–272.

[Blu+96] Richard Blute, Robin Cockett, Robert Seely, and Todd Trimble. "Natural Deduction and Coherence for Weakly Distributive Categories". In: *Journal of Pure and Applied Algebra* 113 (1996), pp. 229–296.

[Brü04] Kai Brünnler. *Deep inference and symmetry in classical proofs*. Logos, 2004. ISBN: 3832504486.

[BV95] Gianluigi Bellin and Jacques Van de Wiele. "Subnets of Proof-nets in MLL−". In: *Advances in Linear Logic*. Ed. by Jean-Yves Girard, Yves Lafont, and Laurent Regnier. Vol. 222. London Mathematical Society Lecture Note Series. Cambridge University Press, 1995, pp. 249–270.

[Chu21] Florian Chudigiewitsch. "Computational Complexity of Deciding Provability in Linear Logic and its Fragments". MA thesis. 2021. URL: https://arxiv.org/abs/2110.00562.

[CL] Pierre Courtieu and Olivier Laurent. *ill_narratives*. Coq library. URL: https://github.com/Matafou/ill_narratives.

[CLR17] Kaustuv Chaudhuri, Leonardo Lima, and Giselle Reis. "Formalized Meta-Theory of Sequent Calculi for Substructural Logics". In: *Electronic Notes in Theoretical Computer Science* 332 (June 2017). LSFA 2016 - 11th Workshop on Logical and Semantic Frameworks with Applications (LSFA), pp. 57–73. DOI: 10.1016/j.entcs.2017.04.005. URL: https://www.sciencedirect.com/science/article/pii/S1571066117300154.

[CLR19] Kaustuv Chaudhuri, Leonardo Lima, and Giselle Reis. "Formalized meta-theory of sequent calculi for linear logics". In: *Theoretical Computer Science* 781 (2019). Code source at https://github.com/meta-logic/abella-reasoning, pp. 24–38. DOI: https://doi.org/10.1016/j.tcs.2019.02.023. URL: https://www.sciencedirect.com/science/article/pii/S030439751930129X.

[CMS08] Kaustuv Chaudhuri, Dale Miller, and Alexis Saurin. "Canonical Sequent Proofs via Multi-Focusing". In: *Fifth Ifip International Conference On Theoretical Computer Science*. Ed. by Giorgio Ausiello, Juhani Karhumäki, Giancarlo Mauri, and Luke Ong. Vol. 273. Theoretical Computer Science. Springer, 2008, pp. 383–396. ISBN: 978-0-387-09680-3. DOI: 10.1007/978-0-387-09680-3_26.

[Coq] The Coq Development Team. *The Coq Proof Assistant*. DOI: 10.5281/zenodo.1003420. URL: https://coq.inria.fr/.

[CP05] Robin Cockett and Craig Pastro. "A Language For Multiplicative-additive Linear Logic". In: *Electronic Notes in Theoretical Computer Science* 122 (2005). Proceedings of the 10th Conference on Category Theory in Computer Science (CTCS 2004), pp. 23–65. DOI: /10.1016/j.entcs.2004.06.049. URL: https://www.sciencedirect.com/science/article/pii/S1571066105000320.

[DF06] Paolo Di Giamberardino and Claudia Faggian. "Jump from Parallel to Sequential Proofs: Multiplicatives". In: *Computer Science Logic*. Vol. 4207. Lecture Notes in Computer Science. Springer, 2006, pp. 319–333. DOI: 10.1007/11874683_21.

[DF08] Paolo Di Giamberardino and Claudia Faggian. "Proof nets sequentialisation in multiplicative linear logic". In: *Annals of Pure and Applied Logic* 155.3 (Oct. 2008), pp. 173–182. DOI: 10.1016/j.apal.2008.04.002.

[DG99] Roberto Di Cosmo and Stefano Guerrini. "Strong Normalization of Proof Nets Modulo Structural Congruences". In: *Rewriting Techniques and Applications*. Ed. by P. Narendran and M. Rusinowitch. Vol. 1631. Lecture Notes in Computer Science. Springer, 1999, pp. 75–89. DOI: 10.1007/3-540-48685-2_6.

[Di+23] Rémi Di Guardia, Olivier Laurent, Lorenzo Tortora de Falco, and Lionel Vaux Auclair. "Sequentialization is as fun as bungee jumping". In: *Seventh International Workshop on Trends in Linear Logic and Applications 2023*. Ed. by Thomas Ehrhard, Stefano Guerrini, and Lorenzo Tortora de Falco. 2023.

[Di11] Paolo Di Giamberardino. "Jump from Parallel to Sequential Proofs: Additives". In: (2011).

[Di18] Paolo Di Giamberardino. "Jump from Parallel to Sequential Proofs: Exponentials". In: *Mathematical Structures in Computer Science* 28.7 (Aug. 2018), pp. 1204–1252. DOI: 10.1017/S0960129516000414.

[Di95] Roberto Di Cosmo. *Isomorphisms of Types*. Progress in Theoretical Computer Science. Birkhäuser, 1995.

[Die05] Reinhard Diestel. *Graph Theory*. Springer, 2005.

[DL22] Rémi Di Guardia and Olivier Laurent. "Bottom-Up Sequentialization of Unit-Free MALL Proof Nets". In: *Third Joint International Workshop on Linearity and Trends in Linear Logic and Applications 2022*. Ed. by Laurent Regnier and Daniel Ventura. EasyChair Preprint 8431. 2022.

[DL23] Rémi Di Guardia and Olivier Laurent. "Type Isomorphisms for Multiplicative-Additive Linear Logic". In: *International Conference on Formal Structures for Computation and Deduction (FSCD)*. Ed. by Marco Gaboardi and Femke van Raamsdonk. Vol. 260. Leibniz International Proceedings in Informatics (LIPIcs). Full version with proofs on https://hal.science/hal-04082204. Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, July 2023, 26:1–26:21. DOI: 10.4230/LIPIcs.FSCD.2023.26.

[DP18] Christian Doczkal and Damien Pous. *Graph Theory*. Coq library. 2018. URL: https://github.com/coq-community/graph-theory.

[DP20] Christian Doczkal and Damien Pous. "Graph Theory in Coq: Minors, Treewidth, and Isomorphisms". In: *Journal of Automated Reasoning* (2020). DOI: 10.1007/s10817-020-09543-2. URL: https://hal.science/hal-02316859.

[DP97] Kosta Dosen and Zoran Petric. "Isomorphic Objects in Symmetric Monoidal Closed Categories". In: *Mathematical Structures in Computer Science* 7.6 (1997), pp. 639–662. DOI: 10.1017/S0960129596002241.

[DPS21] Abhishek De, Luc Pellissier, and Alexis Saurin. "Canonical proof-objects for coinductive programming: infinets with infinitely many cuts". In: *PPDP 2021: 23rd International Symposium on Principles and Practice of Declarative Programming*. Tallinn, Estonia: Association for Computing Machinery, Sept. 2021, pp. 1–15. DOI: 10.1145/3479394.3479402. URL: https://hal.science/hal-03371935.

[dPS92] U. de'Liguoro, A. Piperno, and R. Statman. "Retracts in simply type lambda beta eta -calculus". In: *Proceedings of the seventh annual symposium on Logic In Computer Science*. IEEE. IEEE Computer Society Press, 1992, pp. 461–469. DOI: 10.1109/LICS.1992.185557.

[DR89] Vincent Danos and Laurent Regnier. "The structure of multiplicatives". In: *Archive for Mathematical Logic* 28 (1989), pp. 181–203. DOI: 10.1007/BF01622878.

[DR95] Vincent Danos and Laurent Regnier. "Proof-Nets and the Hilbert Space". In: *Advances in Linear Logic*. Ed. by Jean-Yves Girard, Yves Lafont, and Laurent Regnier. Vol. 222. London Mathematical Society Lecture Note Series. Cambridge University Press, 1995, pp. 307–328.

[Ehr18] Thomas Ehrhard. "An introduction to differential linear logic: proof-nets, models and antiderivatives". In: *Mathematical Structures in Computer Science* 28.7 (2018), pp. 995–1060. DOI: 10.1017/S0960129516000372.

[ER06] Thomas Ehrhard and Laurent Regnier. "Differential interaction nets". In: *Theoretical Computer Science* 364.2 (Nov. 2006), pp. 166–195.

[FDB02] Marcelo Fiore, Roberto Di Cosmo, and Vincent Balat. "Remarks on Isomorphisms in Typed Lambda Calculi with Empty and Sum Types". In: *Proceedings of the seventeenth annual symposium on Logic In Computer Science*. IEEE. Copenhagen: IEEE Computer Society Press, July 2002, pp. 147–156.

[FR94] Arnaud Fleury and Christian Retoré. "The Mix Rule". In: *Mathematical Structures in Computer Science* 4.2 (1994), pp. 273–285.

[Gal+22] Hortensia Galeana-Sánchez, Rocío Rojas-Monroy, Rocío Sánchez-López, and Juana Imelda Villarreal-Valdés. "H-Cycles in H-Colored Multigraphs". In: *Graphs and Combinatorics* 38.3 (June 2022), p. 62. DOI: 10.1007/s00373-022-02464-4.

[GH83] Jerrold W. Grossman and Roland Häggkvist. "Alternating Cycles in Edge-Partitioned Graphs". In: *Journal of Combinatorial Theory, Series B* 34.1 (1983), pp. 77–81. ISSN: 0095-8956. DOI: 10.1016/0095-8956(83)90008-4. URL: https://www.sciencedirect.com/science/article/pii/0095895683900084.

[Gir01] Jean-Yves Girard. "Locus Solum: From the rules of logic to the logic of rules". In: *Mathematical Structures in Computer Science* 11.3 (June 2001), pp. 301–506.

[Gir11] Jean-Yves Girard. *The Blind Spot*. European Mathematical Society Publishing House, 2011. ISBN: 9783037190883.

[Gir87] Jean-Yves Girard. "Linear logic". In: *Theoretical Computer Science* 50 (1987), pp. 1–102. DOI: 10.1016/0304-3975(87)90045-4.

[Gir91] Jean-Yves Girard. "Quantifiers in Linear Logic II". In: *Nuovi problemi della logica e della filosofia della scienza*. Ed. by Corsi and Sambin. Bologna: CLUEB, 1991, pp. 79–90.

[Gir96] Jean-Yves Girard. "Proof-nets: the parallel syntax for proof-theory". In: *Logic and Algebra*. Ed. by Aldo Ursini and Paolo Agliano. Vol. 180. Lecture Notes In Pure and Applied Mathematics. New York: Marcel Dekker, 1996, pp. 97–124. DOI: 10.1201/9780203748671-4.

[GLR95] Jean-Yves Girard, Yves Lafont, and Laurent Regnier, eds. *Advances in Linear Logic*. Vol. 222. London Mathematical Society Lecture Note Series. Cambridge University Press, 1995.

[GLT89] Jean-Yves Girard, Yves Lafont, and Paul Taylor. *Proofs and Types*. Cambridge tracts in theoretical computer science 7. Cambridge University Press, 1989.

[GM01] Stefano Guerrini and Andrea Masini. "Parsing MELL proof nets". In: *Theoretical Computer Science* 254.1–2 (2001), pp. 317–335.

[Gon23] Georges Gonthier. *A computer-checked proof of the Four Color Theorem*. Tech. rep. Inria, Mar. 2023. URL: https://inria.hal.science/hal-04034866.

[Gro95] Philippe de Groote. "Linear logic with isabelle: Pruning the proof search tree". In: *Theorem Proving with Analytic Tableaux and Related Methods*. Ed. by Peter Baumgartner, Reiner Hähnle, and Joachim Possega. Springer, Sept. 1995, pp. 263–277. DOI: 10.1007/3-540-59338-1_41. URL: https://link.springer.com/chapter/10.1007/3-540-59338-1_41.

[Gue11] Stefano Guerrini. "A linear algorithm for MLL proof net correctness and sequentialization". In: *Theoretical Computer Science* 412.20 (2011), pp. 1958–1978.

[Ham04] Masahiro Hamano. "Softness of MALL proof-structures and a correctness criterion with Mix". In: *Archive for Mathematical Logic* 43 (2004), pp. 751–794. DOI: 10.1007/s00153-004-0222-6.

[Hei11a] Willem Heijltjes. "Graphical Representation of Canonical Proof: Two case studies". Ph.D. thesis. University of Edinburgh, 2011.

[Hei11b] Willem Heijltjes. "Proof Nets for Additive Linear Logic with Units". In: *Proceedings of the twenty-sixth annual symposium on Logic In Computer Science*. IEEE. Toronto: IEEE Computer Society Press, June 2011, pp. 207–216.

[HG05] Dominic Hughes and Rob van Glabbeek. "Proof Nets for Unit-free Multiplicative-Additive Linear Logic". In: *ACM Transactions on Computational Logic* 6.4 (2005), pp. 784–842. DOI: 10.1145/1094622.1094629.

[HG16] Dominic Hughes and Rob van Glabbeek. *MALL proof nets identify proofs modulo rule commutation*. 2016. URL: https://arxiv.org/abs/1609.04693.

[HH15] Willem Heijltjes and Dominic Hughes. "Complexity Bounds for Sum-Product Logic via Additive Proof Nets and Petri Nets". In: *Proceedings of the 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. IEEE, 2015, pp. 80–91. DOI: 10.1109/LICS.2015.18.

[HH16a] Willem Heijltjes and Robin Houston. "Proof equivalence in MLL is PSPACE-complete". In: *Logical Methods in Computer Science* 12.1 (2016). DOI: 10.2168/LMCS-12(1:2)2016.

[HH16b] Willem Heijltjes and Dominic Hughes. "Conflict nets: Efficient locally canonical MALL proof nets". In: *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. ACM Press, July 2016, pp. 437–446. DOI: 10.1145/2933575.2934559.

[HHS19] Willem Heijltjes, Dominic Hughes, and Lutz Strassburger. "Proof Nets for First-Order Additive Linear Logic". In: *4th International Conference on Formal Structures for Computation and Deduction (FSCD)*. Ed. by Herman Geuvers. Vol. 131. Leibniz International Proceedings in Informatics (LIPIcs). Schloss Dagstuhl – Leibniz-Zentrum fuer Informatik, 2019, 22:1–22. DOI: 10.4230/LIPIcs.FSCD.2019.22.

[HJ97] Hongde Hu and André Joyal. "Coherence Completions of Categories and Their Enriched Softness". In: *Electronic Notes in Theoretical Computer Science* 6 (1997). MFPS XIII, Mathematical Foundations of Progamming Semantics, Thirteenth Annual Conference, pp. 174–190. ISSN: 1571-0661. DOI: 10.1016/S1571-0661(05)80162-8. URL: https://www.sciencedirect.com/science/article/pii/S1571066105801628.

[HJ99] Hongde Hu and Andre Joyal. "Coherence completions of categories". In: *Theoretical Computer Science* 227.1 (1999), pp. 153–184. ISSN: 0304-3975. DOI: 10.1016/S0304-3975(99)00051-1. URL: https://www.sciencedirect.com/science/article/pii/S0304397599000511.

[Hue80] Gérard Huet. "Confluent Reductions: Abstract Properties and Applications to Term Rewriting Systems: Abstract Properties and Applications to Term Rewriting Systems". In: *Journal of the ACM* 27.4 (Oct. 1980), pp. 797–821. DOI: 10.1145/322217.322230.

[Hug12] Dominic Hughes. "Simple free star-autonomous categories and full coherence". In: *Journal of Pure and Applied Algebra* 216.11 (2012), pp. 2386–2410. ISSN: 0022-4049. DOI: 10.1016/j.jpaa.2012.03.020. URL: https://www.sciencedirect.com/science/article/pii/S0022404912001089.

[Hug13] Dominic Hughes. "Simple multiplicative proof nets with units". In: *Annals of Pure and Applied Logic* (2013). URL: http://arxiv.org/abs/math/0507003.

[Hug18] Dominic Hughes. "Unification nets: canonical proof net quantifiers". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*. Ed. by Anuj Dawar and Erich Grädel. Association for Computing Machinery, 2018, pp. 540–549. DOI: 10.1145/3209108.3209159. URL: http://arxiv.org/abs/1802.03224.

[JM11] Paulin Jacobé de Naurois and Virgile Mogbil. "Correctness of Linear Logic Proof Structures is NL-Complete". In: *Theoretical Computer Science* 412.20 (Apr. 2011), pp. 1941–1957. DOI: 10.1016/j.tcs.2010.12.020. URL: https://hal.science/hal-00360894.

[Joy95a] André Joyal. "Free Bicomplete Categories". In: *Mathematical Reports of the Academy of Sciences* 17.5 (1995), pp. 219–224.

[Joy95b] André Joyal. "Free Bicompletion of Enriched Categories". In: *Mathematical Reports of the Academy of Sciences* 17.5 (1995), pp. 213–218.

[JU76] J. A. Bondy and U. S. R. Murty. *Graph Theory with Applications*. 1976.

[Kot59] Anton Kotzig. "On the theory of finite graphs with a linear factor. II." In: *Matematicko-Fyzikálny Časopis* 09.3 (1959). In Slovak, with as original title "Z teórie konečných grafov s lineárnym faktorom. II.", pp. 136–159. URL: https://eudml.org/doc/29908.

[KP95] Sara Kalvala and Valeria de Paiva. "Mechanizing Linear Logic in Isabelle". In: *Isabelle Users Workshop*. Sept. 1995. URL: https://www.cl.cam.ac.uk/~lp15/papers/Workshop/papers/kalvala-linear.pdf.

[Lam96] François Lamarche. "From Proof Nets to Games: extended abstract". In: *Electronic Notes in Theoretical Computer Science* 3 (1996), pp. 107–119. DOI: 10.1016/S1571-0661(05)80409-8.

[Lau05] Olivier Laurent. "Classical isomorphisms of types". In: *Mathematical Structures in Computer Science* 15.5 (Oct. 2005), pp. 969–1004.

[Lau17] Olivier Laurent. *Yalla: Yet Another deep embedding of Linear Logic in Coq*. Coq library. July 2017. URL: https://perso.ens-lyon.fr/olivier.laurent/yalla/.

[Lau99] Olivier Laurent. "Polarized proof-nets: proof-nets for LC (Extended Abstract)". In: *Typed Lambda Calculi and Applications '99*. Ed. by Jean-Yves Girard. Vol. 1581. Lecture Notes in Computer Science. Springer, Apr. 1999, pp. 213–227.

[Lin95] Patrick Lincoln. "Deciding provability of linear logic formulas". In: *Advances in Linear Logic*. Ed. by Jean-Yves Girard, Yves Lafont, and Laurent Regnier. Vol. 222. London Mathematical Society Lecture Note Series. Cambridge University Press, 1995, pp. 109–122.

[LL22] The LL Handbook Project. *Handbook of Linear Logic (draft)*. Early draft for the Linear Logic Winter School. Jan. 2022. URL: https://ll-handbook.frama.io/ll-handbook/ll-handbook-public.pdf.

[LM08] Olivier Laurent and Roberto Maieli. "Cut Elimination for Monomial MALL Proof Nets". In: *Proceedings of the twenty-third annual symposium on Logic In Computer Science*. IEEE. IEEE Computer Society Press, June 2008, pp. 486–497.

[LN22] Ruonan Li and Bo Ning. "A revisit to Bang-Jensen-Gutin conjecture and Yeo's theorem". 2022. DOI: 10.48550/arXiv.2207.03793. URL: https://arxiv.org/abs/2207.03793.

[Mac] Saunders Mac Lane. *Categories for the Working Mathematician*. 2nd ed. Springer. DOI: 10.1007/978-1-4757-4721-8.

[Mai07] Roberto Maieli. "Retractile Proof Nets of the Purely Multiplicative and Additive Fragment of Linear Logic". In: *Logic for Programming, Artificial Intelligence, and Reasoning Proceedings*. Ed. by Nachum Dershowitz and Andrei Voronkov. Vol. 4790. Lecture Notes in Computer Science. Springer, 2007, pp. 363–377. DOI: 10.1007/978-3-540-75560-9_27.

[MC] The MC Team. *Mathematical Components*. Coq library. URL: https://math-comp.github.io/.

[Nen14] Gleb Nenashev. "A short proof of Kotzig's theorem". 2014. DOI: 10.48550/arXiv.1402.0949. URL: https://arxiv.org/abs/1402.0949.

[Ngu18a] Lê Thành Dũng Nguyễn. "On the complexity of finding cycles in proof nets". DICE 2018 workshop. 2018. URL: http://cl-informatik.uibk.ac.at/users/zini/events/dice18/abstracts/N.pdf.

[Ngu18b] Lê Thành Dũng Nguyễn. "Unique perfect matchings and proof nets". In: *3rd International Conference on Formal Structures for Computation and Deduction (FSCD 2018)*. Ed. by Hélène Kirchner. Vol. 108. Leibniz International Proceedings in Informatics (LIPIcs). 2018, 25:1–25:20. DOI: 10.4230/LIPIcs.FSCD.2018.25.

[Ngu20] Lê Thành Dũng Nguyễn. "Unique perfect matchings, forbidden transitions and proof nets for linear logic with Mix". In: *Logical Methods in Computer Science* 16.1 (Feb. 2020). DOI: 10.23638/LMCS-16(1:27)2020.

[Ohl98] Enno Ohlebusch. "Church-Rosser theorems for abstract reduction modulo an equivalence relation". In: *Rewriting Techniques and Applications*. Ed. by Tobias Nipkow. Vol. 1379. Lecture Notes in Computer Science. Springer, 1998, pp. 17–31. ISBN: 978-3-540-69721-3.

[Pad01] Vincent Padovani. "Retracts in Simple Types". In: *Typed Lambda Calculi and Applications '01*. Ed. by Samson Abramsky. Vol. 2044. Lecture Notes in Computer Science. Springer, May 2001, pp. 376–384.

[PC] Pierre-Marie Pédrot and Guillaume Claret. *ll_coq*. Coq library. URL: https://github.com/ppedrot/ll-coq.

[PT10] Michele Pagani and Lorenzo Tortora de Falco. "Strong normalization property for second order linear logic". In: *Theoretical Computer Science* 411.2 (2010), pp. 410–444.

[PT17] Michele Pagani and Paolo Tranquilli. "The Conservation Theorem for Differential Nets". In: *Mathematical Structures in Computer Science* 27.6 (2017), pp. 939–992. DOI: 10.1017/S0960129515000456.

[PW99] James Power and Caroline Webster. "Working with Linear Logic in Coq". In: *Theorem Proving in Higher Order Logics: Emerging Trends*. An implementation is available at https://github.com/ComputerAidedLL/PowerWebster_ILL. Sept. 1999. URL: http://www-sop.inria.fr/croap/TPHOLs99/proceeding.html.

[Ret03] Christian Retoré. "Handsome proof-nets: perfect matchings and cographs". In: *Theoretical Computer Science* 294.3 (2003), pp. 473–488.

[Rit91] Mikael Rittri. "Using types as search keys in function libraries". In: *Journal of Functional Programming* 1.1 (1991), pp. 71–89.

[Rob03] Edmund Robinson. "Proof Nets for Classical Logic". In: *Journal of Logic and Computation* 13.5 (Oct. 2003), pp. 777–797. DOI: 10.1093/logcom/13.5.777. URL: https://doi.org/10.1093/logcom/13.5.777.

[RU02] Laurent Regnier and Pawel Urzyczyn. *Retractions of Types with Many Atoms*. 2002. URL: http://arxiv.org/abs/cs/0212005.

[Sad03] Mehrnoosh Sadrzadeh. "Modal Linear Logic in Higher Order Logic, an experiment in Coq". In: *Theorem Proving in Higher Order Logics (01/09/03)*. Ed. by D Basin and W Burkhart. Sept. 2003, pp. 75–93. URL: https://eprints.soton.ac.uk/261814/.

[See89] Robert Seely. "Linear logic, $\star$-autonomous categories and cofree coalgebras". In: *Contemporary mathematics* 92 (1989).

[Sey78] Paul D. Seymour. "Sums of circuits". In: *Graph Theory and Related Topics* (1978). Ed. by J. A. Bondy and U. S. R. Murty, pp. 341–355.

[Sol83] Sergei Soloviev. "The category of finite sets and cartesian closed categories". In: *Journal of Soviet Mathematics* 22.3 (1983), pp. 1387–1400.

[SS78] D. J. Shoesmith and T. J. Smiley. *Multiple-Conclusion Logic*. Cambridge University Press, 1978.

[SS79] D. J. Shoesmith and T. J. Smiley. "Theorem on Directed Graphs, Applicable to Logic". In: *Journal of Graph Theory* 3.4 (1979), pp. 401–406. DOI: 10.1002/jgt.3190030412. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/jgt.3190030412.

[Str09] Lutz Strassburger. "Some Observations on the Proof Theory of Second Order Propositional Multiplicative Linear Logic". In: *Typed Lambda Calculi and Applications* (2009), pp. 309–324. DOI: 10.1007/978-3-642-02273-9_23.

[Str10] Lutz Strassburger. "What is the Problem with Proof Nets for Classical Logic ?" In: *Proceedings of CiE'10*. Apr. 2010. URL: https://www.lix.polytechnique.fr/~lutz/papers/CiE10.pdf.

[Sze04] Stefan Szeider. "On Theorems Equivalent with Kotzig's Result on Graphs with Unique 1-Factors". In: *Ars Combinatoria* 73 (2004), pp. 53–64. URL: https://www.ac.tuwien.ac.at/files/pub/szeider-AC-2004.pdf.

[Ter03] Terese. *Term Rewriting Systems*. Vol. 55. Cambridge tracts in theoretical computer science. Cambridge University Press, 2003.

[Tor01] Lorenzo Tortora de Falco. "Additives of linear logic and normalization- Part II: the additive standardization theorem". Sept. 2001.

[Tor03] Lorenzo Tortora de Falco. "Additives of linear logic and normalization- Part I: a (restricted) Church-Rosser property". In: *Theoretical Computer Science* 294.3 (Feb. 2003), pp. 489–524. DOI: 10.1016/S0304-3975(01)00176-1.

[Tra09] Paolo Tranquilli. "Confluence of Pure Differential Nets with Promotion". In: *Proceedings of the 23rd CSL International Conference and 18th EACSL Annual Conference on Computer Science Logic*. CSL'09/EACSL'09. Springer, 2009, pp. 500–514. DOI: 10.5555/1807662.1807701. URL: https://hal.science/hal-00374777/.

[Xav+18] Bruno Xavier, Carlos Olarte, Giselle Reis, and Vivek Nigam. "Mechanizing Focused Linear Logic in Coq". In: *12th Workshop on Logical and Semantic Frameworks with Applications (LSFA 2017)*. Ed. by Sandra Alves and Renata Wassermann. Vol. 338. Code source available at https://github.com/meta-logic/coq-ll. 2018, pp. 219–236. DOI: 10.1016/j.entcs.2018.10.014. URL: https://www.sciencedirect.com/science/article/pii/S157106611830080X.

[Yeo97] Anders Yeo. "A Note on Alternating Cycles in Edge-Coloured Graphs". In: *Journal of Combinatorial Theory, Series B* 69.2 (1997), pp. 222–225. DOI: 10.1006/jctb.1997.1728.