SHARING, SUPERPOSITION AND EXPANSION

GEOMETRICAL STUDIES ON THE SEMANTICS AND IMPLEMENTATION OF λ-CALCULI AND PROOF-NETS

Marco Solieri

Version 1.2
Last updated on Friday 28th October, 2016 at 02:22

Typeset by the author, using LaTeX, PGF/TikZ, Kile, TeXstudio and many other pieces of free software

Abstract

Elegant semantics and efficient implementations of functional programming languages can both be described by the very same mathematical structures, most prominently within the Curry-Howard correspondence, where programs, types and execution respectively coincide with proofs, formulæ and normalisation. Such a flexibility is sharpened by the deconstructive and geometrical approach pioneered by linear logic (LL) and proof-nets, and by Lévy-optimal reduction and sharing graphs (SG).

Adapting Girard's geometry of interaction, this thesis introduces the geometry of resource interaction (GoRI), a dynamic and denotational semantics, which describes, algebraically by their paths, terms of the resource calculus (RC), a linear and non-deterministic variation of the ordinary lambda calculus. Infinite series of RC-terms are also the domain of the Taylor-Ehrhard-Regnier expansion, a linearisation of LC. The thesis explains the relation between the former and the reduction by proving that they commute, and provides an expanded version of the execution formula to compute paths for the typed LC.

SG are an abstract implementation of LC and proof-nets whose steps are local and asynchronous, and sharing involves both terms and contexts. Whilst experimental tests on SG show outstanding speedups, up to exponential, with respect to traditional implementations, sharing comes at price. The thesis proves that, in the restricted case of elementary proof-nets, where only the core of SG is needed, such a price is at most quadratic, hence harmless.

Résumé

Des sémantiques élégantes et des implémentations efficaces des langages de programmation fonctionnels peuvent être décrits par les mêmes structures mathématiques, notamment dans la correspondance Curry-Howard, où le programmes, les types et l'exécution, coïncident aux preuves, formules et normalisation. Une telle flexibilité est aiguisé par l'approche deconstructif et géométrique de la logique lineaire (LL) et les réseaux de preuve, et de la réduction optimale et les graphes de partage (SG).

En adaptent la géométrie de l'interac-tion de Girard, cette thèse propose une géométrie de l'interaction des ressources (GoRI), une sémantique dynamique et dénotationelle, qui décrit algébriquement par leur chemins, les termes du calcul des ressources (RC), une variation linéaire et non-déterministe du lambda calcul (LC). Les séries infinis dans RC sont aussi le domaine du développement de Taylor-Ehrhard-Regnier, une linéarisation du LC. La thèse explique la relation entre ce dernier et la réduction démontrant qu'ils commutent, et présente une version développé de la formule d'exécution pour calculer les chemins du LC typé.

Les SG sont un modèle d'implémen-tation du LC, dont les pas sont locales et asynchrones, et le partage implique et les termes et les contextes. Bien que les tests ont montré des accélérations exceptionnelles, jusqu'à exponentielles, par rapport aux implémentations traditionnelles, les SG n'ont pas que des avantages. La thèse montre que, dans le cas restreint des reseaux élémentaires, où seule le cœur des SG est requis, les désavantages sont au plus quadratique, donc inoffensifs.

Sommario

Semantiche eleganti ed implemen-tazioni efficienti di linguaggi di programmazione funzionale possono entrambe essere descritte dalle stesse strutture matematiche, più notevolmente nella corrispondenza Curry-Howard, dove i programmi, i tipi e l'esecuzione coincidono, nell'ordine, con le dimostrazioni, le formule e la normalizzazione. Tale flsesibilità è acuita dall'approccio decostruttivo e geometrico della logica lineare (LL) e le reti di dimostrazione, e della riduzione ottimale e i grafi di condivisione (SG).

Adattando la geometria dell'intera-zione di Girard, questa tesi introduce la geometria dell'interazione delle risorse (GoRI), una semantica dinamica e denotazionale che descrive, algebricamente tramite i loro percorsi, i termini del calcolo delle risorse (RC), una variante lineare e non-deterministica del lambda calcolo ordinario. Le serie infinite di termini del RC sono inoltre il dominio dell'espansione di Taylor-Ehrhard-Regnier, una linearizzazione del LC. La tesi spiega la relazione tra quest'ultima e la riduzione dimostrando che esse commutano, e fornisce una versione espansa della formula di esecuzione per calcolare i percorsi del LC tipato.

I SG sono un modello d'implemen-tazione del LC, i cui passi sono locali e asincroni, e la cui condivisione riguarda sia termini che contesti. Sebbene le prove sperimentali sui SG mostrino accellerazioni eccezionali, persino esponenziali, rispetto alle implementazioni tradizionali, la condivisione ha un costo. La tesi dimostra che, nel caso ristretto delle reti elementari, dove è necessario solo il cuore dei SG, tale costo è al più quadratico, e quindi innocuo.

Contents

Chapter 1

Introduction

Computer science is one of the few where elegance and efficacy, which elsewhere are frequently and variously opposed, can both be pursued exploiting their complementarities. Programming language theory is a notable example. Both foundational and practical questions, for instance about semantics and implementations, can be phrased, explored and sometimes answered employing the very same frameworks. A prominent one is the Curry-Howard correspondence with proof theory, since there models of functional programs, types and execution essentially coincide with those of proofs, formulæ and normalisation.

A modest example is given by the contributions presented in this thesis, and the literature which it is based upon, ranging from the denotational semantics of a non-deterministic variation of the λ-calculus, to the efficiency of a distributed model of computation for the ordinary one. Such conceptually distant results have all been understood and expressed within a narrow set of mathematical tools inspired by linear logic.

1.1 Proof nets

Linear logic [Girard, 1987] unveiled the relation between the algebraic concept of linearity and the computational property of a function argument to be used exactly once. Its formulæ are syntactically discriminated depending on their usage in a proof, as if they are resources whose access is not gratis. Linear formulæ must be used exactly once in the proof, whilst the others, marked by the exponential modality, can instead arbitrarily be duplicated or erased. Thanks to this separation, the λ-calculus is allowed to have a subtler type system, where operations of duplication and erasure are detached from β-reduction.

Proof-nets are one of the key tools introduced by linear logic. They are a graphical representation of proofs in which each rule corresponds to a graph constructor named link connecting the main premises of the rule to its main conclusion. Such a graphical approach allows to naturally equate proofs that, because of the rigidity of the traditional

sequent calculus, differ for bureaucratic details only. In the intuitionistic case, a proof-net is in fact a decorated graph representation of the corresponding λ-term (with explicit substitutions, to be precise [Di Cosmo, Kesner, and Polonovski, 2000, Accattoli, 2013]) that adds exponential boxes: delimited sub-graphs that can be erased or duplicated only as a whole. Another remarkable trait of proof-nets is that they allow the definition of the system not only with an inductive formalisation given by axioms and logical rules, but also with a geometrical condition on graphs that are freely built by assembling links together: acyclicity of some kind paths, called switching.

1.2 Geometry of Interaction

The dynamics of β-reduction or cut elimination can be described in a purely geometric way — studying paths in some graph representations of terms or proofs, and looking at those which are *persistent*, i.e. that have a residual path in any reduct. The quest for an effective semantic characterisation of persistence separately produced three notions of paths.

Regularity An algebraical condition arising from a set of equations over a monoidal structure called the dynamic algebra, and whose objects are assigned to paths. Firstly presented by Girard [1989], the formulation has been simplified by Danos and Regnier [1995].

Legality Formulated in terms of pure graphical conditions by Asperti and Laneve [1995], it allow to precisely observe the correspondence with Lévy's labels [1978] and virtual redexes (see also Section 1.6).

Consistency A simple matching condition on execution traces of a token-machine, called contexts, developed by Gonthier, Abadi, and Lévy [1992a], and for which exist an enjoyable tutorial by Mairson [2002]. It was formulated to understand the implementations [Lamping, 1989, Kathail, 1990] of the optimal reduction à la Lévy.

These notions are equivalent [Asperti, Danos, Laneve, and Regnier, 1994], and their common core idea — describing computation by local and asynchronous conditions on routing of paths — inspired the design of efficient parallel abstract machines [among others, Mackie, 1995, Danos et al., 1997, Laurent, 2001, Pinto, 2001, Pedicini and Quaglia, 2007, Dal Lago et al., 2014, Pedicini et al., 2014, Dal Lago et al., 2015]. More recently, the geometry of interaction (GoI) approach has been fruitfully employed for semantic investigations which characterised quantitative properties of programs, with respect to both time [e.g. Dal Lago, 2009, Perrinel, 2014, Aubert et al., 2016] and space complexity [Aubert and Seiller, 2014, 2015, Mazza, 2015b, Mazza and Terui, 2015].

1.3 Taylor-Ehrhard-Regnier expansion and resource calculus

The previously mentioned decomposition of the intuitionistic implication enabled a differential constructor and linear combinations to extend: the λ-calculus into the differential λ-calculus, discovered by Ehrhard and Regnier [2003]; and, more generally, linear logic into the differential linear logic (DiLL) [Ehrhard and Regnier, 2006a, Tranquilli, 2011]. These constructions allow considering the Taylor expansion of a term [Ehrhard and Regnier, 2008], which rewrites it as an infinite series of terms of the resource λ-calculus (RC). It is a completely linear restriction of the differential λ-calculus, similar to the λ-calculus with multiplicities [Boudol, 1993], where the argument of an application is a superposition of terms and must be linearly used. Taylor-Ehrhard-Regnier expansion contains any finite approximation of the head-normalisation of a term, as evoked by its commutativity with Böhm trees: the expansion of the Böhm tree of a term is equal to the normal form of its expansion [Ehrhard and Regnier, 2006b].

The approximation of λ-calculus has been studied through linear logic's sub-structural lens also using affine calculi, those where duplication is forbidden, but erasure is allowed, [e.g. Mazza, 2015a], and also using legal paths to guide the very process of linearisation [Alves and Florido, 2005]. Moreover, Taylor-Ehrhard-Regnier expansion originated various investigations on quantitative semantics, using the concept of power series for describing program evaluation, and has been applied in various non-standard models of computation [see Danos and Ehrhard, 2011, Pagani, Selinger, and Valiron, 2014, for example].

1.4 Light logics

Intuitionistic Light Affine Logic (ILAL) is a variant of Girard's Light Linear Logic (LLL) [1995] introduced by Asperti and Roversi [2002]. The key property of light logics is that they characterise the class of deterministic polynomial time functions: in these logics, the length of cut-elimination is related to the size of the proof by a polynomial function, whose degree depends on the exponential depth of the proof, i.e., on the maximum number of nested exponential boxes in the corresponding proof net.

In light logics, the control of computational complexity is obtained by restricting the use of the !-exponential. In particular, the usual promotion rule of linear logic is replaced by a functorial version that simultaneously introduces the !-modality on all the formulæ of the sequent, with the proviso that the l.h.s. of the sequent contain at most one formula—relaxing this proviso we get elementary complexity instead. In order to code all the polynomial algorithms, light logics requires the introduction of another exponential modality § named paragraph. For this second modality, contraction is not allowed, and its introduction rule is functorial—similarly to the case of the !-promotion, it adds a modality in front of every formula in the sequent—but, differing from the case of the !-promotion, there is no restriction on the size of the l.h.s. of the sequent and, on this side, the added modality can be either a § or an !. In details, here it is the promotion rules for ! and §:

With some provisos due to the presence of second order quantifiers [Dal Lago and Baillot, 2006], the exponential depth of a cut-free ILAL proof of a formula $B = A_k \multimap \ldots A_1 \multimap A_0$ depends on the nesting depth of the exponential modalities in B. As a consequence, by ensuring that a function f can be encoded by a proof of B and that its k arguments can be suitably encoded (i.e., preserving a polynomial relation between the sizes of the input data and of the corresponding proofs) by a proof of type $A_1, \ldots, A_k$, we can conclude that f can be computed in polynomial time. On the other hand, ILAL is complete, since every polynomial time function can be represented into it [Asperti and Roversi, 2002], i.e., encoded by a suitable ILAL proof. Summing up, a function f is polynomial time if and only if it can be represented in ILAL.

IEAL is an elementary time variant of ILAL. In fact, by simply removing the constraint of a unique formula on the left-hand side of the !-promotion rule, the time complexity of the system passes from polynomial to elementary (i.e., bounded by a Kalmar elementary function). In IEAL there is no need of the §-modality. Moreover, since every ILAL proof can be transformed into an IEAL proof by replacing every § with an !, we see that ILAL is indeed a subsystem of IEAL.

1.5 Sharing graphs

Sharing graphs are a graph rewriting system introduced by Lamping [1989] [but also independently by Kathail, 1990] in order to implement Lévy's optimal reductions for λ-calculus β-reduction [1978, 1980]. Lamping's technique was successively cleaned and reformulated by Gonthier, Abadi, and Lévy [1992a], who also showed its relations with Girard's geometry of interaction [1989] and pointed out that, indeed, sharing graphs implement a local and distributed algorithm for linear logic box duplication Gonthier, Abadi, and Lévy [1992b] [see Asperti and Guerrini, 1998, for a comprehensive presentation of sharing graphs and optimal reductions]).

Sharing graphs for optimal reduction can be also seen as a variant of interaction nets [Lafont, 1990], in which the interacting agents (the links) are indexed by a natural number. The key property of interaction nets is that they are strongly confluent. In fact, an interaction net is formed by agents with a given number of ports, and edges connecting pairs of ports in distinct agents, such that each port has a unique incident edge. Each agent has a unique principal port through which it can interact with another agent only if the edge is connected to the principal port of the other agent too. Each redex formed by a pair of connected principal ports can be rewritten in a unique way. Summing up, the redexes of an interaction net are disjoint and can be rewritten in a unique way; as a consequence, the corresponding rewriting system is strongly confluent.

Among the sharing graphs rules that allow to implement β-reduction, only one, the β_s-rule, actually corresponds to reducing one or several shared β-redexes; the other rules correspond instead to bookkeeping operations allowing to manage sharing and to explicit the connection between links corresponding to β-redexes eventually hidden by sharing nodes. Optimal sharing graphs implement Lévy's optimal reduction, in the sense that the optimal sharing reduction of a term t corresponds to a Lévy family reduction of t whose length is equal to the number of β_s-rules in the sharing reduction.

1.6 Lévy-optimal reduction

The aim of Lévy-optimal theory [1978] was to define a notion of optimal cost for the reduction of a β-term. By reasoning on the notion of feasible sharing achievable in an effective implementation of β-reduction, Lévy defined a notion of family of redexes collecting all the redexes that can be shared by some efficient implementation — since in some sense they can be seen as coming from a same origin — and he showed that there exists a parallel reduction strategy reducing a whole family of redexes at each step. According to this, the length of a family reduction reducing a needed redex at each step (a redex is needed when one of its residual must be reduced by, or is definitely present after, any reduction sequence of the initial term) gives a lower bound to the complexity of any reduction of a term. Therefore, by giving an implementation of Lévy optimal reduction with a cost at least polynomially related to the length of the number of Lévy families, one would show that Lévy families are a good measure of the cost of the reduction of a λ-term.

Unfortunately, Asperti and Mairson [2001] proved that this is not case, by showing that the cost of the bookkeeping reduction in the sharing graph reduction of a term may become Kalmar-elementary bigger than the number of β_s-rules in any normalizing reduction, i.e. the cost of reducing such terms is not polynomially related to the number of families in the term. This result is sometimes misunderstood as a negative one about the efficiency of implementations with sharing graphs, even though it does not say anything about it. In fact, it is achieved by showing that the cost of the reduction of the λ-terms in a given family is at least Kalmar-elementar in their size. Nevertheless, such terms have a polynomial number of families. Thus the mismatch between the effective cost of the reduction and the Lévy optimal cost.

We stress that the above results does not address any negative consideration on sharing implementations. They state that the number of bookkeeping operations in an optimal sharing reduction can be much bigger than the number of β_s-rules, not bound by any Kalmar elementary function in the worst cases. But this is a consequence of the intrinsic hardness of the language — the simply types λ-calculus — in which the cost of the reduction cannot be bound by any Kalmar-elementary function [Statman, 1979]. Therefore, no other β-reduction implementation can do better than such a bound neither.

Empirical evidences from real implementations, for limited they can be, showed on the other hand very promising performances. For instance, benchmarks of the Bologna higher order optimal machine [Asperti, Giovannetti, and Naletto, 1996] on pure λ-terms recorded polynomial cost of reduction against an exponential one of traditional implementations of functional languages. This does not scale to 'real world' functional programs, for which performances were measured in the same order of magnitude. At the moment of writing, another optimal implementation, the King's College lambda evaluator [Mackie, 2004] is considered the most efficient implementations of the λ-calculus.

The question of effective *efficiency* of sharing reduction is then still open, and the case of light and elementary logics is particularly significant. In fact, in the general case, one can distinguish two kinds of bookkeeping rules: the duplication rules that propagate sharing links, and the rules for the control links, the so-called brackets, ensuring the proper propagation of sharing links. The duplication rules correspond to a step by step

implementation of the duplication of the arguments of β-rules— after the execution of a $β_s$ rule, the argument of the rule is not duplicated, but a sharing node is inserted. The sharing node will then duplicate link by link the argument when and if necessary. According to the downward orientation from the node corresponding to the root of a term to the nodes corresponding to variables, the scope of a duplication link ends at the variables of the duplicating term. However, in sharing reductions, duplication does not proceed downwards only — from the root of a term towards its leafs, which correspond to its variables — but in some cases (when duplicating a λ-abstraction), a duplication link propagating upwards is inserted. Such a link is like a closed bracket matching a corresponding duplication link propagating downwards and delimit the bottom of its scope. In other words, matching links are associated to the same duplication process and they delimit the part of the graph that is left to duplicate.

Unfortunately, in the general case, matching links do not properly nest as a sequence of matching brackets, since in some cases a duplication link propagating downwards may face a duplication link propagating upwards that do not matches with it. When two matching links faces, their scope is empty — they correspond to an opening bracket immediately followed by a closing one, so they have nothing to duplicate — and they can be erased; otherwise, the two links must swap continuing propagating (see the mux rules in Figure 5.6a).

A direct implementation of sharing reduction requires then a sort of *oracle* allowing to decide when two facing sharing links are matching [see examples in Asperti and Guerrini, 1998, introduction]. Lamping [1989] showed that such an oracle can be implemented by labelling each link with an index and by introducing some new control nodes, named *brackets*, with a suitable set of rewriting rules managing link indexes. Lamping indexing ensures that two facing sharing links are matching if and only if they have the same index. Variants of Lamping's algorithm have been proposed in the literature, but in any case we have some links and rules implementing duplication, and some additional links and rules implementing the oracle. The only exception are implementations of light/elementary logics, where link indexes do not change along the reduction, which implies that the oracle is not needed.

Asperti, Coppola, and Martini [2004] extended the result by Asperti and Mairson [2001] to the particular case of elementary logic, proving that in this case also we can have terms with a polynomial number of Lévy's families, and therefore a polynomial number of sharing β-reductions, which require a non-elementary duplication cost — since, again, their reduction cannot be bound by any Kalmar-elementary function. Their analysis adds then an argument in favor to sharing graph implementations. In fact, Baillot, Coppola, and Dal Lago [2011] proved that the sharing implementation of elementary logic do not do worst then the tight lower-bound already known. Moreover they prove the above result for light logic too. In fact, the basic property of light and elementary logics is that the cost of the reduction of any proof is bound by a polynomial in the size of the proof, for the light case, and by an elementary function, for the elementary case. Baillot, Coppola and Dal Lago show that the cost of the sharing reduction of a light or elementary proof can be bound by a polynomial and elementary function, respectively. Which allow to get two conclusions at the same time: first of all, another proof of the fact that light and elementary logics are sound with respect to the expected computational bound; second, that sharing implementation introduce an

overhead with respect to the usual implementation of reduction that is limited by the same computational bound.

1.7 Summary of contributions

We will present results that can be ascribed to two lines of investigations, both rooted in the related in their essence to the dynamics of linear logic's contraction, and both based on a geometrical viewpoint. The first, within the area of semantics, explore the structure of superposition of resource calculus and that of Taylor expansion. The second, in the theory of implementation, tackle the question of efficiency of sharing graphs.

1.7.1 Superposition and expansion (Part I)

How can geometry of interaction and Taylor-Ehrhard-Regnier expansion interact? What is the GoI for resource calculus? How are paths dynamics related before and after the expansion? Can we expand β-reduction into an infinite parallel step of resource reductions? Is there a linear and non-deterministic GoI for the resource calculus? Can we use it to characterise persistent paths in λ-terms, via expansion? The first part of the thesis addresses these questions and recounts the interplay between the two aforementioned semantic approaches, exploring both directions of their mutual influence.

In Chapter 2 we recall the needed preliminary notions about proof-nets for minimal MELL: the translation of λ-terms (Section 2.2), and the cut-elimination, together with che closed reduction reduction strategy (Section 2.3). Also, we introduce a concise reworked version of the geometry of interaction for such framework. We will introduce the appropriate notion of paths and inspect its dynamics, formalising the property of persistence, which intuitively is the ability to survive to the graph rewriting until the normal form (Section 2.4). On top of this, we articulate the algebraic structure of weights, traditionally named "the dynamic algebra" (Section 2.5). Such algebra features a structural part which models path operations like concatenation and reversal, and a logical part which instead represent the interaction between paths, that lies under the reduction of containing proof-nets. The main intuition behind it is that the algebra's equations check for persistence of a path by either: simplifying, and sometimes even neutralising, weights of sub-paths that can be deformed, or shortened by reduction; or annihilating the weights of those that can be destroyed by the latter. We will show that the execution of a proof-net, i.e. the set of weights assigned to all its paths, is invariant under reduction (Theorem 2.1). This implies that regularity is equivalent to persistence (Corollary 2.1).

In Chapter 3 we introduce resource interaction nets (RINs) and construct their geometry of interaction. RINs are a promotion-free fragment of minimal differential logic, or a linear-logic-like typing system and graphical formalism for RC. We first give an introduction to RC (Section 3.2) and then, following the linear logic tradition, we present RINs (Section 3.3) by mean of a correctness criterion, and we also show that the inductive definition by translation/typing of resource terms is equivalent, once it is closed by reduction.

We then adapt some of the previously introduced notions about paths to deal with the fact that the reduct of a term t is a sum of terms $t_1 + \ldots + t_n$ (Section 3.4). Also, we observe that every path in the net-representation of t_i has to be a residual of some path in the net of t, and that the reduction strongly normalises. Thus, we say that a path of t is persistent whenever it has a residual in at least one of the addends of the reduct of (the net of) t. Restricting to the constant type, whose only inhabitant is the value $\star$, we have $t \to \star + \ldots + \star$. Now there is only one persistent path of $\star$, the trivial one, therefore we prove that persistent paths of t are as many as persistent paths of its normal form (Theorem 3.1).

Furthermore, we define a suitable GoI for RC, in order to characterise persistence (Section 3.5). We define the notion of regularity by $r\mathfrak{L}^*$, a monoidal structure simplifying the dynamic algebra, where exponential modalities (! and ?) become an n-ary variant of the multiplicatives connectives (resp. $\otimes$ and $\invamp$), whose premises are not ordered. Morally, they are the sum of those generalised multiplicatives we obtain by considering all the n! permutations of their premises. We weigh paths with objects of $r\mathfrak{L}^*$, and then consider the sum of the weights of all paths in a RIN. What we obtain is the execution formula for resource nets, which is shown to be invariant under reduction (Theorem 3.5). Therefore, the construction provides a denotational semantics for RC and of the corresponding logic, that is a minimal, propositional and promotion-free fragment of DiLL. From invariance theorem not only we subsume the equivalence of persistence and regularity (Theorem 3.4), that is the usual result of GoI constructions, but we also show that the number of addends in a normal form is equal to the number of regular paths (Corollary 3.1).

In Chapter 4 we introduce a qualitative variant of Taylor-Ehrhard-Regnier expansion (Section 4.2), which maps a proof-net, or a path within it, in an infinite sum of simple RINs, or of set of paths within them. Since there the superposing sum is idempotent, the expansion is essentially the infinite set. Also, we define a notion of expanded cut-elimination (Section 4.3), a kind of infinite parallel reduction for RINs which reduces together all redexes that are copied by the expansion. This allow us to prove (Section 4.4) the commutativity between reduction and Taylor-expansion both at the level of nets (Theorem 4.1) and paths (Theorem 4.3). Thank to this, we discover that the property of being persistent can be transferred along expansion: a path persists to mMELL reduction if and only if there is a path in its expansion which persists to RIN reduction (Theorem 4.4).

This last fact turns out to be particularly interesting, since it enable the definition (Section 4.5) of a variant of the execution formula for typed λ-calculus that is based on the GoRI. The idea is that, if we assign to every path π in a proof-net the infinite sets of $r\mathfrak{L}^*$-weights that belong to the expansions of π, we obtain an expanded formula enjoying the properties of our interest: invariance (Theorem 4.5), hence characterisation of persistence (Corollary 4.1).

1.7.2 Sharing and efficiency (Part II)

Can we tighten the complexity overhead of sharing graph implementation of light and elementary linear logics under the currently known upper bounds (polynomial and ele-

mentary)? Is it possible to do so by a direct syntactical simulation of sharing graphs in proof-nets? Can such approach unveil which components of the sharing graph machinery are the most efficient, and which other the most problematic?

Chapter 5 is devoted to present the sharing implementation of the elementary and light variants of mMELL, recalling its main properties and proving its correctness by syntactical simulations. In Section 5.2 we present the proof-nets of mIELL, that represent a framework of particular convenience because it considerably simplifies the rewriting system, whilst in Section 5.3 we introduce their sharing implementation — the system SG— and its basic properties. A brief review of the most notable qualitative behavioural properties of SG as a Lévy-optimal implementation of mIELL proof-nets is given in Section 5.4. Among these, there is the correctness: any SG-normal-form of a mIELL proof-net N is the mIELL-normal-form of N. A proof is given in the concluding Section 5.5 (cf. Theorem 5.10), where we employ a simple syntactical simulation of SG in mIELL, exploiting an intermediate rewriting system — the unshared graphs (UG) — that possess the structure of mIELL proof-nets and a some sharing markers corresponding to SG graphs.

In Chapter 6 we illustrate an original analysis of the complexity of SG reductions with respect to mIELL reduction, showing that the former cannot be outperformed by the latter, up to a quadratic factor. The two reduction systems present a substantial behavioural difference in the way that they perform the duplication — small or big step — that is clarified when we firstly need to assign a cost for them (Section 6.2). With the intent to exploit the syntactical simulation between the system, we define on unshared graphs the notion of sharing contexts (Section 6.4) that essentially tells, for every vertex and link in a graph, the exact set of lifts (or their copies) that eventually will come there along any reduction.

This tool allow to formulate the two key quantitative correspondences. On the one hand, we can determine the set of sub-graphs of an unshared graph where a lift propagation will happen, the share, that morally represent the non-local portion of a big step duplication's work (Subsection 6.5.1). On the other hand, among a set of lifts that are the unfolding of a k-ary mux m, we can precisely select those that are on the boundary of the share and show that their number correspond indeed to k. These two facts allow indeed to precisely transfer (Section 6.5) and compare (Section 6.6) the costs of mIELL and SG reductions in UG reductions. We show that the cost of logical operations and duplications essentially match between the two systems — the overhead is bounded by a linear function of the mIELL-cost. The same computational bound is obtained, this time indirectly, for annihilations, merges and most swap rules. We are able to limit some special kind of redundant swaps, instead, only by the product of the depth level of the graph and the square of the mIELL cost. Hence, we will prove that the overhead of an SGRB-reduction, i.e. including the read-back rules, with respect to its mIELL simulation is at most quadratic (Theorem 6.1).

Chapter 2

Lambda-calculus, linear logic and geometry of interaction

Contents

2.1 Introduction

Even since the seminal paper by Girard [1989] it was clear that the elegance of the geometry of interaction (GoI) construction reaches its maximum only in some limited cases, where it represents a true denotational semantics, guaranteeing invariance with respect to reduction. This chapter is devoted to recall preliminary notions of a convenient setting for proof-nets and their translation of λ-terms, and to gently introduce the notion that are needed to present an invariant GoI. In order to do so, we present a minimal fragment of MELL and focus our attention on programs with ground types. There we exploit the convenience of closed reduction, that eliminates an exponential cut only if the box has no auxiliary doors (see Fernández, Mackie, and Sinot [2005] for a recent operational discussion of various closed strategies, also includes complexity comparisions), and we are able to formalise a strikingly simple, though detailed, proof of invariance

2.2 Nets and terms

We introduce the multiplicative exponential fragment of linear logic (MELL) restricting it to the minimal version, with propositional and polarised formulæ/types. We shall call it mMELL for short.

We use the terse formulation once dubbed *nouvelle syntaxe* and introduced by Regnier [1992], where all negative links/rules of the exponential fragment, that are dereliction, weakening and binary contraction, are represented with a unique object having an arbitrary number of premisses. With respect to the seminal presentation by Girard [1987] and the advances with respect to polarisation by Laurent [2002], this approach introduces a notable amount of conciseness and usability — from the syntax for terms/proofs to the dynamics of their computation/cut-elimination — not to mention its fruitful similarities to explicit substitution calculi [Di Cosmo et al., 2000, Accattoli, 2013, for instance].
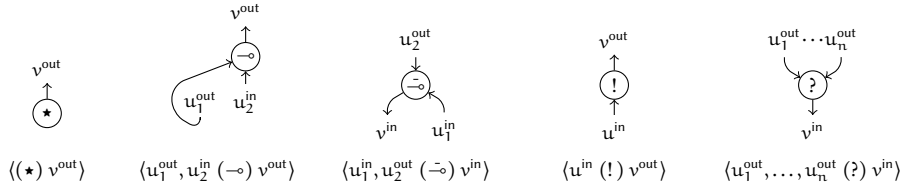
2.2.1 Pre-nets

Definition 2.1 (Links). Given a denumerable set of symbols called *vertices*, a *link* is a biparted and typed hyperedge, i.e. a triple (P, K, C), where:

P is a sequence of vertices, called premisses;

K is an element a finite set of kinds;

C is a singleton of a vertex, called conclusion, disjoint from P.

Figure 2.1 Links: kind, arity and polarity associated to vertices, in both graphical and textual notations. From left to right: constant, abstraction, application, promotion, contraction (or weakening when $n = 0$).

$$\langle (\star)\, v^{out} \rangle \qquad \langle u_1^{out}, u_2^{in}\, (\multimap)\, v^{out} \rangle \qquad \langle u_1^{in}, u_2^{out}\, (\bar{\multimap})\, v^{in} \rangle \qquad \langle u^{in}\, (!)\, v^{out} \rangle \qquad \langle u_1^{out}, \ldots, u_n^{out}\, (?)\, v^{in} \rangle$$

The *polarity* of a vertex is an element in $\{in, out\}$, that we say being opposite, and is determined by K. The *arity* of a link is the length of P, which is also determined by K, except when K = ?. A link $((u_1, \ldots, u_n), \kappa, \{z\})$ will be denoted as $\langle u_1, \ldots u_n\, (\kappa)\, z \rangle$, or more conveniently graphically depicted. In such a case, vertices of a link shall be placed following the usual convention for λ-calculus graphs (out-vertices on top, in-vertices on bottom); the arrow line shall be used to distinguish the conclusion of a link.

A *mMELL link* is a link having kind in $\{\star, \multimap, \bar{\multimap}, !, ?\}$; and whose arity and assigned polarities are determined by the link's kind, as shown in Figure 2.1.

Definition 2.2 (Polarised types). A *polarised type*, or formula, is a word of the following grammar, where $\star$ is the only *ground* type.

$$\mathbb{M} ::= \star \mid \mathbb{E} \multimap \mathbb{M} \tag{2.1}$$

$$\mathbb{E} ::= !\mathbb{M} \tag{2.2}$$

A *typing function* $\mathcal{T}$ is a map from vertices to types. If $l = \langle u_1, \ldots, u_n\, (K(l))\, v \rangle$ is a link, and A, B are types, then $\mathcal{T}$ respects the following constraints.

- If $K(l) = \star$, then $\mathcal{T}(v) = \star$.
- If $K(l) = \multimap$ or $\bar{\multimap}$, then $\mathcal{T}(v) = A \multimap B$ with $A = \mathcal{T}(u_1)$ and $B = \mathcal{T}(u_2)$.
- If $K(l) = ?$ or $!$, then $\mathcal{T}(v) = !A$, where $A = \mathcal{T}(u)$.

Definition 2.3 (Pre-net). A *pre-net* $\mathcal{P}$ is a triple $(V, L, \mathcal{T})$, where V is a set of vertices, L is a set of links and $\mathcal{T}$ is a typing function on V, such that for every vertex $v \in V$ the followings holds.

1. There are at least one and at most two links l, l' such that $l \ni v \in l'$, and when there is only one, then v is called a *conclusion* of $\mathcal{P}$. We wrote $v \in l$ to mean $v \in P(l) \cup C(l)$.

2. the set $C(\mathcal{P})$ of conclusions of $\mathcal{P}$ contains exactly one vertex u with out polarity, and if u is the unique element of $C(\mathcal{P})$ then $\mathcal{P}$ is called *closed*;

3. if $l \ni v \in l'$, then l, l' associate opposite polarities to v. In particular, if $C(l') = v = C(l)$, then v is called a *cut*, if $P(l') \ni v \in P(l)$, then v is called an *axiom*.

We shall also write $V(\mathcal{P})$ and $L(\mathcal{P})$ to denote the first and second component of $\mathcal{P}$, respectively. The type of a pre-net $\mathcal{P}$ is the type $T = \mathcal{T}(v)$, where $v \in C(\mathcal{P})$ of out polarity, written $\mathcal{P} : T$.

The *interior* of $\mathcal{P}$ is the complement of $C(\mathcal{P})$ with respect to $V(\mathcal{P})$. The *interface* of a pre-net $\mathcal{P}$ is the set, for all $v \in C(\mathcal{P})$, of the triple $(v, \mathcal{T}(v), p(v))$ where the last is the polarity of v. Two pre-nets $\mathcal{P}, \mathcal{Q}$ are *equal* when there exists a type-preserving isomorphism $\simeq$ such that $\mathcal{P} \simeq \mathcal{Q}$. Given a pre-net $\mathcal{P} = (V, L, \mathcal{T})$, a sub-pre-net $\mathcal{P}'$ of $\mathcal{P}$ is a pre-net $(V', L', \mathcal{T}')$ such that $V' \subseteq V(\mathcal{P})$, $L' \subseteq L(\mathcal{P}')$, and $\mathcal{T}'$ is the restriction of $\mathcal{T}$ to V'.

2.2.2 Proof-nets and paths

Definition 2.4 (Boxes). A *boxing* b is a function that maps promotions to sub-pre-nets such that if $\mathcal{B} \subseteq \mathcal{P}$ is associated with a !-link l, then $\mathcal{B}$ is a *box* and the followings are satisfied.

1. The vertex $v = P(l)$ belongs to $V(\mathcal{B})$.

2. Any $v' \in \text{iface}(\mathcal{B})$ different from v is the premiss of a ?-link or a conclusion of $\mathcal{P}$.

3. There is no $v' \in V(\mathcal{B})$ such that $P(l') \ni v' \in C(l'')$ whilst $l' \in L(\mathcal{B})$ and $l'' \notin L(\mathcal{B})$.

4. For any !-link l' such that $\mathcal{B}' = b(l')$, if $L(\mathcal{B}) \cap L(\mathcal{B}') \neq \varnothing$ then $\mathcal{B} \subseteq \mathcal{B}'$ or $\mathcal{B} \supseteq \mathcal{B}'$.

Definition 2.5 (mMELL structures). A *mMELL structure* $\mathcal{N}$ is a pair $(\mathcal{P}, b)$ where $\mathcal{P}$ is a pre-net with polarised types and b is a boxing for !-links of $\mathcal{P}$. A *sub-structure* $\mathcal{N}'$ of $\mathcal{N}$, written $\mathcal{N}' \subseteq \mathcal{N}$, is a structure $(\mathcal{P}', b')$ such that: $\mathcal{P}' \subseteq \mathcal{P}$, and b' is the restriction of b to $L(\mathcal{P}')$ such that for any !-link $l \in L(\mathcal{P}')$, $b(l) \subseteq \mathcal{P}'$. If $b(l) = \mathcal{P}'$ for some !-link l in $L(\mathcal{N})$, then:

1. $\mathcal{B} = \mathcal{N}'$ is a *box* of $\mathcal{N}$;

2. the sub-structure made by $\mathcal{B}$ and l is a *bordered box* and written $\overline{\overline{\mathcal{B}}}$;

3. the vertex $v \in \text{iface}(\mathcal{B})$ is called the *principal door* of $\mathcal{B}$ if $v \in C(l)$, otherwise an *auxiliary door*;

4. if $\mathcal{B}$ has no auxiliary doors then is a *closed box*.

The set of boxes of $\mathcal{N}$ is written $\text{Bxs}(\mathcal{N})$. The box $\mathcal{B}$ is depicted drawing a dashed square enclosing all the vertices in $\mathcal{B}$ and connecting its doors. In textual notation, $\mathcal{B}$ will be denoted as its collapse to a link, e.g. as $\langle x_1, \ldots, x_n \; [\mathcal{B}] \; w \rangle$, where w is the principal door and x_i is an auxiliary one.

The *box depth level*, or simply the *depth*, of a vertex v, written $\ell(v)$, is the number of boxes it belongs to. By extension, the depth of a link l is $\ell(C(l))$, whilst the depth of a box $\langle V \; [\mathcal{B}] \; w \rangle$ is $\ell(w)$. Two structures are *equal* when there exists a box- and type-preserving isomorphism between them.

In the rest of the chapter, we will mainly deal with the low-level notion of pre-net, so for the sake of simplicity we will sometimes abuse the notation and talk about a structure

(or a proof-net) where we mean to refer to its pre-net; and we shall as well omit the pedantry about typing and boxing, for instance simply saying "the box of" or "the type of", instead of "the box assigned by the boxing to" or "the type associated by the typing to".

Definition 2.6 (Paths). Given a pre-net $\mathcal{P}$, two vertices $u, w \in \mathcal{P}$ are *connected*, if there is a link $l \in \mathcal{P}$ s.t. $u, w \in l$. A *path* $\pi = (v_1, \ldots, v_n)$ with $n \geq 0$ in $\mathcal{P}$ is a sequence of vertices s.t. for all $i < n$, the vertices v_i, v_{i+1} are connected. We call π *empty* if its length is 0, *trivial* if its length is 1, *atomic* if it is 2, and remark that in the latter case π crosses exactly one link. We shall write $u \sim v$ when there is a path from u to v.

Definition 2.7 (Basic path operators). Given $\pi = (v_1, \ldots, v_n)$ and $\phi = (u_1, \ldots, u_m)$ in $P(\mathcal{N})$, we denote the *reversal* of π by $\pi^{\dagger} = (v_n, \ldots, v_1)$. If $v_n = u_1$ then the *concatenation* of ϕ to π is defined as $\pi :: \pi' = (v_1, \ldots, v_n = u_1, \ldots u_m)$. If $\pi \in P(\mathcal{N})$ and $\phi \in P(\mathcal{M})$, we say $\pi = \phi$ when $\mathcal{N} = \mathcal{M}$ and, if $\simeq$ is the isomorphism such that $\mathcal{N} \simeq \mathcal{M}$, then $v_i \simeq u_i$, for any $1 \leq i \leq n = m$.

Definition 2.8 (Straight paths). Let $\mathcal{N}$ be a proof-net and π a paths in $\mathcal{N}$. If π crosses consecutively the same link l, then π is called *bouncing*. If l is not a $\star$-link, and π crosses l through v_i, v_{i+1} such that $v_i, v_{i+1} \in C(l)$ or $v_i, v_{i+1} \in P(l)$, then π is *twisting*. When π is not bouncing nor twisting, π is *straight*. Given $u, v \in V(\mathcal{P})$, we say u is *consequent to* v, written $u \leqslant v$, if there is a straight path π from u to v such that all of its links are crossed from a premiss to a conclusion. In such a case, we say π is a *concluding* path. We conversely also say v is *antecedent to* u, written as $v \geqslant u$ or that $\pi^{\dagger}$ is *assuming*.

Definition 2.9 (Switching and cyclic paths [Danos and Regnier, 1989]). A path π in a pre-net $\mathcal{P}$ is *switching* when, for every link in $L(\mathcal{P})$ being $\langle v, v' \ (\multimap) \ u \rangle$, or $\langle V \ (?) \ u \rangle$ with $v, v' \in V$, π does not contain both v, v'. A path $\gamma = (v_0, v_1, \ldots, v_n, v_0)$ for $n > 0$ is called a *cycle*, and any $\pi \supseteq \gamma$ is called *cyclic*.

Definition 2.10 (mMELL proof-nets). Given a net $\mathcal{N}$, let $\mathcal{L}(\mathcal{N})$ be the net obtained by interpreting each box as a single link, i.e. such that we can traverse it with a unitary path. Then, $\mathcal{N}$ is a *mMELL proof-net* if any switching path in $\mathcal{L}(\mathcal{N})$ or in $\mathcal{L}(\mathcal{B})$, for any $\mathcal{B} \in \text{Bxs}(\mathcal{N})$, is acyclic.

2.2.3 Lambda terms and nets

Definition 2.11 (λ-terms). Let $\mathbb{V}$ be the grammar of a denumerable set of *variable* symbols $x, y, z, \ldots$, and let $\star$ be a constant dummy value. Then, the set Λ of terms is generated by the following grammar.

$$\mathbb{T} ::= \star \mid \mathbb{V} \mid \lambda \mathbb{V}.\mathbb{T} \mid (\mathbb{T} \ \mathbb{T}) \tag{2.3}$$

Definition 2.12 (Term translation). Given $t \in \Lambda$, let the i-th free occurrence of a variable x appearing in t to be uniquely indexed with a natural number as $x_{@i}$, with $1 \leq i \leq m$ for m being the number of occurrences. Let Γ be an injection between the variable occurrences in t and a set of vertices V'. The *translation* $[\![t]\!]^{\Gamma}$ is a mMELL structure whose vertices is a superset of V' and that is defined in Figure 2.2. The actual work is performed by the pre-translation of t, denoted as $\langle t \rangle^{\Gamma}$, which goes by induction on the syntax of t; whilst the final step only adds a ?-link linking all occurrences of a given free variable x, for all free variables of t. More precisely, the (pre-)translation is such that

Figure 2.2 Pre-translation $\langle \, \rangle$ and translation $[\![\,]\!]$ of λ-terms into mMELL nets
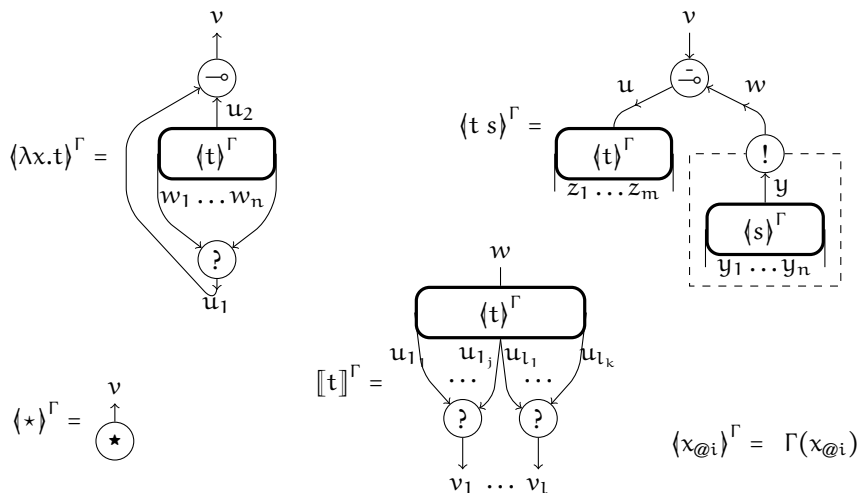
two vertices v, u are premisses of the same ?-link if and only if $\Gamma^{-1}(v) = \Gamma^{-1}(u)$. Since the choice of Γ produce no change in the translation, we shall omit to specify it.

Proposition 2.1. *Any translation is a mMELL proof-net.*

Proof. See the detailed work by Regnier [1992, Proposition 3.2.1]. ∎

Definition 2.13 (Variables). The *free variables* of a proof-net $\mathcal{N}$ is the set $\mathrm{FVar}(\mathcal{N})$ of in-vertices of $\mathrm{iface}(\mathcal{N})$, whilst $\mathrm{BVar}(\mathcal{N})$ is the set of vertices which are connected to the first premiss of a $\multimap$-link in $\mathrm{L}(\mathcal{N})$, and which are called *bounded variables*. The set of *variables* is then $\mathrm{FBVar}(\mathcal{N}) = \mathrm{BVar}(\mathcal{N}) \cup \mathrm{FVar}(\mathcal{N})$.

2.3 Proof-net reductions

Here we introduce the ordinary proof-net reduction recalling its most notable properties, and the closed variant, which acts on a box only if has no secondary doors, proving that in our setting such restriction causes no loss of generality.

2.3.1 General notions

Notation 2.1 (Rewriting). We fix some quite usual notational conventions and terminology we shall employ for rewriting notions. Given a rewriting relation $\rightarrow$ on a set A, the symbols $\rightarrow^+$ and $\rightarrow^*$ respectively denote the transitive and the transitive-reflexive closures of $\rightarrow$. Given $a, a' \in A$, if $a \rightarrow a'$ (resp. $a \rightarrow^* a'$) we say that there is a rewriting step (resp. sequence) from the reducendum a to the reduct a'. Also, if a sequence is made of k steps, we write $\rightarrow^k$. We write $a \not\rightarrow$ and say that a is a normal form, when there exists no a' such that $a \rightarrow a'$. If $a \rightarrow^* a' \not\rightarrow$, then we say that a' is a normal form of a; if a' is unique[1] we also write $\mathrm{NF}(a) = a'$. Reduction steps are named with Greek

letters $\rho, \sigma, \tau, \ldots$, and sequences with barred letters, so that we can denote the reduct of a with respect to a step ρ (resp. a sequence $\bar{\rho}$) as $\rho(a)$ (resp. $\bar{\rho}(a)$).

Definition 2.14 (Context and pre-substitution). A *hole-link* is a link with arbitrary arity, polarity and types. A *single-hole pre-context* $C[\,]$, or simply a pre-context, is a pre-net whose links contains exactly one hole-link h, and whose *internal interface* is the interface of h.

A *mMELL context* is a single-hole pre-context made of mMELL links and equipped with a typing $\mathcal{T}$ and a boxing b. Given a mMELL context $C[\,]$ and a mMELL pre-net $\mathcal{P}$ whose interface is identical to the internal interface of $C[\,]$, the *pre-substitution* of the pre-net in the context, written $C[\mathcal{P}]$, is the pre-net obtained as follows.

1. Replace the hole link in $C[\,]$ with $\mathcal{P}$.

2. Given a bijection $\leftrightarrow$ between vertices of the internal interface of C and those of the interface of $\mathcal{P}$; for any $v \in C$ and any $v' \in \mathcal{P}$, if $v \leftrightarrow v'$, then in $C[\mathcal{P}]$ the two vertices are equated, and we write $v \equiv v'$.

Definition 2.15 (Proof-net reduction). Given a proof-net $\mathcal{N}$ and a cut vertex, its *redex* is a subnet $R \subseteq \mathcal{N}$ containing it, depending on its principal type, so that $\mathcal{N} = C[R]$. For both type constructors we have, we now define: the shape of a redex R; the relation of reduction between R and its reduct R', for which we write $R \to R'$; the notion of context-closure of the relation, i.e. the definition of $C[R']$.

Linear implication The redex sub-net is made by the two links insisting on the cut vertex. Its reduction is depicted in Figure 2.3a and written as:

$$\langle v', v \,(\multimap)\, w \rangle, \langle u', u \,(\bar{\multimap})\, w \rangle \quad \to \quad [v \equiv u, v' \equiv u']. \tag{2.4}$$

The notation $v \equiv u$ denotes the fact that the vertices v, u have been equated. The substitution of the reduct in a mMELL contex is simply its pre-substitution.

Exponential In this case, the redex sub-structure R includes not only the two exponential links $\langle u_1, \ldots, u_k \,(?)\, c \rangle$ and $p = \langle v \,(!)\, c \rangle$, but also the box $\langle W \,[\mathcal{B}]\, v \rangle$ that is connected to the cut !-link, together with the set of every link $\langle X_i \,(?)\, y_i \rangle$ such that there exists $w \in W$ that also belongs to X_i. The redex, together with two exemplifying boxes $C, \mathcal{D}$ which are not part of it, are drawn on the left of Figure 2.3b, the reduct is on the right, and both are also written as follow.

$$\left.\begin{array}{r}\langle u_1, \ldots, u_k \,(?)\, c \rangle, \langle v \,(!)\, c \rangle, \\ \langle W \,[\mathcal{B}]\, v \rangle, \\ \langle X_1 \,(?)\, y_1 \rangle, \ldots, \langle X_s \,(?)\, y_s \rangle\end{array}\right\} \to \left\{\begin{array}{l}\langle W_1 \,[\mathcal{B}_1]\, u_1 \rangle, \ldots, \langle W_k \,[\mathcal{B}_k]\, u_k \rangle, \\ \langle X_1' \,(?)\, y_1 \rangle, \ldots, \langle X_s' \,(?)\, y_s \rangle\end{array}\right. \tag{2.5}$$

The reduction removes the two cut exponential links, and duplicates $\mathcal{B}$ into k copies (erasing it when $k = 0$): $\langle W_1 \,[\mathcal{B}_1]\, u_1 \rangle, \ldots, \langle W_k \,[\mathcal{B}_k]\, u_k \rangle$. Also, the boxing b' of the reduct is obtained by the boxing b of R so that the box copy $\mathcal{B}_h$ is included in any box that in R contains v_h. More precisely:

---

[1]It is the case for all kind of reductions that are considered here.

Figure 2.3 Proof net reductions

(a) Linear implication
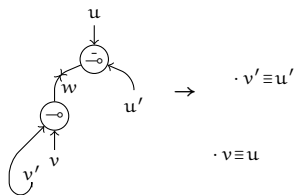
(b) Exponential
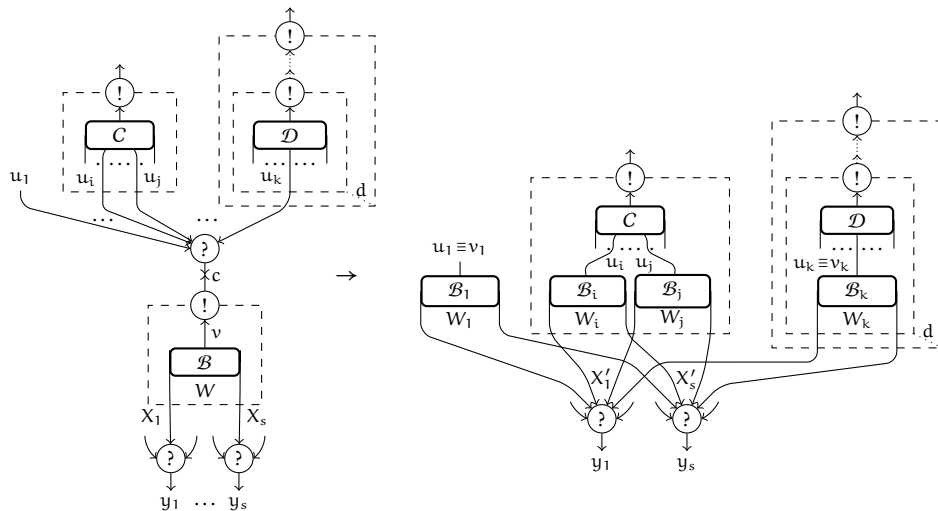
1. if p' is the !-link of a box $\mathcal{B}$ containing u_h for some $0 \leq h \leq k$, then $b'(p') = b(p') \cup \mathcal{B}_h$;

2. if $p' \neq p$ is the !-link of a box $\mathcal{B}' \not\subseteq \mathcal{B}$ then $b'(p'_h) = b(p')$, for any $0 \leq h \leq k$;

3. otherwise $b'(p') = b(p')$.

Notice that for any $0 \leq j \leq s$, the set of premisses of the j-th ?-link containing auxiliary doors for $\mathcal{B}$ is modified by the reduction: the sequence of premisses X'_j of its reduct is obtained by replacing any occurrence of a vertex $w \in W$ with the sequence $(w_1, \ldots, w_k)$ such that its elements respectively belongs to $W_1, \ldots, W_k$.

The substitution of the R' in $C[\ ]$ is defined as the pre-substitution of R' in C, plus a modification of its boxing so that the sub-net $\mathcal{B}_i$ is included to any $\mathcal{B}'$ containing v_i. Pictorially: observe box borders in the upper part of Figure 2.3b. Formally: if for some $0 \leq i \leq k$ and according to $b(C)$, we have $u_i \in \mathcal{B}' \in \text{Bxs}(\mathcal{N})$ and $z \in V(\mathcal{B})$, then in the boxing of the reduct $b'(C)$ we have also $z_i \in V(\mathcal{B}')$. In spite of this detail, we shall abuse the notation and simply write that $C[R] \to C[R']$.

The *proof-net reduction*, also called *cut elimination*, is the graph-rewriting relation on mMELL nets obtained by the union of the context closures of the linear implication reduction and the exponential reduction.

Proposition 2.2. *Any reduct of a proof-net is a proof-net.*

We only sketch the main idea of the proof, and address the interested reader to the well-detailed proof by Regnier [1992, Proposition 4.1.1].

Proof sketch. Consider a reduction step ρ on a redex $R \subset \mathcal{N}$ and take a switching path π that is persistent to ρ. Ad absurdum, suppose that $\rho(\pi)$ is cyclic, and observe that this would imply that π is cyclic as well, thus contradicting Definition 2.10. Hence $\rho(\pi)$ is acyclic. Repeat such argument by induction on the length of the reduction sequence $\mathcal{N} \to^* \mathcal{N}'$, and obtain the thesis. ∎

Proposition 2.3 (Strong normalisation [Pagani and Tortora de Falco, 2010, Accattoli, 2013]). *Any mMELL proof-net $\mathcal{N}$ strongly normalises. If $\mathcal{N}$ has a unique conclusion vertex $v : T$, with T ground, then the normal form of $\mathcal{N}$ is $\langle (\star) \, v \rangle$.*

2.3.2 Closed strategy

Closed reduction perform exponential steps only if the box is closed, i.e. has no auxiliary doors (cf. Definition 2.5).

Definition 2.16 (Closed reduction). The *closed reduction*, written $\to_{cl}$, is the restriction of $\to$ that for the exponential case requires that the box in a redex to be closed.

Lemma 2.1 (Existence of closed exponential cut). *Given a mMELL proof-net $\mathcal{N} : \star$ without $\multimap$-cuts, either $\mathcal{N}$ is cut-free, or $\mathcal{N}$ contains an exponential cut on a closed box.*

Proof. Given $\mathcal{N}$ a mMELL proof-net without $\multimap$-cuts, let $c \in V(\mathcal{N})$ be an exponential cut such that $\ell(c) = 0$, and $\mathcal{B}$ be the box whose !-link has conclusion c. We proceed by induction on the number n of boxes at depth 0 crossed by π, which is finite thanks to the finiteness of $\mathcal{N}$.

1. If $n = 0$ then $\mathcal{B}$ is closed, so we found a witness and we conclude.

2. If $n > 0$, let a be the conclusion of a ?-link having a premiss that is an auxiliary door of $\mathcal{B}$. Consider any straight and consequent path π starting from a. π cannot reach the conclusion of $\mathcal{N}$, because otherwise the type on $\mathcal{N}$ would differ from $\star$ (contradicting such hypothesis). π can neither reach a $\star$-link, since otherwise the typing of links would be broken somewhere along π (Definition 2.2). Therefore π must reach a cut vertex c', whose depth by hypothesis cannot be smaller than the depth of c. But the depth of vertices along π can neither increase, because π is consequent. Therefore $\ell(c') = \ell(c)$. Morever, it must be the case that $c \neq c'$, since $\mathcal{N}$ is switching cyclic by Definition 2.10. Now repeat the previous reasoning right from the beginning, using the cut c' as our target cut instead of c, and the box $\mathcal{B}'$ that is associated with c' in place of the target box $\mathcal{B}$.

∎

Remark 2.1. Previous Lemma 2.1 holds a little more generally in full MELL, with the assumption that a proof-net does not contain any ?-link in any subtype of any of its conclusions. Such assumption is the same that was firstly used by Girard [1989, Theorem

1, p. 239], and in our syntactical setting it implies that the only conclusion of the net has ground type.

Fact 2.1. *For any given proof-net N, if N has a normal form $\overline{N}$ with respect to full mMELL reduction, then there exist a $N \to^*_{\text{cl}} \overline{N}$*

Proof. The claim follows from normalisation (Proposition 2.3), confluence of the ordinary reduction, and the existence of closed exponential cuts (Lemma 2.1). ∎

2.4 Execution paths

In this section we formalise the notion of execution paths, the action of reduction on them. Together with the property of persistence, i.e. the ability of resisting to the rewriting, we show that every path has a unique ancestor. We also present the closed reduction on proof-nets, which acts on a box only if it is closed, and prove that in this case path reduction induces a bijection.

2.4.1 Statics

We now introduce two restrictions on the shape of paths that rule out those which, from a proof-theoretic or computational perspective, we can a priori recognise as meaningless. We want paths which do not bounce, nor twist in a proof-net. A third restriction is instead unnecessary, but considerably improve the simplicity and readability.

Except for a cosmetic difference discussed in Remark 2.2, our formulation is essentially a rework of the presentation given by Danos and Regnier [1995].

Definition 2.17 (Execution paths)**.** Let N be a proof-net and π a paths in N. If there is no other path $\pi' \in N$ such that $\pi \subseteq \pi'$, where $\subseteq$ is the inclusion ordering on sequences, then π is *maximal*. Finally if π is both straight and maximal, then π is an *execution path*. We denote with $\mathbf{P}(N)$ the set of straight paths in N, whilst $\mathbf{P}_{\mathsf{E}}(N)$ is the set of execution paths.

Fact 2.2. *If v is the extremum vertex of an execution path $\pi \in N$, then v is either a conclusion of N, or the conclusion of a weakening.*

Remark 2.2. Most of the previous literature about paths in LL proof-nets uses the notion of 'composition' instead of 'concatenation'. As a result, the appropriate notation in functional style has been preferred, i.e. writing from right to left. We rather preferred the cognitive ease to orthodoxy[2].

Remark 2.3. For the ease of presentation, we deliberately left a bit of ambiguity in the path definition. It may happen that, given an ordered pair of vertices, there actually exist two distinct hyperlink crossings. For instance, consider a pre-net having $l = \langle v\ (?)\ u \rangle$ and $l' = \langle u, v\ (\multimap)\ w \rangle$, and a unitary path $\pi = (u, v)$. What is the link crossed by π? In spite of this, all the possible ambiguities will be clarified either by the straightness of the paths we will consider almost everywhere, or by a direct explanation. For instance,

---

[2].notation favourite its recover easily should writing inverse the with accustomed reader The

again within the net considered in the previous example, consider $\pi' = (w, u, v, w)$ and notice that, if π' is straight, then there is a unique sequence of links crossed by π', i.e. (l', l, l'). Indeed, the only other possible sequence of links, i.e. (l', l', l'), would imply π' being twisting.

2.4.2 Dynamics

We now define the action of reduction on paths and the notion of persistence, and show some elementary properties, the most notable of which is the fact that every path in a reduct has a unique ancestor.

Definition 2.18 (Redex crossing and sufficient length). Given a reduction step ρ on a redex $R \subseteq \mathcal{N}$, a *redex crossing* for R is a straight path χ that is maximal in R. Let ρ be a reduction step on a redex R in $\mathcal{N}$. A path $\pi \in \mathcal{N}$ is *long enough* if and only if for any vertex $r \in \pi$, the fact that $r \in R$ and r is not a conclusion of R implies there is a subpath $\chi \subseteq \pi$ that is a crossing of R, and such that $r \in \chi$. If π is long enough for ρ, then there exist $n \geq 0$ such that
$$\pi = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n :: \pi_n,$$
where for any $0 \leq i \leq n$ the sub-path χ_i is a crossing for R, whilst π_i is not. This unique form of expressing π is called its *redex crossings form* for R, and written $\mathrm{RCF}_R(\pi)$.

Definition 2.19 (Straight crossing reduction). Let $\mathcal{N}$ be a proof-net where $\chi \in \mathbf{P}(\mathcal{N})$ is a crossing of a redex R, and for which ρ is the reduction step on R. The *reduction of* R is a set of paths in $\rho(\mathcal{N})$ defined as follows and denoted as $\rho(\chi)$.

Linear implication Let R be as in Figure 2.3a or equivalently, as in Equation 2.4, Then the *residual of* χ *with respect to* ρ, is defined as follows and by the rule $\rho(\chi^\dagger) = (\rho(\chi))^\dagger$.

$$\rho((v, w, u)) \stackrel{\mathrm{def}}{=} \{(v)[v \equiv u]\} \tag{2.6}$$

$$\rho((v', w, u')) \stackrel{\mathrm{def}}{=} \{(v')[v' \equiv u']\} \tag{2.7}$$

$$\rho((v, w, u')) \stackrel{\mathrm{def}}{=} \varnothing \tag{2.8}$$

$$\rho((v', w, u)) \stackrel{\mathrm{def}}{=} \varnothing \tag{2.9}$$

Exponential Since χ necessarily traverses the box $\mathcal{B}$ whose principal door is cut, let β be the maximal sub-path of χ whose vertices are in $\mathcal{B}$, and let β_j be the copy of β traversing $\mathcal{B}_j$, that is the j-th copy of $\mathcal{B}$. We isolate three kinds of crossings, depending on the polarities (in or out) of the two extrema of β.

out-out

$$\rho\big((u_j, c, v) :: \beta :: (v, c, u_j)\big) \stackrel{\mathrm{def}}{=} \{(u_j) :: \beta_j :: (u_j) [u_j \equiv v_j]\}; \tag{2.10}$$

$$\rho\big((u_j, c, v) :: \beta :: (v, c, u_{j'})\big) \stackrel{\mathrm{def}}{=} \varnothing. \tag{2.11}$$

out-in Let $1 \leq j \leq k$ and $1 \leq l \leq s$. Then

$$\rho\big((u_j, c, v) :: \beta :: (w_l, y_l)\big) \stackrel{\mathrm{def}}{=} \{(u_j) :: \beta_j :: (w_{l_j}, y_l) [u_j \equiv v_j]\}. \tag{2.12}$$

in-in Let $1 \le l, m \le s$. Then

$$\rho\big((y_l, w_l) :: \beta :: (w_m, y_m)\big)$$
$$\stackrel{\mathrm{def}}{=} \big\{(y_l, w_{l_j}) :: \beta_j :: (w_{m_j}, y_m) \,\big|\, 1 \le j \le k\big\}. \tag{2.13}$$

Definition 2.20 (Path reduction and persistence). Let $\pi \in P(N)$ long enough for a redex $R \subset N$ and suppose

$$\mathrm{RCF}_R(\pi) = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n :: \pi_n.$$

The *reduction* of π with respect to the reduction step ρ on R, written $\rho(\pi)$, is the function from paths to sets of paths obtained by concatenation of sets of reducts of any redex crossing, distributing set membership over path concatenation. Formally:

$$\rho(\pi) \stackrel{\mathrm{def}}{=} \big\{\pi_0 :: \chi_1' :: \pi_1 :: \ldots :: \chi_k' :: \pi_k \,\big|\, \chi_1' \in \rho(\chi_1), \ldots, \chi_k' \in \rho(\chi_k)\big\}. \tag{2.14}$$

Observe, in particular, that, if $\rho(\chi_i) \ne \varnothing$ for all $1 \le j \le n$, then also $\rho(\pi) \ne \varnothing$ and π is said to not be *persistent to* ρ. Otherwise $\rho(\pi) = \varnothing$ and π is not *persistent to* ρ. Moreover, if for every reduction sequence $\sigma = (\rho_1, \ldots, \rho_m)$, and for every $1 \le j \le m$, the path π is persistent to ρ_j, then π is said *persistent*. In order to make path reduction uniform, we extend its domain to sets of paths. Given a set of paths Π long enough for R, we then write $\rho(\Pi)$ to denote $\{\rho(\pi) \mid \pi \in \Pi\}$.

Fact 2.3 (Reduction preserve straightness). *For any path π in a proof-net N, and any reduction ρ, if π is straight then any $\pi' \in \rho(\pi)$ is straight.*

Lemma 2.2. *Given N mMELL proof-net and ρ reduction step, for any $\pi \in P_E(\rho(N))$ there exists a unique $\phi \in P_E(N)$ such that $\pi \in \rho(\phi)$.*

Proof. Let $N = C[R]$, for some redex R and some context $C[\,]$. Suppose also that $\pi = \pi_0 :: \chi_1' :: \pi_1 :: \ldots \chi_n' :: \pi_n$, where, for any $1 \le i \le k$, the subpath χ_i' is a maximal sequence of vertices belonging to $\rho(R)$. Observe that the extrema of χ_i' necessarily belong to the interface of $\rho(R)$, because π is an execution path. Now, by definition of reduction, and in particular as per Equation 2.14, if $n = 0$ (i.e. π does not cross $\rho(R)$), then there is nothing to prove, since by definition of reduction we have that $\pi \in P_E(N)$ and $\rho(\pi) = \{\pi\}$. So, suppose otherwise that $n > 0$ and let us discuss χ_i' distinguishing two cases depending on the kind of the redex.

1. $\mathcal{T}(R) = \multimap/\overset{..}{\multimap}$. Because of the persistence of π to ρ, and by Equation 2.14, we have $\chi_i' \ne \varnothing$. Let ρ is as in Figure 2.3a. We observe that in this sub-case we have indeed a bijection:

   (a) $\chi_i' = (v)$ and $\rho(\chi_i) = \{\chi_i'\}$ if and only if $\chi_l = (v, w, u)$;
   (b) $\chi_i' = (v')$ and $\rho(\chi_i) = \{\chi_i'\}$ if and only if $\chi_l = (v', w, u')$.

   $\square$

2. $\mathcal{T}(R) = !/?$. Let ρ be as in Figure 2.3b and recall that $\rho(R)$ is made by the subnets $\mathcal{B}_j$, for any $1 \le j \le k$, that are copies of the box $\langle W\,[\mathcal{B}]\,c\rangle$ in R. We distinguish three sub-cases depending on the polarity of the extrema of π.

(a) out-out. Let $\chi_i' = (u_j) :: \beta_j :: (u_j)$ for some $1 \leq j \leq k$. Then take $\chi_i = (u_j, c, v) :: \beta :: (v, c, u_j)$, and observe that by definition of reduction (cf. Equation 2.10) χ_i is the only crossing of R such that $\rho(\chi_i) = \{\chi_i'\}$.

(b) out-in. Let $\chi_i' = (u_j) :: \beta_j :: (w_{l_j}, y_l)$. Then take $\chi = (u_j, c, v) :: \beta :: (w_l, y_l)$, and verify again (cf. Equation 2.12) that χ_i is the only crossing of R such that $\rho(\chi_i) = \{\chi_i'\}$.

(c) Let $\chi_i' = (y_l, w_{l_j}) :: \beta_j :: (w_{m_j}, y_m)$ for some $1 \leq l, m \leq s$ (recall that s is the number of ?-links having a premiss in $W_{j'}$ for some $1 \leq j' \leq k$). Then take $\chi = (y_l, w_l) :: \beta :: (w_m, y_m)$ and inspect Equation 2.13 to verify that χ_i is the only crossing of R such that $\chi_i' \in \rho(\chi_i)$.

∎

2.4.3 Closed dynamics

If the reduction is closed, its dynamics gains elegant symmetries which make path reduction induce a bijection on execution paths.

Proposition 2.4 (Closed reduction is bijective on persistent paths). *For any mMELL proof-nets N, N', and reduction step $\rho \colon N \to_{cl} N'$ on a redex R, ρ induces a bijection on the set of persistent crossings of R and the set of crossings of $\rho(R)$.*

Proof. Let $\pi \in P_E(N)$ be persistent to ρ, let R the redex of ρ, and suppose $RCF_R(\pi) = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_k :: \pi_k$. Let $0 \leq l \leq k$ and recall that, by hypothesis of persistence of π w.r.t. ρ, and by definition given by Equation 2.14, we have that $\chi_l \neq \emptyset$. Moreover, $\rho(\pi) = \pi_0 :: \rho(\chi_1) :: \pi_1 :: \ldots :: \rho(\chi_k) :: \pi_k$. There are two reduction rules possibly used by ρ.

1. $\mathcal{T}(R) = \multimap / \bar{\multimap}$. Then, if ρ is as in Figure 2.3, then we immediately find a bijection between redex crossings and their (persistent) reducts:

   (a) $\chi_l = (v, w, u)$ if and only if $\rho(\chi_l) = \{(v)\}$;
   (b) $\chi_l = (v', w, u')$ if and only if $\rho(\chi_l) = \{(v')\}$.

2. $\mathcal{T}(R) = !/?$. Suppose the redex R being as in Figure 2.3. By def. of closed reduction, only the out – out has to be discussed in the exponential case.

   Let β be the maximal sub-path of χ_l that crosses the box $\mathcal{B} \subset R$, and let β_j be the copy of β traversing the j-th copy of $\mathcal{B}$. Now, $\mathcal{B}$ is closed because we hypothesised R to be a redex for the closed reduction. Thus, β is necessarily is crossing of kind out-out. It is then immediate to verify that $\chi_l = (u_j, c, v) :: \beta :: (v, c, u_j)$ if and only if $\rho(\chi_l) = \{(u_j) :: \beta_j :: (u_j) [u_j \equiv v_j]\}$.

Such a bijection holds between χ_l and $\rho(\chi_l)$, so we also have a bijection between π and $\rho(\pi)$. ∎

Remark 2.4 (Bijectivity of paths under non-closed reduction). Observe that every $\star$-typed proof-net normalises to $[\![\star]\!]$, that is inhabited by a unique straight path. In spite

of this, the reduction of a path containing in-in-crossings is a set of paths, whilst the only kind of crossing able to "kill" paths is the out-out one. One may therefore wonder if there are any superfluities in the definition of path reduction that could be cleared. The bijection we proved to relate persistent paths under closed reduction (i.e. Proposition 2.4) holds indeed more generally for ordinary reduction. The reason is that, roughly speaking, somewhere in the net there necessarily exists at least one ?-link which will interact with the box at some point of any normalisation sequence. Morally, all those links will split the set in smaller and smaller partitions up to singletons. Thus, if one would accept to lose locality of the definition by considering the whole path starting from the root of the net, then he could also find the complete sequence of ?-premisses that are crossed before reaching the in-in crossing of interest, because such a sequence identify the unique image of the reduct crossing. The introduced complexity surpasses as largely as regrettably the beauty of the earned property.

2.5 Computation as path execution

2.5.1 Dynamic algebra

Definition 2.21 (Dynamic algebra). The $\mathcal{L}^*$ monoid is defined over terminal symbols in $\{0, 1, p, q, l, r, t, d, \star\}$, where p, q are called multiplicative symbols, and l, r, t, d are called exponential symbols. A word of its alphabet, called *weight*, is generated by a binary *concatenation* operator with infix implicit notation and two unary operators: the *adjoint* $(\cdot)^*$, and the *exponential* $!(\cdot)$ The concatenation operator and the set of symbols has the structure of a monoid, whose identity element is 1, equipped with an additional absorbing element 0. If a, b, c are generic weights:

$$a(bc) = (ab)c \tag{2.15}$$
$$a1 = 1a = a \tag{2.16}$$
$$a0 = 0a = 0 \tag{2.17}$$

Moreover, similarly to an adjoint, the inversion operator is involutive and distributes over concatenation by reversing left and right operands.

$$(a^*)^* = a \tag{2.18}$$
$$(ab)^* = b^* a^* \tag{2.19}$$

The exponential operator, instead, commutes over concatenation and inversion:

$$!0 = 0 \tag{2.20}$$
$$!1 = 1 \tag{2.21}$$
$$(!a)^* = !(a^*) \tag{2.22}$$
$$!(a)!(b) = !(ab) \tag{2.23}$$

and also satisfies two swap and two lift equations interacting with exponential symbols:

$$l!(a) = !(a)l \tag{2.24}$$

$$r!(a) = !(a)r \tag{2.25}$$

$$t!(a) = !!(a)t \tag{2.26}$$

$$d!(a) = ad \tag{2.27}$$

Finally, the core of the computation is encoded by the neutralisation and annihilation equations.

$$pp^* = qq^* = ll^* = rr^* = tt^* = dd^* = 1 \tag{2.28}$$

$$qp^* = pq^* = rl^* = lr^* = 0 \tag{2.29}$$

We say that $c \in \mathcal{L}^*$ is *positive* if c does not contain the inversion operator, and that it is in *stable form* if $c = a \cdot (b)^*$ with $a, b \in \mathcal{L}^*$ are positive. When we need to distinguish $\mathcal{L}^*$-equality from syntactic equality, we write $=_{\mathcal{L}^*}$ for the former and $=$ for the latter. We shall denote $\underbrace{t \ldots t}_{n}$ as t_n, $\underbrace{d \ldots d}_{n}$ as d_n, $\underbrace{! \ldots !}_{n}(a)$ as $!^n(a)$, and $l \underbrace{r \ldots r}_{m}$ as e_m.

Fact 2.4. *The following equations are derivable:*

$$t_n !^m(a) = !^{m+n}(a)t_n \tag{2.30}$$

$$d_n !^m(a) = !^{m-n}(a)d_n \qquad \text{with } n \le m \tag{2.31}$$

$$!^m(a)t_n^* = t^* !^{m+n}(a) \tag{2.32}$$

$$!^m(a)d_n^* = d^* !^{m-n}(a) \qquad \text{with } n \le m \tag{2.33}$$

$$e_n e_n^* = 1 \tag{2.34}$$

$$e_n e_m^* = 0 \qquad \text{with } n \ne m \tag{2.35}$$

Definition 2.22 (Path weighting). The *base weighting* is a map w that associate a weight of $r\mathcal{L}^*$ to an atomic straight path $\pi = (u, v) \in \mathcal{P}(\mathcal{N})$. Straightness implies that it goes either: (i) from a conclusion to a conclusion of a $\star$-link; (ii) from a premiss to a conclusion of a binary link; (iii) vice versa, from a conclusion vertex to a premiss of a binary link. The base weighting is defined as follows, where the first clause covers (i), the clauses from the second to the fifth cover (ii), and the last clause covers (iii).

$$w((u,v)) \overset{\text{def}}{=} \begin{cases} \star & \text{if there is } \langle (\star)\, u \rangle \text{ and } u = v \\ p & \text{if there is } \langle u, w\, (\multimap)\, v \rangle \text{ or } \langle u, w\, (\bar{\multimap})\, v \rangle \\ q & \text{if there is } \langle w, u\, (\multimap)\, v \rangle \text{ or } \langle w, u\, (\bar{\multimap})\, v \rangle \\ 1 & \text{if there is } \langle u\, (!)\, v \rangle \\ dt_n e_i & \text{if there is } \langle u_1, \ldots, u_i, \ldots, u_k\, (?)\, v \rangle, \\ & n = \ell(u_i) - \ell(v), \text{ and } u_i = u \\ (w((v, u)))^* & \text{otherwise.} \end{cases} \tag{2.36}$$

If n is the depth of the link l crossed by π, then the *deepened weighting* is $\bar{w}(\pi) = !^n w(\pi)$. The *path weighting* is the lifting of the deepened weighting map from unitary straight path to generic straight paths.

$$\mathfrak{w}(\varnothing) = \mathfrak{w}(v) \overset{\text{def}}{=} 1 \tag{2.37}$$

$$\mathfrak{w}((u,v) :: \pi) \overset{\text{def}}{=} \bar{w}(u,v)\mathfrak{w}(\pi) \tag{2.38}$$

Fact 2.5. *For any paths π, π', path concatenation :: and path reversal $(\ )^\dagger$ commute with their respective algebraic operators $\cdot$ and $(\ )^*$.*

$$\mathbf{w}\left(\pi^\dagger\right) = \left(\mathbf{w}\left(\pi\right)\right)^* \tag{2.39}$$

$$\mathbf{w}\left(\pi :: \pi'\right) = \left(\mathbf{w}\left(\pi\right)\mathbf{w}\left(\pi'\right)\right) \tag{2.40}$$

Definition 2.23 (Regularity and execution). Given a proof-net $\mathcal{N}$, if $\pi \in P(\mathcal{N})$ is both straight and maximal, then is called an *execution path*. The set of execution paths of $\mathcal{N}$ is denoted by $P_E(\mathcal{N})$. If $\mathbf{w}\left(\pi\right) \neq \emptyset$, then π is *regular*. The *execution* of $\mathcal{N}$, written $\mathfrak{E}\mathfrak{x}(\mathcal{N})$, is the sum of the weights of all the execution paths of $\mathcal{N}$, where the sum forms a free commutative monoid on the $\mathbf{r}\mathfrak{L}^*$ structure, and its identity is $\emptyset$. Formally:

$$\mathfrak{E}\mathfrak{x}(\mathcal{N}) \overset{\text{def}}{=} \sum_{\pi \in P_E(\mathcal{N})} \mathbf{w}\left(\pi\right), \tag{2.41}$$

moreover, for any $a, b, c \in (\mathbf{r}\mathfrak{L}^*, +)$:

$$(a + b) + c = a + (b + c) \tag{2.42}$$

$$a + b = b + a \tag{2.43}$$

$$a + \emptyset = a \tag{2.44}$$

Proposition 2.5 (Stability). *Every element in $\mathfrak{L}^*$ has a unique stable form.*

Proof sketch. Let $\rightarrow_{\mathfrak{L}^*}$ be the rewriting system obtained by orienting the equations of $\mathfrak{L}^*$ from left to right. It is locally confluent and terminating. ∎

2.5.2 Equivalence of execution and reduction

Lemma 2.3. *Given a proof-net $\mathcal{N} : \star$, and a closed-reduction step ρ, if $\pi \in P_E(\mathcal{N})$ then:*

$$\mathbf{w}\left(\pi\right) =_{\mathfrak{L}^*} \mathbf{w}\left(\pi'\right) \qquad\qquad \textit{if } \rho(\pi) = \{\pi'\} \tag{2.45}$$

$$\mathbf{w}\left(\pi\right) =_{\mathfrak{L}^*} \emptyset \qquad\qquad \textit{if } \rho(\pi) = \emptyset \tag{2.46}$$

Proof. Let R be the redex of ρ, and recall that maximality of π implies that it is long enough for ρ. So, let

$$\text{RCF}_R(\pi) = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n :: \pi_n, \tag{2.47}$$

where for any $0 \leq i \leq n$, χ_i is a crossing, and π_i is not. By Fact 2.5, we know the $\mathfrak{L}^*$ concatenation operator commutes with path concatenation, so:

$$\mathbf{w}\left(\pi\right) = \mathbf{w}\left(\pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n\right) \tag{2.48}$$

$$=_{\mathfrak{L}^*} \mathbf{w}\left(\pi_0\right)\mathbf{w}\left(\chi_1\right)\mathbf{w}\left(\pi_1\right)\ldots\mathbf{w}\left(\chi_n\right)\mathbf{w}\left(\pi_n\right). \tag{2.49}$$

Now, by Definition 2.20 of closed reduction of path, we already know that $\rho(\pi_i) = \{\pi_i\}$ with $\pi_i = \pi'_i$. Hence trivially $\mathbf{w}\left(\pi'_i\right) = \mathbf{w}\left(\pi_i\right)$ and

$$\mathbf{w}\left(\pi\right) = \mathbf{w}\left(\pi'_0\right)\mathbf{w}\left(\chi_1\right)\mathbf{w}\left(\pi'_1\right)\ldots\mathbf{w}\left(\chi_n\right)\mathbf{w}\left(\pi'_n\right). \tag{2.50}$$

Thus, we only miss to prove that: if $\rho(\chi_i) = \{\chi'_i\}$ then $\mathbf{w}\left(\chi_i\right) =_{\mathfrak{L}^*} \mathbf{w}\left(\chi'_i\right)$; whilst when $\rho(\chi) =$, then $\mathbf{w}\left(\chi\right) =_{\mathfrak{L}^*} 0$. We proceed by case analysis on the kind of cut of ρ.

1. Rule $\multimap$ $/\,\bar{\multimap}$. Let R as in Figure 2.3a. Because of the locality of the reduction step, we can simply consider all the possible crossings, that are four sub-cases (cf. Equation 2.6, 2.7, 2.8 and 2.9).

   (a) If $\chi_i = (v, w, u)$ then we have $\rho(\chi_i) = \{\chi_i'\}$ with $\chi_i' = (v \equiv u)$, and

   $$\mathbf{w}(\chi) = \mathsf{pp}^* =_{\mathcal{L}^*} 1 =_{\mathcal{L}^*} \mathbf{w}(\chi_i')) \tag{2.51}$$

   (b) If $\chi = (v', w, u')$ then we have $\rho(\chi_i) = \{\chi_i'\}$ with $\chi_i' = (v' \equiv u')$, and

   $$\mathbf{w}(\chi) = \mathsf{qq}^* =_{\mathcal{L}^*} 1 = \mathbf{w}(\chi_i') \tag{2.52}$$

   (c) If $\chi_i = (v, w, u')$ then $\rho(\chi_i) = \varnothing$ and

   $$\mathbf{w}(\chi) = \mathsf{pq}^* =_{\mathcal{L}^*} 0 \tag{2.53}$$

   (d) If $\chi_i = (v', w, u)$ then $\rho(\chi_i) = \varnothing$ and

   $$\mathbf{w}(\chi) = \mathsf{qp}^* =_{\mathcal{L}^*} 0 \tag{2.54}$$

2. Rule $!/?$. In this case we need a non-local reasoning and exploit the fact that, by definition of closed reduction (cf. Definition 2.16), the box $\mathcal{B} \subset R$ is closed. This fact, together with the maximality of χ for R, implies that χ_i enters $\mathcal{B}$ if and only if it exits from $\mathcal{B}$, and also that every entrance and exit necessarily happens in the principal door of $\mathcal{B}$. Let R as in Figure 2.3b. In particular let $\chi_i = (u_j, c, v) :: \beta :: (v, c, u_{j'})$, where u_j is a generic premiss of the $?$-link of R (i.e. not necessarily as in the figure). Notice preliminarily that the $?$-link cannot have an empty sequence of premisses, since no straight path traversing the redex can possibly be maximal (cf. Figure 2.3b where $k = 0$). Also, let $l = \ell(u_j) - \ell(c)$, and $l' = \ell(u_{j'}) - \ell(c)$. Consider the weight of χ_i.

   $$\mathbf{w}(\chi_i) = \quad \mathsf{dt_l e_j} \,!(b)\, e_{j'}^* t_{l'}^* d \tag{2.55}$$
   $$=_{r\mathcal{L}^*} \mathsf{dt_l} \,!(b)\, e_j e_{j'}^* t_{l'}^* d \qquad \text{swapping on } e \tag{2.56}$$

   We distinguish two cases depending on the emptiness of $\rho(\chi_i)$.

   (a) If $\rho(\chi_i) \neq \varnothing$, let $\rho(\chi_i')$. Then by Definition 2.19 and in particular by Equation 2.10 it must be the case that $j = j'$, which implies that also $l = l'$. Therefore:

   $$\mathbf{w}(\chi_i) =_{r\mathcal{L}^*} \mathsf{dt_l} \,!(b)\, t_l^* d^* \qquad \text{neutralisation on } e \tag{2.57}$$
   $$=_{r\mathcal{L}^*} \mathsf{d} \,!^{1+l}(b)\, t_l t_l^* d^* \qquad \text{lifting on } t \tag{2.58}$$
   $$=_{r\mathcal{L}^*} \mathsf{d} \,!^{1+l}(b)\, d^* \qquad \text{neutralisation on } t \tag{2.59}$$

   But by definition of path reduction: $\chi_i' = (u_j) :: \beta_j :: (u_j)$. Therefore we conclude:

   $$= \mathbf{w}(\chi_i'). \tag{2.60}$$

   (b) If $\rho(\chi_i) = \varnothing$ then by Definition 2.19 and Equation 2.11 we have $j \neq j'$. Thus:

   $$\mathbf{w}(\chi_i) =_{r\mathcal{L}^*} \mathsf{dt_l} \,!(b)\, 0 t_{l'}^* d^* \qquad \text{annihilation on } e \tag{2.61}$$
   $$=_{r\mathcal{L}^*} 0. \qquad \text{absorbtion on } 0 \tag{2.62}$$

■

Lemma 2.4. *Given a proof-net* $\mathcal{N} : \star$, *and a closed-reduction sequence* $\bar{\rho}$, *if* $\pi \in P_E(\mathcal{N})$ *then:*

$$\mathbf{w}(\pi) =_{\mathcal{L}^*} \mathbf{w}(\pi') \qquad\qquad \text{if } \bar{\rho}(\pi) = \{\pi'\} \qquad\qquad (2.63)$$

$$\mathbf{w}(\pi) =_{\mathcal{L}^*} \mathbf{0} \qquad\qquad \text{if } \bar{\rho}(\pi) = \varnothing \qquad\qquad (2.64)$$

Proof sketch. Lemma 2.3 proved the claim for a single step of reduction. A straightforward induction on the length of the sequence $\bar{\rho}$ suffices to generalise. ■

Theorem 2.1 (Execution invariance). *For any proof-net* $\mathcal{N} : \star$ *and a reduction sequence* $\bar{\rho}$,

$$\mathsf{Ex}(\mathcal{N}) =_{\mathcal{L}^*} \mathsf{Ex}(\bar{\rho}(\mathcal{N})). \qquad\qquad (2.65)$$

Proof. Since closed reduction is normalising (Fact 2.1), let $\bar{\rho}$ be a closed-normalisation such that $\bar{\rho} : \mathcal{N} \rightarrow^*_{\text{cl}} \mathsf{NF}(\mathcal{N})$. Take $\pi \in P_E(\mathcal{N})$ and recall that π is by definition long enough for all possible reductions, including $\bar{\rho}$. So, by Lemma 2.4, if π is persistent to $\bar{\rho}$ then $\mathbf{w}(\pi) =_{\mathcal{L}^*} \mathbf{w}(\bar{\rho}(\pi))$, otherwise $\mathbf{w}(\pi) =_{\mathcal{L}^*} \mathbf{0}$. Recall also that by Proposition 2.4, $\bar{\rho}$ induces a bijective relation on persistent maximal paths, therefore $\mathsf{Ex}(\mathcal{N}) = \mathsf{Ex}(\mathsf{NF}(\mathcal{N}))$. Let $\bar{\rho}'$ be a closed-normalisation on $\mathcal{N}'$ which, by confluence of closed reduction, is such that $\bar{\rho}' : \mathcal{N}' \rightarrow^*_{\text{cl}} \mathsf{NF}(\mathcal{N})$. If we repeat the reasoning above for $P_E(\mathcal{N}')$, we obain that $\mathsf{Ex}(\mathcal{N}') = \mathsf{Ex}(\mathsf{NF}(\mathcal{N}))$. Therefore the claim. ■

Theorem 2.1 of equivalence implies the usual weaker results about GoI for MELL [Regnier, 1992, Théorème 2.2.1, p. 85] [Danos and Regnier, 1995, Theorem 14, p. 325].

Corollary 2.1 (Regularity and persistence). *For any proof-net* $\mathcal{N}$ *and any path* $\pi \in P_E(\mathcal{N})$, π *is persistent if and only if* π *is regular.*

Part I

Superposition and expansion

Chapter 3

Geometry of Resource Interaction

Contents

3.1 Introduction

In this chapter, we consider the resource calculus and its translation into interaction nets, called resource nets, where we develop a quite simple theory of their paths. A suitable restriction of the mMELL algebra is showed to be an extremely concise and precise semantics for the resource calculus.

The material here presented originally appeared in EPTCS [2015].

3.2 Resource calculus

The resource calculus (RC) is, on one hand, a linear and thus finitary restriction of the λ-calculus: in the application t S the function t must use exactly once each s belonging to the multiset of arguments S, i.e. s cannot be duplicated nor erased, so reduction enjoys strong normalisation. On the other hand, RC represent a non-deterministic extension of the λ-calculus, since arguments are now finite multisets of ordinary terms. Therefore, the reduct of t S is defined as the superposition, i.e. a sum, of all the possible ways of substituting each $s \in S$. In particular, the number of arguments provided to a function can be insufficient or excess the function's request, i.e. the number of occurrences of the variable bounded by the abstraction under consideration. In this case, computation is deadlocked and the application reduces to 0, i.e. the empty sum.

3.2.1 Syntax

Definition 3.1 (Simple resource terms and polyterms). The set Δ of all the *simple terms* and the set $\Delta^!$ of all *simple poly terms* are inductively generated by the following grammar.

$$
\begin{array}{rclr}
\mathbb{V} & ::= & x, y, z, \ldots & \text{variables} \quad (3.1) \\
\mathbb{M} & ::= & \star \mid \mathbb{V} \mid \lambda\mathbb{V}.\mathbb{M} \mid (\mathbb{M}\ \mathbb{B}) & \text{simple resource terms} \quad (3.2) \\
\mathbb{B} & ::= & 1 \mid [\mathbb{M}] \mid \mathbb{B} \cdot \mathbb{B} & \text{simple resource polyterms} \quad (3.3)
\end{array}
$$

$\mathbb{V}$ is a denumerable set. $\star$ is the constant dummy value. Brackets delimit multisets (or bags), the multiset sum (associative and commutative) is expressed in multiplicative notation with the central dot $\cdot$, whose neutral element is the empty multiset 1. So that $([x] \cdot 1) \cdot [y] = [x, y]$ is a simple polyterm. Simple terms are denoted by lowercase Latin letters $s, t, u, \ldots$, polyterms by uppercase $S, T, U, \ldots$. The set $\mathbb{N}\langle \Delta \rangle$ of *terms* (resp. the set $\mathbb{N}\langle \Delta^! \rangle$ of *polyterms*) is the set of linear combinations of simple terms (resp. polyterms) having coefficients in the semiring of natural numbers $\mathbb{N}$. All syntactic constructors of simple terms and polyterms to be extended to sums by (bi-)linearity, i.e. to commute with sums.

$$\lambda x.(T + S) \overset{\text{def}}{=} \lambda x.T + \lambda x.S \tag{3.4}$$

$$(s + t)\, U \overset{\text{def}}{=} (s\, U) + (t\, T) \tag{3.5}$$

$$s\,(T + U) \overset{\text{def}}{=} (s\, T) + (s\, U) \tag{3.6}$$

$$[S + T] \cdot U \overset{\text{def}}{=} [S] \cdot U + [T] \cdot U \tag{3.7}$$

In order to increase readability and simplicity, in the rest of the chapter we shall omit the "resource" qualification in the calculus nomenclature.

Example 3.1. A non-simple term could be written as $(\lambda x.(2x + y))[z^2 + 4u]$ to mean: $2(\lambda x.x)[z, z] + 8(\lambda x.x)[u] + (\lambda x.y)[z, z] + 4(\lambda x.y)[u]$.

Remark 3.1. The set of resource (poly-)term was defined by Ehrhard and Regnier [2003] as the finite sums over a generic commutative unit-equipped semiring $\mathbb{R}$, even if they preferred to restrict it to $\mathbb{N}$. Indeed, whenever $\mathbb{R}$ admits negative elements, showed Vaux [2007], strong normalisation of the reduction is lost.

3.2.2 Reduction

Definition 3.2 (Context, redex and reduction). A *simple context* $c(\cdot)$ is a term of the following grammar.

$$\mathbb{C} \quad ::= \quad (\cdot) \mid \lambda \mathbb{V}.\mathbb{C} \mid \mathbb{C}\, \mathbb{B} \mid \mathbb{M}\, [\mathbb{C}] \cdot \mathbb{B} \tag{3.8}$$

A *context* $C(\cdot)$ is a sum of a simple context and a polyterm. A *completion* of a context $C(\cdot)$ with a term t, written $C(t)$, is the substitution of the hole $(\cdot)$ within $C(\cdot)$ for t, possibly capturing variables.

A *redex* is a simple term of the form: $(\lambda x.s)T$. Let the i-th free occurrence of x appearing in s to be uniquely indexed with a natural number as $x_{@i}$, with $1 \leq i \leq m$ for m being the number of occurrences. The *reduction* is the relation $\rightarrow$ between polyterms obtained by the context closure and the linear extension to sum of the following elementary reduction rule.

$$(\lambda x.s)\,[t_1, \ldots, t_n] \;\rightarrow\; \begin{cases} \displaystyle\sum_{\sigma_n \in \mathfrak{S}_n} s\,\{t_1/x_{@\sigma_n(1)}, \ldots, t_n/x_{@\sigma_n(n)}\} & \text{if } n = m \\ 0 & \text{if } n \neq m \end{cases} \tag{3.9}$$

Where $\mathfrak{S}_n$ denotes the set of permutations over the set $\{1, \ldots, n\}$, and $\{t/x\}$ is the usual capture-avoiding substitution.

Example 3.2. Let $I = I' = \lambda x.x_{@1}$ and also let $t = \lambda f.f_{@1}[f_{@2}[\star]]$. Then we have:

$$t[I, I']$$
$$\rightarrow \; f_1[f_{@2}[\star]]\{I/f_{@2}, I'/f_{@2}\} + f_1[f_{@2}[\star]]\{I/f_{@2}, I'/f_{@1}\}$$
$$= \; I[I'[\star]] + I'[I[\star]]$$
$$\rightarrow^2 \; I'[\star] + I[\star]$$
$$\rightarrow^2 \; 2\star$$
$$= \; NF(t[I, I'])$$

Note also a case of annihilation in $t[I] \rightarrow 0$. Finally, observe that if $s = (\lambda x.\star)T \rightarrow \star$ then T must be 1, otherwise $s \rightarrow 0$.

Proposition 3.1. *(Ehrhard and Regnier [2003]) Resource reduction is confluent and strong normalising.*

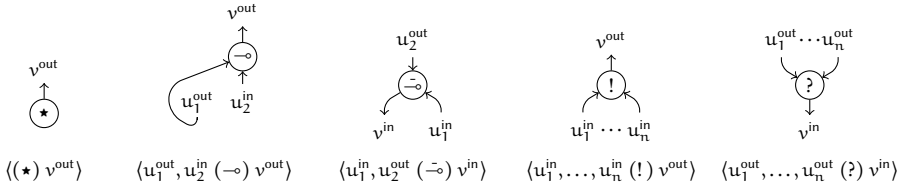## 3.3   Resource interaction nets

From an operational point of view, a net can be seen as a graphical, finer representation of typed terms by means of a variation of the syntax of linear logic proof-nets. From a proof-theoretic point of view, a net is a proof of a minimal, propositional, promotion-free fragment of differential linear logic [Ehrhard and Regnier, 2006a]. The negative exponential links have a symmetrical dual but is deprived of promotion, so that it merely represents superposition of proofs and contexts.

Differential interaction nets (DINs) were introduced by Ehrhard and Regnier [2005] as the graphical computational counterpart of a calculus that is not precisely the *differential* λ-calculus [2003] (that would be the *differential* nets (DNs) Tranquilli [2011]), but instead the *resource* λ-calculus, the latter being a promotion-free restriction of the former. Avoiding this confusion is the reason for adopting, in our nomenclature, the term "resource interaction nets". The syntax of (DINs) has two unary links, dereliction and co-dereliction; two binary links, contraction and co-contraction; and two zero-ary links, weakening and co-weakening. Resource nets are a restriction of DINs, formulated with the "so-called" *nouvelle syntaxe*, where exponential links' arities are factorised to arbitrary arity. The presentation, firstly employed in this setting by Mazza and Pagani [2007] is not just much more compact, since the less are the links, much less are the reduction rules of the system, but it also enables us to nimbly reason modulo associativity, for we have it built-in for free.

3.3.1   Definition

Our presentation of resource nets employs a concise syntax [Mazza and Pagani, 2007] which belongs to the tradition of the so-called *nouvelle syntaxe* for linear logic proof-nets, and many technical solutions are inspired by Tranquilli's careful work [2011] on the more general setting of differential nets, that includes also promotion rule and boxes. Following the traditional presentation we employed also in Section 2.2, we define first the pre-nets as a mere hypergraph-like structure, from which we obtain the nets by

Figure 3.1 Links: kind, arity and polarity associated to vertices, in both graphical and textual notations. From left to right: constant, abstraction, application, co-contraction (or co-weakening when $n = 0$), contraction (or weakening when $n = 0$).

$$\langle (\star)\, v^{out} \rangle \qquad \langle u_1^{out}, u_2^{in}\, (\multimap)\, v^{out} \rangle \qquad \langle u_1^{in}, u_2^{out}\, (\bar{\multimap})\, v^{in} \rangle \qquad \langle u_1^{in}, \ldots, u_n^{in}\, (!)\, v^{out} \rangle \qquad \langle u_1^{out}, \ldots, u_n^{out}\, (?)\, v^{in} \rangle$$

imposing a geometrical, path-based property — the correctness criterion. Comparing our criterion to the one for mMELL proof-nets (see Definition 2.10), and to the original one, discovered by Danos and Regnier [1989] to characterise proof-nets of the multiplicative fragment of linear logic, we begin to observe how close resource nets are to multiplicative ones.

Definition 3.3 (Resource links and pre-nets). A *resource link* is a link (cf. Definition 2.1) that has kind in $\{\star, \multimap, \bar{\multimap}, !, ?\}$ and whose arity and assigned polarities are determined by the link's kind, as shown in Figure 3.1. A *simple resource pre-net* $\mathcal{N}$ is a pre-net (see Definition 2.3, but also Definition 2.2) whose links are resource links. A *pre-net* $\mathcal{P}$ of interface I is a linear combination $c_1 \mathcal{P}_1 + \ldots + c_n \mathcal{P}_n$ of simple pre-nets on the semiring $\mathbb{N}$, where for any $1 \leq i \neq j \leq n$ we have $V(\mathcal{P}_i) \cap V(\mathcal{P}_j) = \varnothing$; moreover, there exists a bijection $\sim$ such that if $I(\mathcal{P}_i) \sim I(\mathcal{P}_j)$ and $(v_i, \mathcal{T}(v_i), p(v_i)) \sim (v_j, \mathcal{T}(v_j), p(v_j))$ where $v_i \in C(\mathcal{P}_i)$ and $v_j \in C(\mathcal{P}_j)$, then $\mathcal{T}(v_i) = \mathcal{T}(v_j)$ and $p(v_i) = p(v_j)$). We shall simply use 0 to denote each of the empty sums of pre-nets having the same interface I, for every interface I. Let $\mathcal{P} = c_1 \mathcal{P}_1 + \ldots + c_n \mathcal{P}_n$ and $\mathcal{Q} = d_1 \mathcal{Q}_1 + \ldots + d_m \mathcal{Q}_m$ be two pre-nets. We say the two pre-nets to be *equal*, written $\mathcal{P} = \mathcal{Q}$, when $n = m$ and there is a bijection $\sim$ between the non-zero addends of $\mathcal{P}$ and those of $\mathcal{Q}$ such that if $c_i \mathcal{P}_i \sim d_j \mathcal{P}_j$, for some $1 \leq i \leq n$ and $1 \leq j \leq m$, then $c_i = d_j$ and $\mathcal{P}_i \simeq \mathcal{Q}_j$.

Definition 3.4 (Resource net). A *simple resource net* is a simple resource pre-net whose switching paths are acyclic (cf. Definition 2.9). A *resource net* is resource pre-net whose addends are simple resource nets. In the rest of the chapter we shall omit the "resource" qualification in the nomenclature of nets, as we did for terms.

Remark 3.2. (Sum's syntax and superposition) The first syntax for resource calculus in the form of interaction nets [Lafont, 1990] was indirectly given by Ehrhard and Regnier [2005], where they defined the more general system of Differential Interaction Nets. Their formal sum of simple nets was formulated as an actual superposition of vertices and arcs (a single vertex may belongs to many simple nets), while our definition explicitly imposes disjointness of any two addends. While Ehrhard and Regnier's formulation is more appealing, since it is conceptually closer to the abstract notion of sum as a superposing operator, their formalisation suffers of two non negligible issues we preferred to avoid.

Firstly, the intuitive definition of path, i.e. a sequence of vertices, does not work in a sum of simple nets. Indeed, without additional constraints, a path would be allowed, from a given addend, to enter a superposed sub-net and then exit in another addend. Secondly, the behaviour of nets reduction as a rewriting system becomes critically more involved and less expressive. Indeed, with actual superposition, the reduction easily

breaks many of valuable properties of interaction nets. For instance, the length of any two reduction sequences between two given nets may differ, and a redex in a simple net is no more necessarily orthogonal to a redex in another addend (as we will show in Proposition 3.5). Even when such a complexity is correctly managed (albeit not effortlessly), the standard notion of reduction turns out to implement a sort of parallel reduction, since it simultaneously acts on all addends sharing a superposed redex, and consequently narrowing its expressive potential. A notable resemblance with this notion can be remarked in Chapter 4, about Taylor-Ehrhard expansion, where we shall define a notion reduction that acts in parallel on any redex that belong to the expansion of the same mMELL redex (Definition 4.7).

3.3.2   Term translation

Resource terms can be mapped into resource nets using the well known call-by-name translation. Similarly to what happens translating λ-terms into mMELL proof-nets, the $\multimap$-link is used for translating λ-abstraction, the $\bar{\multimap}$-link for application, and the ?-link for contracting together all the occurrences of the same variable. In addition to this, we use the generalised !-link for polyterm, and formal sum of nets for... formal sum of terms.

Definition 3.5 (Term translation). Let t be a simple term, and Γ an injection between the variable occurrences in t and a set of vertices V'. The *simple pre-translation* of t, denoted as $\langle t \rangle^\Gamma$, is the simple pre-net defined in Figure 3.2 by induction on the syntax of t and by the following constraint. Two vertices v, u are premisses of the same ?-link if and only if $\Gamma^{-1}(v) = \Gamma^{-1}(u)$. The *simple translation* $[\![t]\!]^\Gamma$ is a pre-net whose vertices is a superset of V' and, again in Figure 3.2, which finalises the pre-translation by adding a ?-link connecting all occurrences of a given free variable x, for all free variables of t. Since the choice of Γ produce no change in the translation, we shall omit to specify it.

The *translation* is the extension of the simple translation to linear combinations.

$$[\![c_1 t_1 + \ldots + c_n t_n]\!] \stackrel{\text{def}}{=} c_1 [\![t_1]\!] + \ldots + c_n [\![t_n]\!], \qquad (3.10)$$

where we recall that for any $1 \le i \neq j \le n$ it must be the case that $V([\![t_i]\!]) \cap V([\![t_j]\!]) = \varnothing$.

The translation is well defined, since its domain is the set of nets. A converse property, more weakly stated about translation and reduction, is presented later in Proposition 3.4.

Proposition 3.2. *The translation of a term is a simple net.*

Proof. Omitted. It suffices to adapt, or to restrict, in fact, the ordinary proof for the homologous proposition for ordinary λ-terms and proof-nets of the multiplicative and exponential fragment of linear logic. (See Proposition 2.1.)   ∎

Remark 3.3. A naive generalisation of the previous result is not possible. A net translation is always defined for simple terms, while it is not for general terms, because of possible incompatibility in the interfaces of translated addends.

Figure 3.2 Simple pre-translation $\langle\,\rangle$ and simple translation $[\![\,]\!]$ of simple terms into simple nets.
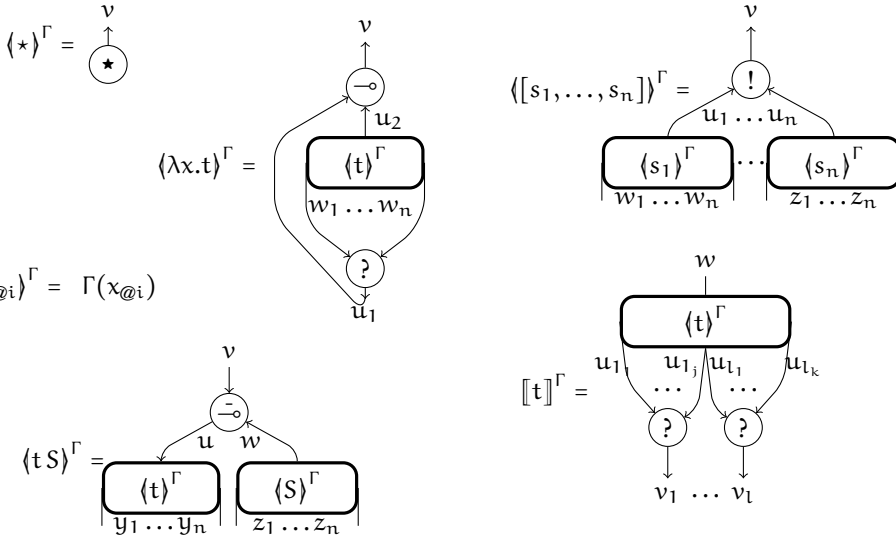
3.3.3 Reduction

Definition 3.6 (Resource permutations). Given a simple pre-net $\mathcal{P}$, a *resource permutation* $\sigma_{\mathcal{P}}$ is a function from the set of !-links in $\mathcal{P}$, to $\bigcup_{n\in\mathbb{N}}\mathfrak{S}_n$, where $\mathfrak{S}_n$ is the group of permutations over the set $\{1,\ldots,n\}$, such that: if a !-link l has arity m, then $\sigma_{\mathcal{P}}(l)$ is an element σ_m of $\mathfrak{S}_m$. We shall write σ_l for $\sigma_{\mathcal{P}}(l)$ and denote the set of resource permutation of $\mathcal{P}$ as $\mathfrak{S}_{\mathcal{P}}$. In particular, if $\mathcal{P}$ contains no !-links, then $\mathfrak{S}_{\mathcal{P}}$ is an empty function and the empty set its codomain.

Definition 3.7 (Contexts). A *simple resource context* $C[\,]$ is a single-hole pre-context (see Definition 2.14) made of resource links, and a *resource context* $\boldsymbol{C}[\,]$ is the sum of a simple resource context C and a resource pre-net $\mathcal{P}$.

Given $\mathcal{P}$ a simple pre-net and $C[\,]$ a simple context with hole link h, the *substitution* of the former in the latter, written $C[\mathcal{P}]$, is defined whenever the interface of $\mathcal{P}$ is the same as the internal interface of $C[\,]$, and is equal to the pre-substitution $C[\mathcal{P}]$ (see Definition 2.14) which replaces $\mathcal{P}$ for h. The substitution of non-simple nets is the extension to linear combinations:

$$C[0] \stackrel{\text{def}}{=} 0; \tag{3.11}$$

$$C[c\,\mathcal{P} + \mathcal{P}] \stackrel{\text{def}}{=} c\,C[\mathcal{P}] + C[\mathcal{P}]; \tag{3.12}$$

$$\boldsymbol{C}[\mathcal{P}] \stackrel{\text{def}}{=} C[\mathcal{P}] + \mathcal{P}, \qquad \text{when } \boldsymbol{C}[\,] = C[\,] + \mathcal{P}. \tag{3.13}$$

We stress that in rightmost side of (3.12) the two copies of C are distinct, since sums of simple pre-nets are always disjoint.

Definition 3.8 (Redex and reduction). Recall that a vertex w in a simple pre-net is a cut if it the conclusion of two links l, l'. The *redex* of w is the pair l, l'. The *reduction* $\to_r$ is the graph-rewriting relation on pre-nets defined by closing with respect to contexts the relation given by the two elementary rules that we are going to define, which maps

Figure 3.3 Reduction rules: linear implication and exponential.
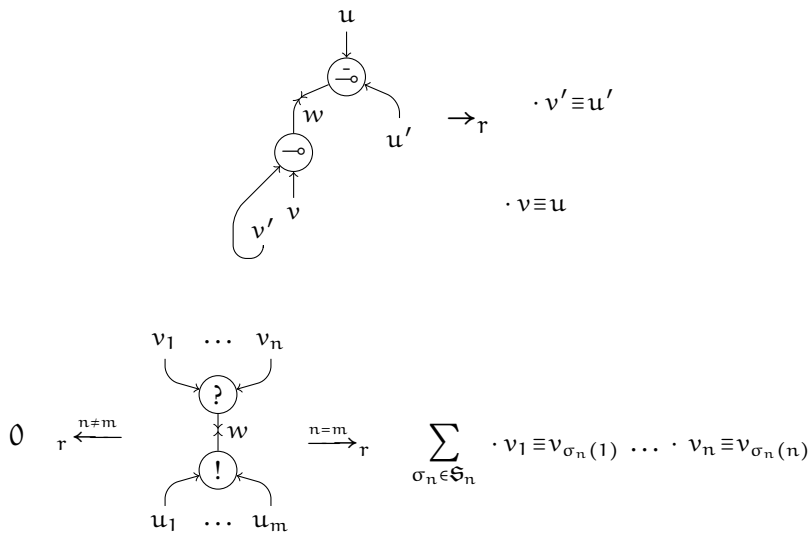
simple pre-nets into pre-nets. Namely, if according to the elementary rules $\mathcal{P} \rightarrow_r \mathcal{P}$ (where $\mathcal{P}$ may be a simple net), then $C[\mathcal{P}] \rightarrow_r C[\mathcal{P}]$, for any context $C[\ ]$. We shall write $[v \equiv u]$ to denote as usual the fact that the two vertices have been equated. Other notational conventions about rewriting have been previously fixed in Notation 2.1.

Linear implication Identically to mMELL case (see Definition 2.15), the redex reduction is defined as follows and depicted on the left of Figure 3.3.

$$\langle v', v \ (\multimap) \ w \rangle, \langle u', u \ (\bar{\multimap}) \ w \rangle \rightarrow_r [v' \equiv u', v \equiv u]. \tag{3.14}$$

Exponential This case introduces a sum of simple resource nets spanning over any possible permutation between in and out premises of the involved links and is empty in case of a mismatch. It is formally drawn on the right of Figure 3.3 and textually written as follows.

$$\langle v_1, \dots, v_n \ (!) \ w \rangle, \langle u_1, \dots, u_m \ (?) \ w \rangle$$

$$\rightarrow_r \begin{cases} \displaystyle\sum_{\sigma_n \in \mathfrak{S}_n} [v_1 \equiv v_{\sigma_n(1)}, \dots, v_n \equiv u_{\sigma_n(n)}] & \text{if } n = m; \\ 0 & \text{if } n \neq m. \end{cases} \tag{3.15}$$

We can now appreciate not only the well-definedness of reduction, but also its expressivity. We outline, along one direction, that the reduction preserve the definition of RINs, i.e. a reduct is a net. Conversely, we see that any RIN is obtainable from the translation of a resource λ-term by reflexive and transitive closure of reduction.

Proposition 3.3. *The reduct of a net is a net.*

Proof sketch. Consider a reduction step ρ on a redex R in a net $\mathcal{N}$, and take a switching path π that is persistent to ρ. Ad absurdum, suppose that $\rho(\pi)$ is cyclic, and observe that this would imply that π is cyclic as well, thus contradicting Definition 3.4. Hence $\rho(\pi)$ has to be acyclic and $\rho(\mathcal{N})$ is a net. ∎

Proposition 3.4. *For any net N there exists a (poly-)term* t *such that* $[\![t]\!] \to^* N$.

Proof sketch. From a proof-theoretic point of view, RINs are very close to the minimal multiplicative fragment of LL. Then, standard sequentialisation techniques developed for the latter can be easily applied here.

We can go by induction on the number of addends appearing in N. For any N in $\mathcal{N}$ it is possible to infer at least one sequence of anti-reduction that goes back to a net N' which contains no exponential redexes. It is then viable to go by structural induction on N' by considering, at each step, the link $l \in L(N)$ (if any) that is connected with the root of N (i.e. its unique out conclusion). In the inductive steps, one necessarily find that l is either a $\multimap$ or a $\bar{\multimap}$ link. Moreover, because of typing constraints, the first premiss of l is necessarily the conclusion of an exponential link l', respectively of kind ? or !. These two cases correspond to the two key rules of pre-translation, hence the inductive step can be proved by inductive hypothesis on all sub-nets of the net obtained by removing l, l' from N'. ∎

Example 3.3. Consider $\delta = \lambda x.(x_{@1}[x_{@2}])$ and notice $[\![\delta]\!]$ is not defined, because of the lack of adequate typing function. Recall the terms $I = \lambda x.x_{@1}$ and $t = \lambda f.(f_{@1}[(f_{@2}[\star])])$ from Example 3.2; we now explain Figure 3.4, which shows the translations of these terms as nets, and provides an example of net reduction. Above on the left: $[\![I]\!]$ is closed and $[\![I]\!] : !\star \multimap \star$. On the right: $N : \star$ is not a translation of a term, but it is a net, because $[\![t[x, y]]\!] \to N$ by eliminating a linear implication cut. Also, N is not a closed net, because it has three conclusions: v_1, z_1, z_2. Below: an exponential reduction step involving index permutation, that rewrites N as a sum of two normal simple nets. In order to stress the fact that addends do not share vertices, those of the rightmost addend have been labelled differently from the leftmost one. Observe the reduct is equal to $[\![x[y[\star]] + y[x[\star]]]\!]$.

Consider $[\![\lambda f.f_{@1}[f_{@2}[\star]][I, I]]\!]$, that is a closed net of type $\star$, and observe the reduct M of the only linear implication cut that is depicted in Figure 3.5 (with the same vertex notation as before). The normalisation requires: one exponential step (above), two linear implication steps per addend (below), and finally two other exponential steps per addend (omitted) to reach the net $\langle(\star) v_1 \equiv v_8\rangle + \langle(\star) v_1' \equiv v_8'\rangle = [\![2\star]\!]$.

Lemma 3.1. *Given a closed net N, let $v \in V(N)$ that is not a cut, nor in $C(N)$. Then v is the conclusion of an exponential link if and only if v is the first premiss of a multiplicative link.*

Proof. Given Proposition 3.4, we can proceed by induction on the length of the reduction sequence $\bar{\rho} : [\![t]\!] \to^* N$, for some term t.

1. *Base.* Suppose $|\bar{\rho}| = 0$. Observe first, that the syntax of terms (Definition 3.1) and the typing constraints (Definition 2.2,) of their translation into nets (Definition 3.5), forces $[\![t]\!]$ to have the out conclusion r typed either with $\star$ or with $\multimap$, otherwise t would be a polyterm, contradicting the hypothesis. Therefore, a vertex that is the conclusion of an exponential link or the first premiss of a multiplicative one, cannot be r, nor any another in conclusion of $[\![t]\!]$, because, since $[\![t]\!]$ is closed, there are not. Secondly, an immediate verification of the definition of translation shows that the statement holds for $[\![t]\!]$, since: a $\multimap$-link and a ?-link
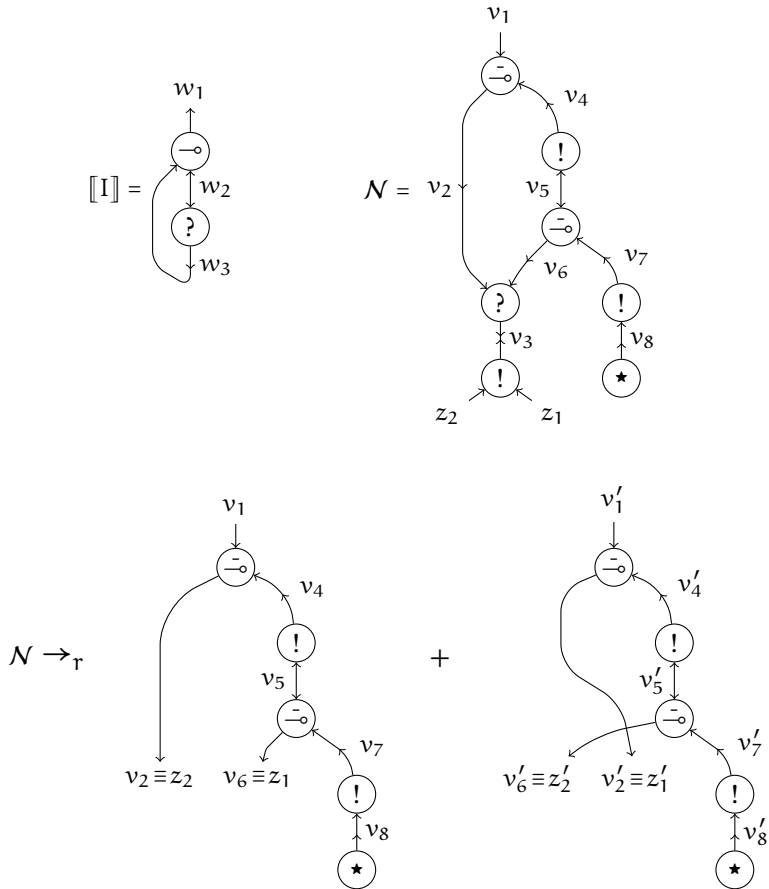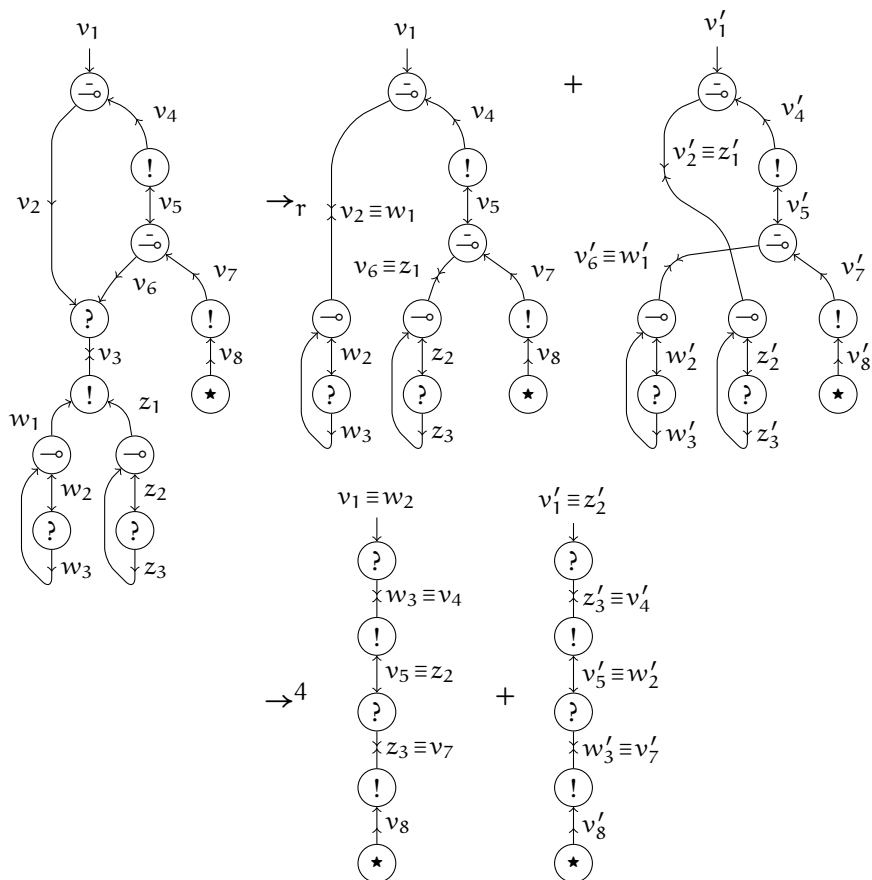
Figure 3.4 Example: nets and reduction.

Figure 3.5 Example: net reductions. Lowermost reduction is made of four steps, two on each addend.

are introduced when pre-translating an abstraction; a $\multimap$-link and a !-link are introduced when pre-translating an application and the polyterm; and in both cases the statement holds.

2. *Step.* Suppose $|\bar{\rho}| > 0$ and let $\rho = \rho'\bar{\rho}''$, for some reduction sequence $\bar{\rho}''$, and for some step ρ' of our interest acting on a redex R. For any $z \notin R$ the inductive hypothesis (IH) is trivially preserved. So let us focus on R and distinguish two cases depending its type.

   (a) $\mathcal{T}(R) = \multimap/\bar{\multimap}$. Let $R = \langle u, v \ (\multimap) \ w \rangle, \langle u', v' \ (\bar{\multimap}) \ w \rangle$. By IH, since v and v' are the first premisses of two multiplicative links, they must be conclusions of a ?- and a !-link, respectively. Now, in $\rho'(R)$ we have the vertex $v \equiv v'$, where the claim trivially holds since it is a cut.

   (b) $\mathcal{T}(R) = !/?$. Let $R = \langle v_1 \ldots v_n \ (?) \ w \rangle, \langle u_1 \ldots u_m \ (!) \ w \rangle$. If $n \neq m$ then $\rho'(R) = 0$ and there is nothing to prove, so let us assume otherwise. We show that no vertex z in R can satisfy any of the two statements whose co-implication is claimed.

      i. If $z = w$, then it is a cut. Moreover, z cannot be the premiss of any link, since in R there are already two links connected to z, (cf. Definition 2.3, item 1).

      ii. If $z = v_i$ (or $z = u_i$) for some $1 \leq i \leq n$, then we observe that it cannot be the conclusion of an exponential link l, nor the first premiss of a multiplicative link l', since it would violate the typing constraints. Indeed, it must be the case that $\mathcal{T}(z) = !A$, where either $A = \mathcal{T}(x)$ and x is any premiss of l, or $!A \multimap B = \mathcal{T}(y)$ and y is the conclusion of l'. This would require that $\mathcal{T}(w) = !!A$, which does not belong to the grammar of types (cf. Definition 2.2).

∎

The following lemma may be used to prove the completeness of a new correctness criteria with respect to the simple net we defined by translation.

Lemma 3.2 (Quasi-connectedness). *For any simple net N such that $N \not\twoheadrightarrow 0$, and any subnet $N' \subset N$, there exists no path connecting N' to the positive conclusion of N if and only if all the links of N' are (co-)weakenings.*

Proof. The "if" direction of the bi-implication in the thesis is obvious, from the fact that any subnet built from (co-)weakenings have no conclusions and must be disconnected from the main net's conclusion. For the "only if" part, we use Proposition 3.4, and we proceed by induction on the length of the (possibly empty) reduction sequence $\rho : [\![t]\!] \to^* N + \mathcal{N}$, for some term t and net $\mathcal{N}$.

1. $|\rho| = 0$. Trivial – simply observe the $[\![\cdot]\!]$ does not allow disconnections. □

2. $|\rho| > 0$ Suppose that $\rho = \rho'\rho''$, for some reduction sequence ρ'' where the thesis holds (IH) and some reduction step ρ' on a redex R.

(a) If ρ' is a linear implication step, then, because of Lemma 3.1, the only disconnection possibly created comes from a weakening connected to the firsts premiss of the $\multimap$-link $l \in R$. Let us suppose it is the case, because otherwise IH is trivially preserved. Now, by definition of nets, and in particular by Definition 2.3 and 2.2, the first premiss of the $\bar{\multimap}$-link must be a !-link l'. Then in $\rho'(\mathcal{N})$ the vertices l, l' form an exponential redex, which by hypothesis cannot make $\mathcal{N}$ implode in 0, therefore by Equation 3.15 l must be a co-weakening. Hence $\mathcal{N}' = \{l, l'\}$ and we conclude. ⧫

(b) Otherwise ρ' is a exponential implication step between two exponential links l, l' None of the premisses of l, l' can be the conclusion of a (co-)weakening, as stated by Lemma 3.1, therefore no new disconnection may be possibly created. Hence, the IH is trivially preserved and we conclude. ⧫

□

■

Proposition 3.5. *Any two resource redex in a net are parallel.*[1]

Proof. Notice that any redex in a resource net involve exactly two links, one positive and one negative, and also that the cut involves the principal ports of the two links. These two facts imply that two redexes cannot overlap. ■

Proposition 3.6. *Net reduction strongly normalises. Moreover, it simulates term reduction via translation: for any term* t, *if* $t \to t'$, *then* $[\![t]\!] \to_r^2 [\![t']\!]$ *with a* $(\multimap / \bar{\multimap})$-*step followed by a* $(!/?)$-*step.*

Proof. See Tranquilli [2011] on the promotion-enabled generalisation of RINs and RC. ■

3.4 Resource paths

We study the action of the reduction on paths, first by defining the appropriate notion of residual, i.e. the result of a reduction, and that of persistence, i.e. the capability of resisting any reduction, that is a sort of normalisability.

3.4.1 Statics

Recall the basic definitions needed to ensure visibility by a GoI construction of the computation in a path, that are straightness and maximality introduced in Definition 2.17. In addition to those, we define a notion of comprehensiveness enabling a path to see the entirety of the computation. This last notion is the only original difference with respect to the classic notion of path as formulated in Danos and Regnier [1995].

---

[1] For an introduction to basic concepts of rewriting theory in the more classic settings of terms, we address the reader, for instance, at Baader and Nipkow [1998].

Definition 3.9 (Comprehensiveness). A path $\pi \in N$ is *comprehensive* when it crosses all the premises of all the exponential links. The set of *comprehensive execution paths* in any simple net N being a (non-zero) addend of N is written $P_{EC}(N)$.

Example 3.4. Recall the nets discussed in Example 3.3 and observe again Figure 3.4. Consider the net $[\![I]\!]$ and the path $\phi = (w_1, w_2, w_3)$, which is straight and also maximal. Moreover, $P_E([\![I]\!]) = \{\phi, \phi^\dagger\}$. Notice that in N the paths (v_1, v_4) and (v_1, v_2, v_1) are not straight — the former is twisting, while the latter is bouncing. What about $P_E(N)$? If we start from v_1 we find two paths seeking for the head variable: $\pi_1 = (v_1, v_2, v_3, z_1)$ and $\pi_2 = (v_1, v_2, v_3, z_2)$. Both π_1, π_2 are straight and maximal, thus execution, but they are not comprehensive, since they do not cross v_4 nor v_7.

3.4.2 Dynamics

With the notions just introduced we now define the property of path persistence, that intuitively means "surviving cut-elimination". We first inspect the action of reduction on paths, and isolate the notion of residual of a given path with respect to a given reduction. We shall go by induction on its subpaths that pass through a given redex, called *crossings*, and say that the residual(s) of a path are obtained by the substitution of the crossing with its residual(s), that are the images of reduction as naturally induced by the graph-rewriting. The case of linear implication is straightforward, because the rewriting is local and we only have to ensure that a path does not partially belong to the redex. The case of exponentials is instead more delicate, because the rewriting is global: a simple net is rewritten as a sum of simple nets, hence a path may be duplicated in several addends or destroyed. Which addends contain the residual(s) of a given crossing of the redex? If the reduction of an exponential redex R rewrites the simple net to which it belongs as the empty sum, then also the residual of the crossing is 0. Otherwise, the reduction rewrites the net as a sum of simple nets where any crossing always has a residual (cf. Definition 3.8). But two crossings of R within a path may have residuals in different simple nets created by the reduction, each one obtained from a fixed permutation of premises to substitute R. Therefore, the residual of a path is a sum of paths, morally varying on the set of permutations they are allowed to follow.

Definition 3.10 (Linear implication residual). Let N be a net where $\chi \in P(N)$ is a crossing of a linear implication redex R (see Definition 2.18). Suppose R as in the top-most redex of Figure 3.3 or, equivalently, as in Equation 3.14, and let ρ be the reduction step on R. Then the residual of χ with respect to ρ, is defined as follows and by the rule $\rho(\chi^\dagger) = (\rho(\chi))^\dagger$.

$$\rho((v, w, u)) \stackrel{\mathrm{def}}{=} (v); \tag{3.16}$$

$$\rho((v', w, u')) \stackrel{\mathrm{def}}{=} (v'); \tag{3.17}$$

$$\rho((v, w, u')) \stackrel{\mathrm{def}}{=} 0; \tag{3.18}$$

$$\rho((v', w, u)) \stackrel{\mathrm{def}}{=} 0. \tag{3.19}$$

Let $\pi \in P(N)$ be long enough for R (see Definition 2.18), and let $\mathrm{RCF}_R(\pi) = \pi_0 :: \chi_1 ::$

$\pi_1 :: \ldots :: \chi_k :: \pi_k$ (*ibidem*). Then:

$$\rho(\pi) \overset{\text{def}}{=} \begin{cases} \pi_0 :: \rho(\chi_1) :: \pi_1 :: \ldots :: \rho(\chi_k) :: \pi_k & \text{if for any i, } \rho(\chi_i) \neq 0; \\ 0 & \text{otherwise.} \end{cases} \tag{3.20}$$

Definition 3.11 (Exponential residual)**.** Let $\mathcal{N}$ be a net where $\chi \in P(\mathcal{N})$ is a crossing of an exponential redex R. Suppose R as in the lowermost redex of Figure 3.3, or in textual notation given in Equation 3.15, and let ρ be the reduction step on R and $\sigma_n \in \mathfrak{S}_n$. The residual of χ with respect to ρ and σ_n is defined as follows and by rule $\rho(\chi^\dagger) = (\rho(\chi))^\dagger$. For any $1 \leq i \leq n$, and $1 \leq j \leq m$:

$$\rho^{\sigma_n}((v_i, w, u_j)) \overset{\text{def}}{=} \begin{cases} (v_i) & \text{if n = m, and } \sigma_n(i) = j, \\ 0 & \text{if n} \neq \text{m, or } \sigma_n(i) \neq j; \end{cases} \tag{3.21}$$

where in $V(\rho(R))$ we have that $v_i \equiv u_{\sigma_n(i)}$. Now, similarly to the last definition, given $\pi \in P(\mathcal{N})$ such that $\text{RCF}_R(\pi) = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_k :: \pi_k$, we define the residual of π with respect to σ_n:

$$\rho^{\sigma_n}(\pi) \overset{\text{def}}{=} \begin{cases} \pi_0 :: \rho^{\sigma_n}(\chi_1) :: \pi_1 :: \ldots :: \rho^{\sigma_n}(\chi_k) :: \pi_k & \begin{aligned}&\text{if n = m, and}\\&\text{for all } 1 \leq l \leq k, \rho^{\sigma_n}(\chi_l) \neq 0;\end{aligned} \\ 0 & \text{otherwise.} \end{cases} \tag{3.22}$$

Notice that $\rho^{\sigma_n}(\pi)$ may be 0 even if k = 0, i.e. when π does not cross R. Finally, we can define the residual of π as the sum of all the residuals, for any σ_n:

$$\rho(\pi) = \sum_{\sigma_n \in \mathfrak{S}_n} \rho^{\sigma_n}(\pi). \tag{3.23}$$

Definition 3.12 (Reduction and persistence)**.** For a given reduction step ρ, the *path reduction* function w.r.t. ρ is the function, written ρ, that maps a path π in $\mathcal{N}$ to the residual of π, i.e. a sum of paths in $\rho(\mathcal{N})$. If $\rho(\pi) \neq 0$, then π is *persistent w.r.t.* ρ. If, for every reduction sequence $\bar{\rho} = (\rho_1, \ldots, \rho_m)$, and for every $1 \leq i \leq m$, $\rho_1(\ldots \rho_{i-1}(\pi) \ldots)$ persists to ρ_i, then π is *persistent*.

Example 3.5. Recall the nets discussed in Example 3.3 and let ρ be the reduction illustrated in Figure 3.4, where we will denote by $\mathcal{N}_l$ and $\mathcal{N}_r$ the left and right addend of the reduct, respectively. Observe the execution paths $\pi_1 = (v_1, v_2, v_3, z_1)$ and $\pi_2 = (v_1, v_2, v_3, z_2)$, mentioned in Example 3.4. Both are persistent, since: $\text{NF}(\pi_1) = \pi_{1_r} = (v_1, v_2 \equiv z_1)$ and $\text{NF}(\pi_2) = \pi_{2_l} = (v_1, v_2 \equiv z_2)$. Remark also that π_1, π_2 cross the exponential redex differently, and they do not belong to the same addend of the reduct, for $\pi_{1_r} \in \mathcal{N}_r$, while $\pi_{2_l} \in \mathcal{N}_l$. Also, if we begin with the in conclusion z_1, which morally represents a free variable, we find straight paths that search for the argument to be substituted there, eventually reaching the out conclusion. Consider $\pi_3 = (z_1, v_3, v_2, v_4, v_5, v_6, v_3, z_1)$. It crosses the same exponential redex with two crossings, namely (z_1, v_3, v_2) and (v_6, v_3, z_1), that are incompatible since they belong to different permutations. Therefore π_3 not persistent, as it morally uses the same variable twice, for both the applications in $\mathcal{N}$.

Fact 3.1. *Straightness, maximality and comprehensiveness are preserved by reduction.*

3.4.3 Comprehensiveness and bijection

The persistence property of a path naturally implies that it travels only through vertices whose links morally contribute to the normalisation — a persistent path cannot cross (co-)weakenings[2]. Therefore, if we restrict ourselves to closed nets of constant type, we obtain a stronger property: a persistent execution path travel through all vertices of the net that are not conclusions of a 0-ary exponential links. Despite what the creation of sums may suggest at a first glance, RINs do not allow for duplication of paths. Intuitively, the set of persistent ones is split from the simple net containing the redex into the addends created by the reduction. We shall show indeed that path reduction for any step ρ induces a bijection between the persistent paths of a net N and those of $\rho(N)$. To prove these two facts, we first explicit in the next lemma an expected property: the alternation between multiplicative and exponential operators that lies in the grammar of types has a natural and graphical counterpart in links of nets.

Lemma 3.3. *In a closed net $N : \star$ any persistent execution path is comprehensive.*

Proof. We shall prove a stronger statement: given a persistent path $\pi \in P_E(N)$, a vertex $v \notin \pi$ if and only if there exists a (co-)weakening l such that $v \in C(l)$.

1. The "if" direction of the claim follows from a mere observation of the Definition 2.6 of execution paths. If π includes a conclusion of a (co-)weakening, then π is necessarily bouncing or non-maximal, in both cases contradicting the hypothesis that π is an execution path. □

2. In order to prove the "only if" part of the claim, let us first recall that, by Definition 3.4, for any simple net N' there exists a term such that N' either appears in its translation, or in some of its reducts. We now go by induction on a sequence $(\bar{\rho})^{-1}$ of expansions (or an anti-sequence of reductions) from $NF(N)$ back to $N = [\![t]\!]$, for some term t. If $NF(N) = 0$ there is nothing to prove, so we shall assume it to be non-zero.

   a. *Base.* Suppose $|\bar{\rho}| = 0$. Then $NF(N) = N = c\,[\![t]\!] = [\![c\,t]\!]$ for some $c \in \mathbb{N}$. Therefore $t = \star$, because the only closed term whose translation is normal with respect to net reduction is $\star$. Then $[\![t]\!] = \langle (\star)\,v \rangle$, and $P_E([\![t]\!]) = \{(v,v)\}$, hence the claim. ♦

   b. *Step.* Suppose $|\bar{\rho}| > 0$. Let $\rho : N \to N'$ and $N' \to^* NF(N)$. Also, let R be the redex eliminated by ρ, and $C[\;]$ its context. We then distinguish two sub-cases depending on the type of R.

      i. $\mathcal{T}(R) = {\multimap}/{\bar{\multimap}}$. Suppose N' to be an addend of N' containing the vertices $v \equiv v_1 \equiv v_2$ and $u \equiv u_1 \equiv u_2$, and assume N to be the addend of N containing u, v. Let the expansion step be the following, which introduces the distinct vertices v_1, v_2, u_1, u_2, w.

      $$N' = C[u_1 \equiv u_2, v_1 \equiv v_2] \quad \leftarrow \quad C[\langle u_1, v_1\,({\multimap})\,w \rangle, \langle u_2, v_2\,(\bar{\multimap})\,w \rangle] \tag{3.24}$$

We firstly assume that $\pi \in \mathcal{N}'$, because otherwise π is unaffected and the IH would be trivially preserved. Now v_1, v_2 are second premises of the cut links, therefore, as established by Lemma 3.1, they cannot be the conclusion of a (co-)weakening, and consequently neither v can. Hence, by IH, $v \in \pi$ and it is enough to observe, by Definition 3.10, that also $v_1, v_2 \in \rho^{-1}(\pi)$. Let's now discuss u. Observe first that, since u_1 and u_2 are first premises of multiplicative links, we know from Lemma 3.1 that they both must be conclusion of exponential links. Therefore, and by definition of reduction, u has to be an exponential cut. We distinguish two sub-cases.

A. If $u \notin \pi$, then by definition of path reduction, $u_1, u_2 \notin \rho^{-1}(\pi)$. Moreover, by IH we have that u the conclusion of a weakening or a co-weakening link.

   I. If u is the conclusion of a weakening, then observe that the co-contraction whose conclusion is u must have arity 0. Otherwise, the reduction of u would rewrite $\mathcal{N}'$ as 0, contradicting the persistence hypothesis for π.

   II. If instead u the conclusion of a co-weakening, then, dually with respect to the previous case, and for the same reductio ad absurdum, the contraction that has conclusion in u must have arity 0.

   Hence, in both cases u is a cut between 0-ary exponentials, which by definition of reduction implies that u_1, u_2 are respectively conclusions of a weakening and a co-weakening link. ▲

B. Otherwise, $u \in \pi$. Then again by inspection of Definition 3.10, we verify that $u_1, u_2 \in \rho^{-1}(\pi)$. ▲

◇

ii. $\mathcal{T}(R) = !/?$. If the expansion affects no addends, i.e. if the reduction rewrites as 0 a simple net in $\mathcal{N}'$, then π is unaffected, since we assumed it to be persistent, and IH is trivially preserved. Otherwise, the arity of the two exponential links are equal, so let ρ be as follows.

$$C\left[\sum_{\sigma_n \in \mathfrak{S}_n} v_1 \equiv u_{\sigma_n(1)}, \ldots, v_n \equiv u_{\sigma_n(n)}\right]$$
$$\leftarrow C\left[\langle v_1, \ldots, v_n \, (!) \, w\rangle, \langle u_1, \ldots, u_n \, (?) \, w\rangle\right]. \qquad (3.25)$$

A. If $n = 0$, and the reduct of the redex is empty, then trivially π cannot not cross it. Nor can cross the redex, since the redex is made by a wakening and a co-weakening link, and we supposed π maximal and persistent. ▲

B. If $n > 0$, then let $\sigma_n \in \mathfrak{S}_n$ and consider the vertex $v_i \equiv u_{\sigma_n(i)}$. Notice that it cannot be the conclusion of a 0-ary exponential link. By contraposition, suppose otherwise and notice that, by definition of typing, in the reducendum we would have $\mathcal{T}(v_i \equiv v_{\sigma_n(i)}) = !A$ for some type A. This would absurdly imply $\mathcal{T}(w) = !!A$, that is not a valid type. Not being a (co-)weakening conclusion, by IH we have that $v_i \equiv u_{\sigma_n(i)} \in \pi$. To conclude it is then sufficient to observe that by definition of reduction, either $(v_i, w, v_{\sigma_n(i)})$ or its reversal belong to $\rho^{-1}(\pi)$. ▲

Remark 3.4. The notion of path comprehensiveness appears to be similar the the *trip lengthiness* introduced by Girard [1989, §2.1, p. 30] to characterise proof-nets among the nets in the case of MLL. This should not surprise, since the structure of resource nets, in its essence, only adds non-deterministic superposition to the structure of the multiplicative fragment of linear logic. Furthermore, we conjecture path comprehensiveness can be easily adapted to obtain a concise and elegant criterion for the correctness of simple resource net.

Theorem 3.1 (Reduction is bijective on persistent paths). *For any closed net $N : \star$, every reduction step ρ induces a bijection between execution paths in N that are persistent to ρ and those in $\rho(N)$.*

Proof. Let $\pi \in P_E(N)$ be persistent to ρ, let R the redex of ρ, and suppose $\mathrm{RCF}_R(\pi) = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_k :: \pi_k$. There are two reduction rules possibly used by ρ.

1. $\mathcal{T}(R) = \multimap / \overset{\_}{\multimap}$. Because of the persistence of π to ρ, and by the definition given by Equation 3.20, we have $\chi_l \neq 0$, for all $0 \leq l \leq k$, and $\rho(\pi) = \pi_0 :: \rho(\chi_1) :: \pi_1 :: \ldots :: \rho(\chi_k) :: \pi_k$. Then, if ρ is as in Figure 3.3, the bijection is given as follows:

   (a) $\chi_l = (v, w, u)$ if and only if $\rho(\chi_l) = (v)$;
   (b) $\chi_l = (v', w, u')$ if and only if $\rho(\chi_l) = (v')$.

   Such a bijection holds between χ_l and $\rho(\chi_l)$, so we also have a bijection between π and $\rho(\pi)$. □

2. $\mathcal{T}(R) = !/?$. Suppose the redex R being as in Figure 3.3. Because of the persistence of π to ρ, and by the definition given by Equation 3.22 and 3.23, it must be the case that $n = m$ and that there exists a permutation $\sigma_n \in \mathfrak{S}_n$ such that for all $0 \leq l \leq k$, we have $\rho^{\sigma_n}(\chi_l) \neq 0$. Moreover, by Lemma 3.3, π is comprehensive, which in particular means that $v_i, u_j \in \pi$ for any $1 \leq i, j \leq n$. Hence, for any $1 \leq i \leq n$ there exist $0 \leq l \leq k$ and $0 \leq j \leq n$ such that χ_l is either (v_1, w, u_j) or its reversal. In both cases, by Equation 3.21, it must be the case that $\sigma_n(i) = j$ so that $\rho^{\sigma_n}(\chi_l) = (v_{\sigma_n(i)} \equiv u_j)$. This means that σ_n is unique, and for any other σ'_n we have $\rho^{\sigma'_n}(\pi) = 0$. We then obtained a one-to-one correspondence between π and $\rho(\pi)$. □

■

Theorem 3.2 (Paths and addends). *For any closed simple net $N : \star$, the number of execution persistent paths in N is equal to the number of addends of the normal form of N.*

Proof. Any addend in a normal net is the simple net made by $\star$, which evidently has only one execution path, the unitary one on the only vertex. The thesis therefore is a corollary of Theorem 3.1. ■

3.4.4 Confluence and persistence

Since reductions on both RC and on RINs enjoy local confluence, the property is easily verified also on path reduction. This implies that persistence of a path with respect to a normalisation sequence is a sufficient condition for its general persistence.

Lemma 3.4. *Let N be a resource net and let $\pi \in P_E(N)$. For any two reduction steps ρ, σ on N, there exist two sequences $\bar{\rho}, \bar{\sigma}$ such that $\bar{\rho}(\sigma(\pi)) = \bar{\sigma}(\rho(\pi))$.*

Proof. Let R, S be the respective redexes of ρ and σ, and let N, M be the simple nets to which R, S respectively belong.

1. If $N \neq M$, then let $N = M + N + M$. By Definition 3.8 of reduction, $\rho(N) = M + \rho(N) + M$ and $\sigma(N) = M + N + \sigma(M)$. Therefore $R \subset \sigma(N)$ and $S \subset \rho(N)$. Let $\bar{\rho}$ be the reduction step acting on R within $\sigma(N)$ and let $\bar{\sigma}$ be that on S within $\rho(N)$. We immediately obtain that $\bar{\rho}(\sigma(N)) = M + \bar{\rho}(N) + \sigma(M)$, and $\bar{\sigma}(\rho(N)) = M + \rho(N) + \bar{\sigma}(M)$, which are equal as claimed. □

2. If $N = M$, then consider the partitioning of π obtained by iteratively isolating: first the longest prefix π' which does not cross S, then the longest prefix π'' which does not cross R.

$$\pi = \pi'_1 :: \pi''_1 :: \ldots :: \pi'_h :: \pi''_h \tag{3.26}$$

By definition of reduction, R, S are disjoint, i.e. if $R \ni v \in S$ then $\mathrm{iface}(R) \ni v \in \mathrm{iface}(S)$. This means that π'_i, π''_i, for any $1 \leq i \leq h$, are long enough respectively for R, S. Therefore, we can highlight $\mathrm{RCF}_R(\pi'_i)$ and $\mathrm{RCF}_S(\pi''_i)$ within Equation 3.26 and obtain a combined redex crossing form of π for R, S:

$$\begin{aligned}
\mathrm{RCF}_{R,S}(\pi) = {}& \pi_{0,0} :: \\
& \chi_{1,1} :: \pi_{1,1} :: \ldots :: \chi_{1,r_1} :: \pi_{1,r_1} :: \\
& \psi_{1,1} :: \pi_{1,r_1+1} :: \ldots :: \psi_{1,s_1} :: \pi_{1,r_1+s_1} :: \\
& \vdots \\
& \chi_{h,1} :: \pi_{h,1} :: \ldots :: \chi_{h,r_h} :: \pi_{h,r_h} :: \\
& \psi_{h,1} :: \pi_{h,r_h+1} :: \ldots :: \psi_{h,s_h} :: \pi_{h,r_h+s_h}.
\end{aligned} \tag{3.27}$$

where $\chi_{i,j}$ (resp. $\psi_{i,j}$) is the j-th crossing of R (resp. of S) within π'_i (resp. π''_i). We distinguish three cases, depending on the kind of redexes.

(a) $\mathcal{T}(R) = \mathcal{T}(S) = \multimap/\bar{\multimap}$. Recall that R, S are disjoint, and observe that by Definition 3.8 of net reduction, both ρ and σ rewrite N as a simple net. Thus, $S \subset \rho(N)$ and $R \subset \sigma(N)$. This means that we can simply take $\bar{\rho} = \rho$ reducing R and $\bar{\sigma} = \sigma$ reducing S. We immediately verify the claim by

applying Definition 3.10 of multiplicative residual to Equation 3.27.

$$\bar{\sigma}(\rho(\pi)) = \bar{\rho}(\sigma(\pi))$$
$$= \pi_{0,0} ::$$
$$\rho(\chi_{1,1}) :: \pi_{1,1} :: \ldots :: \rho(\chi_{1,r_1}) :: \pi_{1,r_1} ::$$
$$\sigma(\psi_{1,1}) :: \pi_{1,r_1+1} :: \ldots :: \sigma(\psi_{1,s_1}) :: \pi_{1,r_1+s_1} ::$$
$$\vdots$$
$$\rho(\chi_{h,1}) :: \pi_{h,1} :: \ldots :: \rho(\chi_{h,r_h}) :: \pi_{h,r_h} ::$$
$$\sigma(\psi_{h,1}) :: \pi_{h,r_h+1} :: \ldots :: \sigma(\psi_{h,s_h}) :: \pi_{h,r_h+s_h}. \tag{3.28}$$

♦

(b) $\mathcal{T}(R) = \multimap/\bar{\multimap}$ and $\mathcal{T}(S) = !/?$, or $\mathcal{T}(R) = !/?$ and $\mathcal{T}(S) = \multimap/\bar{\multimap}$. We assume
the former, since the proof is identical once the roles of R and S are swapped.
Let n be the arity of the ?-link in R. This time ρ rewrites $\mathcal{N}$ as a simple net,
while σ rewrites it as a sum of n' simple nets (cf. Definition 3.8). Observe
again that $S \subset \rho(\mathcal{N})$, so take $\bar{\sigma} = \sigma$ to be the reduction step for such S.

i. If $n' = 0$, R is not present in $\sigma(\mathcal{N}) = 0$. So, let $\bar{\rho}$ be the empty reduction
sequence. We immediately verify the claim. ◇

ii. Otherwise $n' > 0$, which means that R is copied in the n' addends of
$\sigma(\mathcal{N}) = \sum_{\sigma_n \in \mathfrak{S}_n} \sigma^{\sigma_n} \mathcal{N}$. To close the reduction diagram, we then have to
reduce all the duplicates of R. So, let us denote the sum of reducts of R as
$\sum_{\sigma_n \in \mathfrak{S}_n} R_{\sigma_n}$, and consider the set, for any $\sigma_n \in \mathfrak{S}_n$, of the reduction step
ρ_{σ_n} which reduces R_{σ_n}. Let $\bar{\rho}$ be the sequence obtained by any ordering
on that set. We can now unfold Definition 3.10 and Definition 3.11 of
exponential residual onto Equation 3.27.

$$\bar{\sigma}(\rho(\pi)) = \sum_{\sigma_n \in \mathfrak{S}_n} \pi_{0,0} ::$$
$$\rho(\chi_{1,1}) :: \pi_{1,1} :: \ldots :: \rho(\chi_{1,r_1}) :: \pi_{1,r_1} ::$$
$$\sigma^{\sigma_n}(\psi_{1,1}) :: \pi_{1,r_1+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{1,s_1}) :: \pi_{1,r_1+s_1} ::$$
$$\vdots$$
$$\rho(\chi_{h,1}) :: \pi_{h,1} :: \ldots :: \rho(\chi_{h,r_h}) :: \pi_{h,r_h} ::$$
$$\sigma^{\sigma_n}(\psi_{h,1}) :: \pi_{h,r_h+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{h,s_h}) :: \pi_{h,r_h+s_h}.$$
$$\tag{3.29}$$

$$\bar{\rho}(\sigma(\pi)) = \sum_{\sigma_n \in \mathfrak{S}_n} \pi_{0,0} ::$$
$$\rho_{\sigma_n}(\chi_{1,1}) :: \pi_{1,1} :: \ldots :: \rho_{\sigma_n}(\chi_{1,r_1}) :: \pi_{1,r_1} ::$$
$$\sigma^{\sigma_n}(\psi_{1,1}) :: \pi_{1,r_1+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{1,s_1}) :: \pi_{1,r_1+s_1} ::$$
$$\vdots$$
$$\rho_{\sigma_n}(\chi_{h,1}) :: \pi_{h,1} :: \ldots :: \rho_{\sigma_n}(\chi_{h,r_h}) :: \pi_{h,r_h} ::$$
$$\sigma^{\sigma_n}(\psi_{h,1}) :: \pi_{h,r_h+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{h,s_h}) :: \pi_{h,r_h+s_h}.$$
$$\tag{3.30}$$

Comparing the last two equations, we verify immediately that we have
$\rho_{\sigma_m}(\chi_{i,j}) = \rho(\chi_{i,j})$, for any $\sigma_n \in \mathfrak{S}_n$. Ergo $\bar{\rho}(\sigma(\pi)) = \bar{\sigma}(\rho(\pi))$. ◇

♦

(c) $\mathcal{T}(R) = \mathcal{T}(S) = !/?$. Let n, m respectively be the arities of the ?-links in R, S, and let n', m' be the numbers of simple nets respectively generated by ρ, σ on $\mathcal{N}$. We separate three cases depending on the condition of nullity of n', m'.

    i. $n' = m' = 0$, which means that n (resp. m) is not equal to the arity of the !-link in R (resp. S). Then clearly $\rho(\pi) = \sigma(\pi) = 0$, so let $\bar\rho$ and $\bar\sigma$ both be the empty reduction sequence, and immediately obtain the claim: $\bar\rho(\sigma(\pi)) = \bar\sigma(\rho(\pi)) = 0$. $\Diamond$

    ii. Either n' or m' is null. Suppose the former (the other case is dual), i.e. $n' = 0$ and $m' > 0$. This last fact implies that $\rho(\pi) = 0$, so on the one hand, let $\bar\sigma$ be the empty reduction sequence. On the other hand notice that, as in case 2(b)ii, R is copied by σ into $\sum_{\sigma_n \in \mathfrak{S}_n} R_{\sigma_n}$. So, let $\bar\rho$ be an enumeration of the set including the reduction step ρ_{σ_n} reducing on $R_{\mathfrak{S}_n}$, for all $\sigma_n \in \mathfrak{S}_n$. We then obtain:

$$
\begin{aligned}
\bar\rho(\sigma(\pi)) = \sum_{\substack{\sigma_m \in \mathfrak{S}_m \\ \sigma_n \in \mathfrak{S}_n}} & \pi_{0,0} :: \\
& \rho_{\sigma_n}^{\sigma_m}(\chi_{1,1}) :: \pi_{1,1} :: \ldots :: \rho_{\sigma_n}^{\sigma_m}(\chi_{1,r_1}) :: \pi_{1,r_1} :: \\
& \sigma^{\sigma_n}(\psi_{1,1}) :: \pi_{1,r_1+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{1,s_1}) :: \pi_{1,r_1+s_1} :: \\
& \vdots \\
& \rho_{\sigma_n}^{\sigma_m}(\chi_{h,1}) :: \pi_{h,1} :: \ldots :: \rho_{\sigma_n}^{\sigma_m}(\chi_{h,r_h}) :: \pi_{h,r_h} :: \\
& \sigma^{\sigma_n}(\psi_{h,1}) :: \pi_{h,r_h+1} :: \ldots :: \sigma^{\sigma_n}(\psi_{h,s_h}) :: \pi_{h,r_h+s_h};
\end{aligned}
\tag{3.31}
$$

where for any $\sigma_n \in \mathfrak{S}_n$, and any $\sigma_m \in \mathfrak{S}_m$, we have $\rho_{\sigma_n}^{\sigma_m}(\chi_{i,j}) = 0$. Thus, $\bar\rho(\sigma(\pi)) = 0$. $\Diamond$

    iii. $n', m' > 0$. Let $\bar\rho$ be defined as in sub-sub-case 2(c)ii. Dually, since S is copied by ρ into $\sum_{\sigma_m \in \mathfrak{S}_m} S_{\sigma_m}$, define $\bar\sigma$ as an enumeration of the set of any reduction step σ_{σ_n} reducing on $S_{\mathfrak{S}_m}$, for all $\sigma_m \in \mathfrak{S}_m$. We obtain:

$$
\begin{aligned}
\bar\rho(\sigma(\pi)) = \sum_{\substack{\sigma_n \in \mathfrak{S}_n \\ \sigma_m \in \mathfrak{S}_m}} & \pi_{0,0} :: \\
& \rho^{\sigma_m}(\chi_{1,1}) :: \pi_{1,1} :: \ldots :: \rho^{\sigma_m}(\chi_{1,r_1}) :: \pi_{1,r_1} :: \\
& \sigma_{\sigma_m}^{\sigma_n}(\psi_{1,1}) :: \pi_{1,r_1+1} :: \ldots :: \sigma_{\sigma_m}^{\sigma_n}(\psi_{1,s_1}) :: \pi_{1,r_1+s_1} :: \\
& \vdots \\
& \rho^{\sigma_m}(\chi_{h,1}) :: \pi_{h,1} :: \ldots :: \rho^{\sigma_m}(\chi_{h,r_h}) :: \pi_{h,r_h} :: \\
& \sigma_{\sigma_m}^{\sigma_n}(\psi_{h,1}) :: \pi_{h,r_h+1} :: \ldots :: \sigma_{\sigma_m}^{\sigma_n}(\psi_{h,s_h}) :: \pi_{h,r_h+s_h}.
\end{aligned}
\tag{3.32}
$$

Comparing it with Equation 3.31, we observe that for any $\sigma_n \in \mathfrak{S}_n$ and any $\sigma_m \in \mathfrak{S}_m$, we clearly have $\rho^{\sigma_m}(\chi_{i,j}) = \rho_{\sigma_n}^{\sigma_m}(\chi_{i,j})$, and $\sigma^{\sigma_n}(\psi_{i',j'}) = \rho_{\sigma_m}^{\sigma_n}(\psi_{i',j'})$. Hence $\bar\rho(\sigma(\pi)) = \bar\sigma(\rho(\pi))$. $\Diamond$

                                                     $\blacklozenge$

                                                     $\square$

                                                     $\blacksquare$

Lemma 3.5. *Let $\mathcal{N}$ be a resource net and let $\pi \in P_E(\mathcal{N})$. If π is persistent w.r.t. a reduction sequence $\bar\rho$ such that $\bar\rho(\mathcal{N}) = NF(\mathcal{N})$, then π is persistent.*

Proof. Let the rank $\text{rk}(\mathbf{N})$ of a resource net $\mathbf{N}$ be the length of the longest reduction sequence on $\mathbf{N}$. Observe that the rank is always finite, since the reduction on RINs is strongly normalising and thanks to Kőnig lemma. So we can go by induction on $\text{rk}(\mathbf{N})$ and show that π is persistent to any normalisation sequence $\bar{\sigma}$.

1. *Base:* $\text{rk}(\mathbf{N}) = 0$. This implies that $|\bar{\rho}| = 0$, which means that $\text{NF}(\mathbf{N}) = \mathbf{N}$. Therefore, $\bar{\rho}$ is the only possible reduction sequence on $\mathbf{N}$. Hence, π is trivially persistent. □

2. *Step:* $\text{rk}(\mathbf{N}) > 0$. Since this implies that $|\bar{\rho}|, |\bar{\sigma}| > 0$, let $\bar{\rho} = \rho_1 \bar{\rho}_2$ and $\bar{\sigma} = \sigma_1 \bar{\sigma}_2$ for some reduction steps ρ_1, σ_1 and reduction sequences $\bar{\rho}_2, \bar{\sigma}_2$. We can then apply the local confluence property (Lemma 3.4) on ρ_1, σ_1 and obtain that there exist two sequences $\bar{\theta}, \bar{\phi}$ such that $\bar{\theta}(\rho_1(\pi)) = \bar{\phi}(\sigma_1(\pi))$.

   Now, since $\text{rk}(\rho_1(\mathbf{N})) < \text{rk}(\mathbf{N})$ and $\rho_1(\pi)$ is persistent w.r.t. the normalisation $\bar{\rho}_2$, per IH we have that $\rho_1(\pi)$ is persistent. This in particular implies that $\rho_1(\pi)$ persists to $\bar{\theta}$. Moreover, if $\bar{\mu}$ is a normalisation sequence on $\bar{\theta}(\rho_1(\pi))$, then $\rho_1(\pi)$ persists to $\bar{\theta}\bar{\mu}$ as well. Therefore, we can apply IH again, because $\text{rk}(\bar{\theta}(\rho_1(\mathbf{N}))) < \text{rk}(\rho_1(\mathbf{N})) < \text{rk}(\mathbf{N})$, and obtain that $\bar{\theta}(\rho_1(\pi))$ is persistent. But $\bar{\phi}(\sigma_1(\pi)) = \bar{\theta}(\rho_1(\pi)) \neq 0$, therefore this means that $\sigma_1(\pi)$ persists to $\bar{\phi}$ and to $\bar{\phi}\bar{\mu}$. Now $\text{rk}(\sigma_1(\mathbf{N})) < \text{rk}(\mathbf{N})$, so we can apply the IH one last time and conclude that $\sigma_1(\pi)$ is persistent. Hence, $\sigma_1(\pi)$ persists in particular to $\bar{\sigma}_2$, which means that π is persistent w.r.t. $\sigma_1\bar{\sigma}_2$, i.e. w.r.t. $\bar{\sigma}$. □

■

3.5 Resource execution

3.5.1 Dynamic algebra and execution

We are ready to formulate the GoI construction for RINs, adapting the formulation for the case of mMELL as most classically formulated by Regnier [1992], Danos and Regnier [1995], and presented here in Chapter 2. The main difference is that we characterise our resource exponentials, which have no promotion, as a sort of superposition of n-ary multiplicatives. We define a weight assignment for paths, so that the execution of a net is the sum of the weights of any execution path within it, and we formulate a monoidal structure $\mathsf{r}\mathfrak{L}^*$ of weights representing the computation. A crossing of an exponential link is morally weighed with a sum of indexed symbols, where the index varies in the set of permutations of the link's premises, and exponential weights interacts exactly as multiplicatives ones, i.e. by nullification or neutralisation.

Definition 3.13 (Dynamic algebra)**.** The $\mathsf{r}\mathfrak{L}^*$ monoid, essentially a restriction of $\mathfrak{L}^*$ defined in Definition 2.21, is defined over terminal symbols in $\{0, 1, \mathsf{p}, \mathsf{q}, \mathsf{l}, \mathsf{r}, \star\}$. A word of its alphabet, called *weight*, is generated by a binary *concatenation* operator with infix implicit notation and a unary *adjoint* operator $(\cdot)^*$. The concatenation operator and the set of symbols has the structure of a monoid, whose identity element is 1,

equipped with an additional absorbing element 0.

$$a(bc) = (ab)c; \tag{3.33}$$

$$a1 = 1a = a; \tag{3.34}$$

$$a0 = 0a = 0. \tag{3.35}$$

Moreover, similarly to an adjoint, the inversion operator is involutive and distributes over concatenation by reversing left and right operands.

$$(a^*)^* = a; \tag{3.36}$$

$$(ab)^* = b^*a^*. \tag{3.37}$$

Finally, it satisfies the neutralisation and annihilation equations, which represent the core of the computation.

$$pp^* = qq^* = ll^* = rr^* = 1; \tag{3.38}$$

$$qp^* = pq^* = rl^* = lr^* = 0. \tag{3.39}$$

We denote $l\underbrace{r\ldots r}_{n}$ as e_n, so that for any $n \neq m \in \mathbb{N}$ we have $e_n e_n^* = 1$ and $e_n e_m^* = 0$.

Definition 3.14 (Path weighting). The *permuted base weighting* is a map w that associate a weight of $r\mathcal{L}^*$ to an atomic straight path $\pi = (u, v) \in P(\mathcal{N})$ and a resource permutation σ. Straightness of π implies that it goes either: (i) from a conclusion to a conclusion of a $\star$-link; (ii) from a premiss to a conclusion of a binary link; (iii) vice versa, from a conclusion vertex to a premiss of a binary link. The permuted base weighting is defined as follows, where the first clause covers (i), the clauses from the second to the fifth cover (ii), and the last clause covers (iii).

$$w^\sigma((u,v)) = \begin{cases} \star & \text{if there is } \langle (\star)\, u \rangle \text{ and } u = v; \\ p & \text{if there is } \langle u, w\ (\multimap)\ v \rangle \text{ or } \langle u, w\ (\overset{\sim}{\multimap})\ v \rangle; \\ q & \text{if there is } \langle w, u\ (\multimap)\ v \rangle \text{ or } \langle w, u\ (\overset{\sim}{\multimap})\ v \rangle; \\ e_i & \text{if there is } \langle u_1, \ldots, u_i, \ldots, u_n\ (?)\ v \rangle \text{ and } u_i = u; \\ e_{\sigma_r(i)} & \text{if there is } r = \langle u_1, \ldots, u_i, \ldots, u_n\ (!)\ v \rangle \text{ and } u_i = u; \\ \left(w^\sigma((v,u))\right)^* & \text{otherwise.} \end{cases} \tag{3.40}$$

The *permuted weighting*, denoted again as $w^\sigma(\pi)$, is the lifting of the permuted base weighting to generic straight paths, and the *path weighting*, written $w(\pi)$, is the sum of all the permuted weights of a path, for any resource permutation.

$$w^\sigma((v)) = 1; \tag{3.41}$$

$$w^\sigma((u,v) :: \pi) = w^\sigma((u,v))\, w^\sigma(\pi); \tag{3.42}$$

$$w(0) = 0; \tag{3.43}$$

$$w(\pi) = \sum_{\sigma \in \mathfrak{S}_\mathcal{N}} w^\sigma(\pi). \tag{3.44}$$

The sum forms a free commutative monoid on the $r\mathcal{L}^*$ structure, and its identity is 0. Formally, we have that for any $a, b, c \in (r\mathcal{L}^*, +)$:

$$(a+b)+c = a+(b+c); \tag{3.45}$$

$$a+b = b+a; \tag{3.46}$$

$$a+0 = a. \tag{3.47}$$

Recall that $\mathfrak{S}_N$ cannot be empty — it contains at least the empty permutation, hence the weighting cannot be the empty sum.

Definition 3.15 (Regularity and execution)**.** A path π is *regular* if $\mathfrak{w}(\pi) \neq 0$. The *execution* of a net $\mathcal{N}$, is the sum of the weights of all its comprehensive execution paths, i.e.

$$\mathfrak{E}\mathfrak{x}(\mathcal{N}) \stackrel{\text{def}}{=} \sum_{\pi \in \mathcal{P}_{\text{EC}}(\mathcal{N})} \mathfrak{w}(\pi). \tag{3.48}$$

Example 3.6. Consider again the closed simple net M, whose reduction has been discussed in Example 3.3, and that is depicted in the leftmost extremity of Figure 3.5. To have an idea of the execution of M and of the behaviour of the algebraic structure, let us consider an execution comprehensive path, one of the persistent two, and compute its weight. Given that the path is palindromic, i.e. has the form $\pi :: \pi^\dagger$, we will consider only its first half, that goes from the root of the term to the constant. Moreover, we will break lines when a path inverts its polarity direction, i.e. if it walks from in to out or vice versa.

$$(v_1, v_2, v_3, w_1, w_2, w_3, \qquad\qquad \sum_{\sigma \in S_2} q\, e_1\, e^*_{\sigma(2)}\, q^*\, e_1 \cdot$$
$$w_1, v_3, v_2, \qquad\qquad\qquad p\, e_{\sigma(2)}\, e^*_1 \cdot$$
$$v_4, v_5, v_6, v_3, z_1, z_2, z_3, \qquad\qquad p^*\, e^*_1\, q\, e_2\, e^*_{\sigma(1)}\, q^*\, e_1 \cdot$$
$$z_1, v_3, v_6, \qquad\qquad\qquad p\, e_{\sigma(1)}\, e^*_2 \cdot$$
$$v_7, v_8); \qquad\qquad\qquad p^*\, e^*_1\, \star.$$

On the path: reduce it using the exponential rule. On the weight: apply (3.39) and then (3.35) on the addend s.t. $\sigma = (2, 1)$, apply (3.38) and then (3.34) on the one s.t. $\sigma = (1, 2)$.

$$\rightarrow (v_1, v_2 \equiv w_1, w_2, w_3, \qquad\qquad =_{\text{r}\mathcal{L}^*} q\, q^*\, e_1 \cdot$$
$$w_1 \equiv v_2, \qquad\qquad\qquad p \cdot$$
$$v_4, v_5, v_6 \equiv z_1, z_2, z_3, \qquad\qquad p^*\, e^*_1\, q\, q^*\, e_1 \cdot$$
$$z_1 \equiv v_6, \qquad\qquad\qquad p \cdot$$
$$v_7, v_8) + \qquad\qquad\qquad p^*\, e^*_1\, \star +$$
$$0; \qquad\qquad\qquad 0.$$

Forget zeros on both side. On the path: reduce it using the leftmost linear implication rule. On the weight: apply (3.38) and then (3.34).

$$\rightarrow (v_1 \equiv w_2, w_3 \equiv v_4, v_5, v_6 \equiv z_1, z_2, z_3, \qquad\qquad =_{\text{r}\mathcal{L}^*} e_1\, e^*_1\, q\, q^*\, e_1 \cdot$$
$$z_1 \equiv v_6, \qquad\qquad\qquad p \cdot$$
$$v_7, v_8); \qquad\qquad\qquad p^*\, e^*_1\, \star.$$

On the path: reduce it using the linear implication rule. On the weight: apply (3.38) and then (3.34).

$$\rightarrow (v_1 \equiv w_2, w_3 \equiv v_4, v_5 \equiv z_2, z_3 \equiv v_7, v_8); \qquad\qquad =_{\text{r}\mathcal{L}^*} e_1\, e^*_1\, e_1\, e^*_1\, \star.$$

On the path: reduce it twice using exponential rules. On the weight: apply (3.38) and then (3.34), and repeat.

$$\rightarrow (v_1 \equiv w_2 \equiv v_5 \equiv z_2, z_3 \equiv v_7, v_8) \qquad\qquad =_{\text{r}\mathcal{L}^*} e_1\, e^*_1\, \star$$
$$\rightarrow (v_1 \equiv w_2 \equiv v_5 \equiv z_2 \equiv v_8); \qquad\qquad =_{\text{r}\mathcal{L}^*} \star.$$

Therefore the persistent path turns out to be regular. Even more, along the reduction we managed to apply, for each step, some $r\mathcal{L}^*$ equations so that the weight of every reduct is equal to the manipulated weight. The next two theorems shall generalise these two facts.

Remark 3.5 ($\mathcal{L}^*$ versus $r\mathcal{L}^*$). With respect to the $\mathcal{L}^*$-monoid of mMELL that has been presented in Section 2.5 (cf. Definition 2.21), $r\mathcal{L}^*$ has no exponential modality. In particular $r\mathcal{L}^*$ can be obtained by removing from $\mathcal{L}^*$: the exponential operator ! and the commutation and lifting equations. Hence, $r\mathcal{L}^* \subset \mathcal{L}^*$. This is the reason for which we choose to define the weighting function by externally factorising the complexity arising from the non-deterministic behaviour of RINs. Albeit the weight assignment looses compositionality, i.e. compatibility with respect to path concatenation, since it is not always the case that $w\,(\pi :: \pi') = w\,(\pi)\,w\,(\pi')$, the formulation is remarkably simple and evokes projections on orthogonal bases.

3.5.2 Invariance and regularity

We now show that the $r\mathcal{L}^*$ monoid introduced above accurately computes path reduction. We prove the equivalence between regularity and persistence, and show that execution is invariant by reduction. Not only the construction is a suitable semantic for ground typed RINs, but also possesses quantitative awareness, since, for any term, the number of execution paths that are regular is equal to the number of addends in its normal form.

Lemma 3.6. *For any closed net $\mathcal{N} : \star$, any reduction step ρ, and any path $\pi \in P_{EC}(\mathcal{N})$:*

$$w\,(\pi) =_{r\mathcal{L}^*} w\,(\rho(\pi)). \tag{3.49}$$

Proof. Let $\pi \in P_E(N)$ with N addend of $\mathcal{N}$ containing the redex R. Recall that π has to be long enough for ρ, for it is maximal. Suppose the RCF of π w.r.t. R is $\pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_k :: \pi_k$. We proceed by a case analysis of the kind of reduction rule.

1. $\mathcal{T}(R) = \multimap/\overline{\multimap}$. Let R be as in the leftmost redex of Figure 3.3. We distinguish two sub-cases, depending on the nullity of $\rho(\pi)$.

   (a) Suppose $\rho(\pi) = 0$. By definition of weighting (Equation 3.43) we have $w\,(\rho(\pi)) = 0$. Moreover, by Definition 3.10 and in particular (3.20), there must exists $0 \leq l \leq k$, such that $\rho(\chi_l) = 0$. Hence, it must be the case that χ_l is either as in (3.18), or as in (3.19).

      i. Suppose $\chi_l = (v, w, u')$. Then $\rho(\chi_l) = 0$, and $w\,(\chi_l) = pq^* =_{r\mathcal{L}^*} 0$.
      ii. Suppose $\chi_l = (v', w, u)$. Then $\rho(\chi_l) = 0$, and $w\,(\chi_l) = qp^* =_{r\mathcal{L}^*} 0$.

      Again by definition of weighting (Equation 3.42, 3.44), and applying (3.35), we conclude $w\,(\pi) =_{r\mathcal{L}^*} 0 = w\,(\rho(\pi))$. $\blacklozenge$

   (b) Suppose $\rho(\pi) \neq 0$. Then, again by definition of multiplicative residual, and in particular by Equation 3.20, we know $\rho(\chi_l) \neq 0$ for any $0 \leq l \leq k$. Hence, it must be the case that χ_l is either as in Equation 3.16, or as in 3.17.

      i. If $\chi_l = (v, w, u)$, then $\rho(\chi_l) = (v' \equiv u')$, and $w\,(\chi_l) = pp^* =_{r\mathcal{L}^*} 1$.

ii. If otherwise $\chi_l = (v', w, u')$, then $\rho(\chi_l) = (v' \equiv u')$, and $\boldsymbol{w}(\chi_l) = qq^* =_{r\mathcal{L}^*} 1$.

Now, applying this fact on the definition given by Equation 3.20, and using Equation 3.34 we conclude.

$$\boldsymbol{w}(\pi) = \sum_{\sigma \in \mathfrak{S}_N} \boldsymbol{w}^\sigma(\pi_0)\, \boldsymbol{w}^\sigma(\chi_1)\, \boldsymbol{w}^\sigma(\pi_1)\, \ldots\, \boldsymbol{w}^\sigma(\chi_k)\, \boldsymbol{w}^\sigma(\pi_k)$$

$$= \sum_{\sigma \in \mathfrak{S}_N} \boldsymbol{w}^\sigma(\pi_0)\, \boldsymbol{w}^\sigma(\pi_1)\, \ldots\, \boldsymbol{w}^\sigma(\pi_k)$$

$$= \boldsymbol{w}(\rho(\pi)).$$

♦

□

2. $\mathcal{T}(R) = !/?$. Let R be as in the rightmost redex of Figure 3.3, and let r be the !-link involved in it. We distinguish again two sub-cases, depending on the nullity of $\rho(\pi)$.

(a) Suppose $\rho(\pi) = 0$. Then by Definition 3.11, in particular Equation 3.22, there are only two possible causes.

i. Arity mismatch, i.e. when $n \neq m$, where n, m are the arities of the two links. Because of the hypothesis of comprehensiveness of π, it must be the case that $k \geq \max(n, m)$. Then, whatever permutation $\sigma_n \in \mathfrak{S}_n$ we choose for the premises of the !-link in R, there always exists a crossing χ_l, for some $0 \leq l \leq k$, such that $\chi_l = (u_{\sigma_n(i)}, w, v_j)$ and $\sigma_n(i) \neq j$.

ii. Permutation incoherence, i.e. when $n = m$, but for any $\sigma_n \in \mathfrak{S}_n$ there exists a crossing $\chi_l \subseteq \pi$ such that $\sigma_n(i) \neq j$. This morally happens when π tries to use more than once a resource, travelling from the same premiss of the ?-link to two different premises of the corresponding !-link.

Thus, in both cases there is a "wrong" crossing $\chi_l \subset \pi$ such that, for any resource permutation $\sigma \in \mathfrak{S}_N$ (recall Definition 3.6), we have $\boldsymbol{w}^\sigma(\chi_l) = e_{\sigma(r)(i)}e_j^*$ where $\sigma(r)(i) \neq j$. Hence, by (3.39), $\boldsymbol{w}^\sigma(\chi_l) = 0$. By definition of weighting (Equation 3.42, 3.44) and applying (3.35), we have $\boldsymbol{w}(\pi) =_{r\mathcal{L}^*} 0$. But by the same definition we also have $\boldsymbol{w}(\rho(\pi)) = 0$, so we conclude.

(b) Suppose $\rho(\pi) \neq 0$. Again by definition of path reduction, it must be the case that $n = m$, and that there exists $\sigma'_n \in \mathfrak{S}_n$ such that for all $0 \leq l \leq k$ we have $\rho^{\sigma'_n}(\chi_l) \neq 0$. In particular, let χ_l be as in (3.21), and observe it must be also the case that $\sigma'_n(i) = j$, which allows $\rho^{\sigma'_n}(v_i, w, u_j) = (v_{\sigma_n(i)} \equiv u_j)$. Moreover, by the comprehensiveness hypothesis for π, σ'_n has to be unique, so that for any other σ''_n, we have $\rho^{\sigma''_n}(\pi) = 0$. So, according to this, we split resource permutations $\mathfrak{S}_N$ into $\mathfrak{S}'_N \cup \mathfrak{S}''_N$, where the former is the set of any σ' such that $\sigma'(r) = \sigma'_n$, while, symmetrically, the latter contains any σ'' for

which $\sigma''(r) \neq \sigma'_n$. Hence, by Definition 3.14 we obtain:

$$
\mathfrak{w}(\pi) = \sum_{\sigma' \in \mathfrak{S}'_{\mathcal{N}}} \mathfrak{w}^{\sigma'}(\pi) + \sum_{\sigma'' \in \mathfrak{S}''_{\mathcal{N}}} \mathfrak{w}^{\sigma''}(\pi)
$$

$$
= \sum_{\sigma' \in \mathfrak{S}'_{\mathcal{N}}} \mathfrak{w}^{\sigma'}(\pi_0)\, \mathfrak{w}^{\sigma'}(\chi_1)\, \mathfrak{w}^{\sigma'}(\pi_1)\, \dots\, \mathfrak{w}^{\sigma'}(\chi_k)\, \mathfrak{w}^{\sigma'}(\pi_k) +
$$

$$
\sum_{\sigma'' \in \mathfrak{S}''_{\mathcal{N}}} \mathfrak{w}^{\sigma''}(\pi_0)\, \mathfrak{w}^{\sigma''}(\chi_1)\, \mathfrak{w}^{\sigma''}(\pi_1)\, \dots\, \mathfrak{w}^{\sigma''}(\chi_k)\, \mathfrak{w}^{\sigma''}(\pi_k).
$$

In the leftmost series, by (3.38), $\mathfrak{w}^{\sigma'}(\chi_l) = e_{\sigma'(r)(i)} e_j^* = 1$. While in the rightmost, by (3.39), $\mathfrak{w}^{\sigma''}(\chi_l) = e_{\sigma''(r)(i)} e_j^* = 0$, so it neutralises to 0. Therefore we concluded:

$$
=_{r \mathcal{L}^*} \sum_{\sigma' \in \mathfrak{S}'_{\mathcal{N}}} \mathfrak{w}^{\sigma'}(\pi_0)\, \mathfrak{w}^{\sigma'}(\pi_1)\, \dots\, \mathfrak{w}^{\sigma'}(\pi_k)
$$

$$
= \mathfrak{w}(\rho(\pi)).
$$

□

■

Theorem 3.3. *For any closed net* $\mathcal{N} : \star$, *any reduction sequence* $\bar{\rho}$, *and any path* $\pi \in \mathcal{P}_{EC}(\mathcal{N})$:

$$
\mathfrak{w}(\pi) =_{r \mathcal{L}^*} \mathfrak{w}(\bar{\rho}(\pi)). \tag{3.50}
$$

Proof. A straightforward induction on the length n of $\bar{\rho}$.

1. *Base.* Suppose $n = 0$. Trivially, $\bar{\rho}(\pi) = \pi$, so $\mathfrak{w}(\bar{\rho}(\pi)) = \mathfrak{w}(\pi)$.

2. *Step.* Suppose $n > 0$. Let $\bar{\rho} = \rho'\bar{\rho}''$, with ρ' a single step, $\bar{\rho}''$ a sequence of reductions. By Definition 3.12 of path reduction, $\rho'(\bar{\rho}''(\pi)) = \bar{\rho}(\pi)$. By previous Lemma 3.6, $\mathfrak{w}(\bar{\rho}''(\pi)) =_{r \mathcal{L}^*} \mathfrak{w}(\rho'(\bar{\rho}''(\pi)))$. But, by inductive hypothesis we have that $\mathfrak{w}(\pi) =_{r \mathcal{L}^*} \mathfrak{w}(\bar{\rho}''(\pi))$ so we conclude.

■

Theorem 3.4. *For any closed net* $\mathcal{N} : \star$, *a path* $\pi \in \mathcal{P}_{EC}(\mathcal{N})$ *is persistent if and only if* π *is regular.*

Proof. Immediate from Theorem 3.3. ■

Theorem 3.5. *For any closed net* $\mathcal{N} : \star$ *and any reduction sequence* $\bar{\rho}$,

$$
\mathcal{E}x(\mathcal{N}) =_{r \mathcal{L}^*} \mathcal{E}x(\bar{\rho}(\mathcal{N})). \tag{3.51}
$$

Proof. Consider first the execution of $\mathcal{N}$ (Definition 3.14 and Equation 3.48), and notice that we can separate the weights of paths that are persistent to $\bar{\rho}$ from those of paths that are not.

$$
\mathcal{E}x(\mathcal{N}) = \sum_{\pi \in \mathcal{P}_{EC}(\mathcal{N})} \mathfrak{w}(\pi) = \underset{\substack{\pi' \in \mathcal{P}_{EC}(\mathcal{N}) \\ \text{persistent to } \bar{\rho}}}{\sum} \mathfrak{w}(\pi') + \underset{\substack{\pi'' \in \mathcal{P}_{EC}(\mathcal{N}) \\ \text{not persistent to } \bar{\rho}}}{\sum} \mathfrak{w}(\pi''). \tag{3.52}
$$

But, by hypothesis of non-persistence w.r.t. $\bar{\rho}$ we have $\bar{\rho}(\pi) = 0$, which implies by Theorem 3.3 that $\mathbf{w}(\pi'') = 0$. Hence, the rightmost series is 0, and we obtain that:

$$= \sum_{\substack{\pi' \in \mathsf{P}_{\mathrm{EC}}(\mathcal{N}) \\ \text{persistent to } \bar{\rho}}} \mathbf{w}(\pi') . \tag{3.53}$$

Consider now the execution of $\bar{\rho}(\mathcal{N})$, which by definition is:

$$\mathbf{Ex}(\bar{\rho}(\mathcal{N})) = \sum_{\pi''' \in \mathsf{P}_{\mathrm{EC}}(\bar{\rho}(\mathcal{N}))} \mathbf{w}(\pi''') . \tag{3.54}$$

Observe that, by a straightforward induction on the length of $\bar{\rho}$, we can generalise Theorem 3.1 to obtain the bijection induced by $\bar{\rho}$ between the paths in $\mathsf{P}_{\mathrm{EC}}(\mathcal{N})$ persist to $\bar{\rho}$ and those in $\mathsf{P}_{\mathrm{EC}}(\bar{\rho}(\mathcal{N}))$. So let $\pi' \in \mathsf{P}_{\mathrm{EC}}(\mathcal{N})$ persistent to $\bar{\rho}$, and $\pi''' \in \mathsf{P}_{\mathrm{EC}}(\bar{\rho}(\mathcal{N}))$, such that $\bar{\rho}(\pi') = \pi'''$. But by Theorem 3.3 we have $\mathbf{w}(\pi') = \mathbf{w}(\bar{\rho}(\pi'))$, hence we conclude.

$$= \sum_{\substack{\pi' \in \mathsf{P}_{\mathrm{EC}}(\mathcal{N}) \\ \text{persistent to } \bar{\rho}}} \mathbf{w}(\pi') = \mathbf{Ex}(\mathcal{N}) . \tag{3.55}$$

■

Corollary 3.1. *For any term $[\![t]\!] : \star$, the regular paths in $[\![t]\!]$ are as many as the (non-zero) addends in* $\mathsf{NF}(t)$.

Proof. By definition of the calculus and of its nets syntax, $\mathsf{NF}(t) = n\star$, for some natural number n. Clearly, $\mathsf{P}_{\mathrm{EC}}([\![\star]\!])$ contains a unique path, made by the unique vertex of $[\![\star]\!]$. Then $|\mathsf{P}_{\mathrm{EC}}([\![\mathsf{NF}(t)]\!])| = n$. But from last Theorem 3.5, $\mathbf{Ex}([\![t]\!]) = \mathbf{Ex}([\![\mathsf{NF}(t)]\!])$, therefore the claim. ■

3.6 Discussion

3.6.1 Related works

A GoI construction for differential interaction nets (DINs) [Ehrhard and Regnier, 2006a] has already been formulated by De Falco [2008]. Besides the similarities in the technical setting of DINs, the geometry of resource interaction turns out to be simpler and more effective, mainly thanks to: (i) the restriction to closed and ground-typed resource nets, (ii) the associative syntax we adopted for exponential links, and (iii) the stronger notion of path we use. The first simplifies the shape of persistent paths, because it implies that they are palindromes — they first start from the root of the net, then travel until to a link representing the constant term $\star$, and finally return to the root — and unique in every normal net/term. The second simplifies the management of the exponential links, because it ensures associativity and delimits their dynamics in only one pair of links, whilst in De Falco's work this property was completely lost and the system more verbose. In the latter paper, the author uses binary exponential links and introduces a syntactical embedding of the sum in nets by mean of binary links of named sums, and then recover associativity with an equivalence on nets. Compared to ours, De Falco's choice results in a drastically more complex GoI construction, even though he

suggests possible extensions of his approach with promotion (corresponding to the full differential λ-calculus) or even additives. The third ingredient allows us to consider full reduction, i.e. including the annihilating rule, whilst De Falco studied a "weak" variant, where that kind of redexes are frozen, and the GoI only characterises the corresponding notion of "weak-persistence". Indeed, we restrict to paths that cross every exponential in the net and prove this assumption not being a limitation, since it is always true, in case of persistence. Thus whenever $t \to 0$ a path necessarily crosses the annihilating redex, and the dynamic algebra is able to detect it.

3.6.2 Open questions

3.6.2.1 Higher expressivity

The geometry of resource interacton here presented is formulated on a restricted calculus: it lacks not only ordinary constants, such as booleans or natural numbers, but also, and more importantly, it lacks the power of full recursion, i.e. of Turing completeness. Can we strengthen the expressivity of the calculi here considered beyond the minimalist formulation to get closer to real programming-language class?

If one wants to preserve the typed setting, the obvious direction is the inclusion of fixed-point combinators, to define a PCF-like variant of the resource calculus (RC), where the restriction to ground types remains innocuous.

3.6.2.2 Geometry of differential interaction

Is it possible to define a more general GoI-based model for the full differential λ-calculus, and for the full differential linear logic, where both superposition and non-linearity are present?

In that case, the shape of persistent crossings in an exponential redex does not necessarily respect the definition we gave here by means of fixed permutations, because different copies of a box containing a redex may need different resource assignments. Moreover, in order to obtain a degree of compositionality higher than our GoRI, one should most probably sacrifice the beauty of the invariance under reduction, and make do with a result of equivalence between regularity and persistence. Nonetheless, the results about the dynamics of Taylor expansion for paths that are presented in Chapter 4 provide a promising starting point for these investigations.

Chapter 4

Taylor-Ehrhard-Regnier Expansion and Geometry of Interaction

Contents

4.1   Introduction

4.1.1   Expansion and paths computation

The linear nature of resource calculus enables it to be considered as a suitable target language for the linearisation of ordinary λ-calculus. The graphical syntax of resource interaction nets enhances such linearisation with a tight correspondence at the level of Linear Logic types: boxes' content is duplicated with any arbitrary cardinality (thus also erased), promotions becomes co-contractions, boxes themselves are forgotten, and linear implications just preserved. A quick look at Figure 4.1 provides the visual intuition. Each boxing-depth-recursive choice of those cardinalities is a simple resource net, while the (possibly infinite) series of all of them is a resource net, called the Taylor-Ehrhard-Regnier expansion of the original proof-net. Probably inspired by Girard's notion of approximants of the exponential modality [1987], the expansion was originally formulated by Ehrhard and Regnier [2008] as a map from λ-terms to differential λ-terms, and then simplified to be targeted into its linear fragment, the resource calculus, expansion was also refined in a typed net-like counterpart, employing not only the call-by-name translation that we consider here [Mazza and Pagani, 2007, Pagani and Tasson, 2009], but also the call-by-value discipline [Carraro and Guerrieri, 2014]. Given that resource nets strongly normalise, the expansion of a term can be interpreted as the series of its finite approximations, hence the link with Taylor series and the notion of differentiation.

Paths in λ-terms, or proof-nets, already own a sort of linear nature, since they enter in a box from its principal door every time its content is needed by some external piece of computation. In other, more operational, words, a path flows down an argument from its root every time it crosses an occurrence of a variable in need of being substituted.

How is paths dynamics related before and after the expansion? Can we expand β-reduction into an infinite parallel step of resource reductions? Can we use the linear and non-deterministic GoI for the resource calculus presented in last chapter to characterise persistent paths in ordinary λ-terms, via expansion? This chapter addresses these questions and recounts the interplay between the two aforementioned semantic approaches, exploring both directions of their mutual influence.

4.1.2 Outline

We first introduce a qualitative variant of Taylor-Ehrhard-Regnier expansion, which maps a proof-net, or a path within it, in an infinite sum of simple RINs, or of set of paths within them (Section 4.2, 4.2.3). Since here this sum is idempotent, the expansion is essentially an infinite set. Also, we define a notion of expanded cut-elimination, a kind of infinite parallel reduction for RINs which reduces together all redexes that are copied by the expansion (Section 4.3). This enables us to show the commutativity Theorem 4.3: the expansion of the reducts of a path π is equivalent to the expanded reducts of the expansion of π. As a consequence, we can also show that persistence property can be transferred along expansion, i.e. π persists to MELL reduction if and only if there is π' in its expansion which persists to RIN reduction (Theorem 4.4). Lastly, we exploit these last results to define a variant of the execution formula for the λ-calculus based on the GoRI (Chapter 3) — if we assign to every path π in a proof-net the infinite sets of $\mathsf{r}\mathcal{L}^*$-weights belonging to expansions of π, we obtain an expanded formula enjoying the properties of our interest: invariance under reduction (Theorem 4.5), hence characterisation of persistence (Corollary 4.1).

Most of the original material of the chapter will appear in MSC [2016].

4.2 Taylor-Ehrhard-Regnier expansion

4.2.1 Net expansion

Following definitions in the previous literature [e.g. Mazza and Pagani, 2007, Pagani and Tasson, 2009], we introduce here the notion of Taylor-Ehrhard-Regnier expansion, as a map from mMELL proof-nets (see Definition 2.10 and 2.4) to infinite sums of resource interaction nets (see Definition 3.3). Although the challenging study of multiplicity coefficients is postponed to future investigations, we preserve coherence and employ sums to put together simple nets, even though the sum is idempotent so they essentially represent the support of the infinite series used in the original definition. Another particularity of the definition that is worth to remark is the fact that links' premises are sequences, and not mere sets, of vertices, which means that essentially co-contraction is represented in a non-commutative way. Because of this, the definition of expansion reflects this information growth and preserve the ordering of all contraction links whose premises are involved in the expansion of a box.

Definition 4.1 (Mixed nets). A *simple mixed net* is a pre-net built with resource or mMELL links, i.e. possibly containing co-contractions and promotions, and a boxing function which assign boxes to promotions (not to co-contractions, nor its premises). A *mixed net* is a possibly infinite sum of simple mixed nets. The sum + is not only associative, commutative, and having an identity element, i.e. the empty sum 0, but it is also idempotent: $\mathcal{M} + \mathcal{M} = \mathcal{M}$.

Definition 4.2 (Net expansion). Given a mixed pre-net $\mathcal{N}$, and $\overline{\mathcal{B}} \in \mathrm{Bxs}\,(\mathcal{N})$, the i-*ary box expansion* of $\overline{\mathcal{B}}$, written $\overline{\mathcal{B}}^{\mathrm{i}}$, is depicted in Figure 4.1. Contractions' premises are duplicated together with $\mathcal{B}$ enforcing stability with respect to their ordering: given

Figure 4.1 i-ary box expansion.
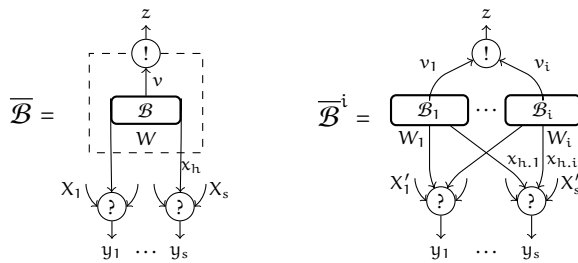
a contraction $\langle X_j \; (?) \; y_j \rangle$ such that there exists $x_h \in X_j$ secondary door of $\mathcal{B}$, in the expansion such a contraction becomes $\langle X'_j \; (?) \; y_j \rangle$, where the sequence X'_j is obtained from X_j by replacing any such x_h with the sequence $x_{h.1}, \dots, x_{h.i}$.

The *outermost expansion*, is a function from simple to generic mixed nets, written as $\mathcal{M}^\circ$ when applied to $\mathcal{M}$, and defined by induction on $\ell(\mathcal{M})$. If $\ell(\mathcal{M}) = 0$, then simply $\mathcal{M}^\circ = \mathcal{M}$. Otherwise, if $\overline{\mathcal{B}}_1, \dots, \overline{\mathcal{B}}_n$ are the outermost bordered boxes of $\mathcal{M}$, and $\mathbb{N}^n$ is the set of functions from $\{1, \dots, n\}$ to $\mathbb{N}$, then:

$$\mathcal{N}^\circ \stackrel{\text{def}}{=} \sum_{a \in \mathbb{N}^n} \mathcal{N}^a, \tag{4.1}$$

$$\mathcal{N}^a \stackrel{\text{def}}{=} \mathcal{N}\left[\overline{\mathcal{B}}_1^{a(1)} \middle/ \overline{\mathcal{B}}_1, \dots, \overline{\mathcal{B}}_n^{a(n)} \middle/ \overline{\mathcal{B}}_n \right]. \tag{4.2}$$

Observe that in (4.2) all substitutions are well-defined, since $\text{iface}\left(\overline{\mathcal{B}}_i^{a(i)} \right) = \text{iface}\left(\overline{\mathcal{B}}_i \right)$ for any $1 \leq i \leq n$. We define the *complete expansion* of a simple mixed net $\mathcal{M}$ as the fixed point of the outermost expansion, whose domain is extended to generic mixed nets.

$$0^\circ \stackrel{\text{def}}{=} 0 \tag{4.3}$$

$$\left(\sum \mathcal{M} \right)^\circ \stackrel{\text{def}}{=} \sum \mathcal{M}^\circ \tag{4.4}$$

$$\mathcal{M}^\bullet \stackrel{\text{def}}{=} (\mathcal{M}^\bullet)^\circ \tag{4.5}$$

Observe that the complete expansion may generate an (infinite) number of equal simple nets which collapse to a single simple net, by sum's idempotence. The *support* of a mixed net $\mathcal{M}$ is the set $\text{Supp}(\mathcal{M})$ of all addends in $\mathcal{M}$. An element in $\text{Supp}(\mathcal{M}^\bullet)$ is called an *expansion* of $\mathcal{N}$.

Example 4.1 (Net expansion). Consider first the mMELL proof-net $\mathcal{N} = [\![I(\star)]\!]$, and its complete expansion $\mathcal{N}^\bullet$. Let $\mathcal{B}$ be the only box in $\mathcal{N}$, which contains the $\star$-link. Then $\mathcal{N}^\bullet$ has the cardinality of $\mathbb{N}$, since:

$$\mathcal{N}^\bullet = \sum_{i=0}^{\infty} \mathcal{N}\left[\overline{\mathcal{B}}^i \middle/ \overline{\mathcal{B}} \right] \tag{4.6}$$

Notice also that $\mathcal{N}\left[\overline{\mathcal{B}}^1 \middle/ \overline{\mathcal{B}} \right]$ is the only addend that does not reduce to 0, because $[\![I]\!]$ uses its argument once.

Now take $\mathcal{N}' = [\![(I(I))(\star)]\!]$ and observe that the term contains two disjoint arguments, i.e. I and $\star$, therefore its translation contains two disjoing boxes, say $\mathcal{B}_I$ and $\mathcal{B}_\star$, respectively. Hence $(\mathcal{N}')^\bullet$ is in bijection with $\mathbb{N} \times \mathbb{N}$, because:

$$(\mathcal{N}')^\bullet = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \mathcal{N}'\left[\overline{\mathcal{B}}_I^i \middle/ \overline{\mathcal{B}}_I, \overline{\mathcal{B}}_\star^j \middle/ \overline{\mathcal{B}}_\star \right]. \tag{4.7}$$

Figure 4.2 Example: a mMELL proof-net (left) and two simple mixed nets that belong to its expansion (middle and right). See also Example 4.1, 4.3, 4.4.
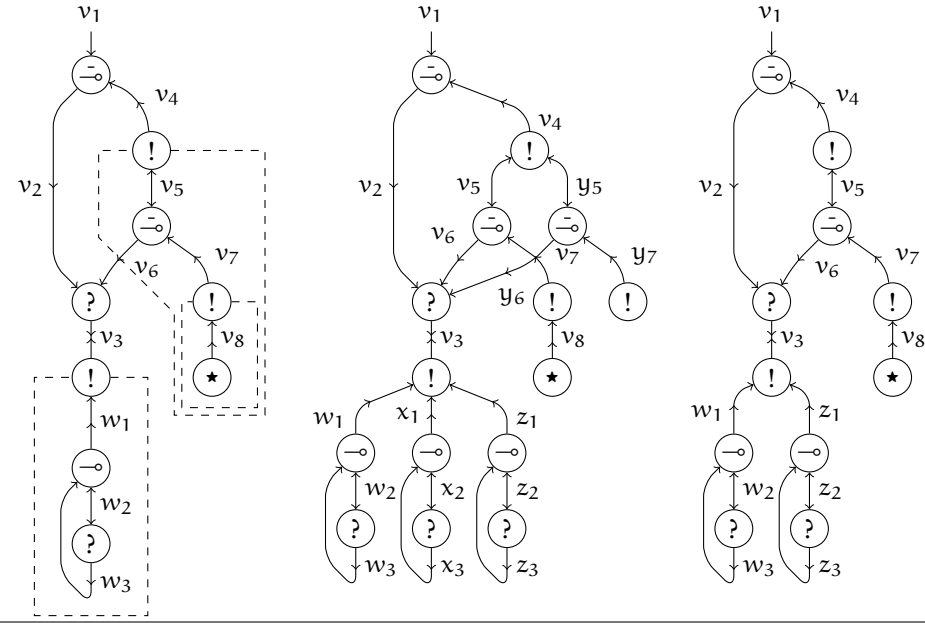
Now consider the proof-net $\mathcal{N}''$ in Figure 4.2 (left). It contains three boxes: let $\mathcal{B}_f$ be the box containing for $[\![f(\star)]\!]$, let $\mathcal{B}_I$ the one with $[\![I]\!]$, and $\mathcal{B}_\star$ that with $[\![\star]\!]$. Since $\mathcal{B}_f \supset \mathcal{B}_\star$, here we see that the expansion becomes quite more complicated than before:

$$(\mathcal{N}'')^\bullet = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \sum_{k_1=0}^{\infty} \ldots \sum_{k_i=0}^{\infty} \mathcal{N}'' \left[\overline{\mathcal{B}}_f^i / \mathcal{B}_f, \overline{\mathcal{B}}_I^j / \mathcal{B}_I, \overline{\mathcal{B}}_\star^{k_1} / \mathcal{B}_{\star_1}, \ldots, \overline{\mathcal{B}}_\star^{k_i} / \mathcal{B}_{\star_i} \right]. \quad (4.8)$$

Observe in particular that the expansion of $\mathcal{B}_\star$ is replicated by the expansion of its containing box $\mathcal{B}_f$: each addend first selects a number i of copies of the latter, then it selects i numbers $k_1, \ldots, k_i$ of copies, one for each copy of the former induced by the first step. $(\mathcal{N}'')^\bullet$ has therefore the cardinality of $\mathbb{N}^{\mathbb{N}}$. Look again at Figure 4.2 and find:

- in the middle: $\mathcal{N}'' \left[\overline{\mathcal{B}}_f^2 / \mathcal{B}_f, \overline{\mathcal{B}}_I^3 / \mathcal{B}_I, \overline{\mathcal{B}}_\star^1 / \mathcal{B}_{\star_1}, \overline{\mathcal{B}}_\star^0 / \mathcal{B}_{\star_2} \right]$;

- on the right: $\mathcal{N}'' \left[\overline{\mathcal{B}}_f^1 / \mathcal{B}_f, \overline{\mathcal{B}}_I^2 / \mathcal{B}_I, \overline{\mathcal{B}}_\star^1 / \mathcal{B}_{\star_1} \right]$.

Along the normalisation of the latter, the first step generates 3! identical addends, each of which originates in 3 multiplicative steps 3 exponential redexes. 2 of them reduce to 0, hence the normal form is 0.

Finally, remark also that the three complete expansions, i.e. $\mathcal{N}^\bullet$, $(\mathcal{N}')^\bullet$ and $(\mathcal{N}'')^\bullet$, are infinite sums of simple resource nets. This property will be generalised later by Proposition 4.2.

Remark 4.1 (The structure of expansion). The support of the complete expansion of a proof-net unveils its elegance once we abstract away the multiplicative structure from the proof-net, and keep only the nesting relation between boxes: a tree. In this abstract formulation, where the complete expansion is a set of hypertrees (a similar definition has been independently formalised by Guerrieri and Tortora de Falco [2014]), it is easy

to define the relation between elements of the expansion, that intuitively represents the concept of being "recursively less expanded than". Such a relation can be proven not only to be reflexive, transitive and antisymmetric, hence forming unsurprisingly a partial ordering, but also to enjoy the existence of supremum and infimum elements, thus forming a lattice. Even more, the structure admits the existence of minimum elements, but not that of maximum elements. The natural completion of this structure that would enable it to be a complete lattice are (hypertrees of) resource nets where co-contractions allow infinite arities. We conjecture that this structure corresponds, very naturally, to the full differential nets, i.e. resource nets with promotions.

4.2.2 Term expansion and translation

Although the present chapter employs the fine-grained formulation of programs as nets, all the results about resource nets and expansion can be easily translated to resource calculus. To show it explicitly we now observe that the translation from terms to nets, and the expansion from the ordinary to the resource domains commutes — the resource-net translation of the term expansion of a λ-term t is equal to the net expansion of the mMELL-translation of t.

Definition 4.3 (Term expansion). The *term expansion* is the function from ordinary λ-terms to possibly infinite sums of resource terms defined by the following structural induction on the λ-syntax.

$$x^{\bullet} \stackrel{\text{def}}{=} x \tag{4.9}$$

$$(\lambda x.t)^{\bullet} \stackrel{\text{def}}{=} \lambda x.t^{\bullet} \tag{4.10}$$

$$(t\,s)^{\bullet} \stackrel{\text{def}}{=} \sum_{i=0}^{\infty} t^{\bullet}[\underbrace{s^{\bullet}, \ldots, s^{\bullet}}_{i}] \tag{4.11}$$

Example 4.2 (Term and net expansions). Recall that $[s^n]$ is a shortand for $[\overbrace{s, \ldots, s}^{i}]$. Now consider $I(\star)$ and its expansion: $(I(\star))^{\bullet} = \sum_{i=0}^{\infty} I[\star^i]$. We notice that $\llbracket I(\star) \rrbracket^{\bullet} = \llbracket \sum_{i=0}^{\infty} I[\star^i] \rrbracket$. Does the same hold for $t = (\lambda f.f_{@1}(f_{@2}(\star)))(I)$? Its expansion is:

$$t^{\bullet} = \left(\lambda f.f_{@1} \left[\sum_{i=0}^{\infty} f_{@2} \left[\sum_{k_1=0}^{\infty} \star \right], \ldots, f_{@i+1} \left[\sum_{k_i=0}^{\infty} \star \right] \right] \right) \left[\sum_{j=0}^{\infty} I^j \right], \tag{4.12}$$

where we observe that the number of occurrences of the variable f varies with the choice of the number of copies of the argument $f_{@2}(\star)$. Consider $\llbracket t \rrbracket$, which can be obtained by a multiplicative step of anti-reduction from the proof-net $\mathcal{N}''$ depicted in Figure 4.2. With the help of Example 4.1, where we explored $(\mathcal{N}'')^{\bullet}$ (see in particular Equation 4.8), verify that $\llbracket t \rrbracket^{\bullet} = \llbracket t^{\bullet} \rrbracket$. The next proposition generalises such intuition.

Proposition 4.1 (Commutativity of translation and expansion). *The net expansion of a mMELL-translated term is equal to the RIN-translation of its expansion. For any* $t \in \Lambda$,

$$\llbracket t \rrbracket^{\bullet} = \llbracket t^{\bullet} \rrbracket \tag{4.13}$$

Proof. The definitions of ordinary and resource terms translations into proof-nets and resource nets are not uniformly inductive, since they both need the final step which adds ?-link on all in conclusions that represent variable occurrences, (cf. Definition 2.12 and 3.5). Therefore we will first work with the pre-translation, which can be easily handled, and then consider the final step.

1. We go by induction on the syntax of t, and we will prove as a sub-claim that $\left(\langle t \rangle^{\Gamma}\right)^{\bullet} = \langle t^{\bullet} \rangle^{\Gamma'}$, for any variable to vertices injection Γ and some, appropriately chosen one, Γ'. We will omit to explicitly show that such claim holds conversely hold for any Γ' and some Γ, since it will be evident from the proof.

   (a) $t = \star$. Then $[\![t]\!]^{\Gamma} = \langle (\star)\, v \rangle$ for some vertex v, and $\left(\langle t \rangle^{\Gamma}\right)^{\bullet} = \langle (\star)\, v \rangle$. On the other hand, $t^{\bullet} = \star$ and we immediately conclude $\langle t^{\bullet} \rangle^{\Gamma} = \langle (\star)\, v \rangle = \left(\langle t \rangle^{\Gamma}\right)^{\bullet}$ ◆

   (b) $t = x$. Almost identically to previous case, we have $\langle t \rangle^{\Gamma} = v$, i.e. a mixed net with no links and with one vertex v, therefore also $\left(\langle t \rangle^{\Gamma}\right)^{\bullet} = v$. But given that $t^{\bullet} = x$, we already have $\langle t^{\bullet} \rangle^{\Gamma} = v = \left(\langle t \rangle^{\Gamma}\right)^{\bullet}$. ◆

   (c) $t = \lambda x.s$. This inductive case is still quite simple. On the one hand we have

   $$\langle t \rangle^{\Gamma} = \langle u_1, u_2 \,(\multimap)\, v \rangle, \langle w_1, \ldots w_n \,(?)\, u_2 \rangle, \langle s \rangle^{\Gamma} \qquad (4.14)$$

   and trivially

   $$\left(\langle t \rangle^{\Gamma}\right)^{\bullet} = \langle u_1, u_2 \,(\multimap)\, v \rangle, \langle w_1, \ldots w_l \,(?)\, u_2 \rangle, \left(\langle s \rangle^{\Gamma}\right)^{\bullet}, \qquad (4.15)$$

   where the arity l of the ?-link depends on $\left(\langle s \rangle^{\Gamma}\right)^{\bullet}$. With the same simplicity, since by definition of expansion $t^{\bullet} = \lambda x.s^{\bullet}$, we know that its translation is:

   $$\langle t^{\bullet} \rangle^{\Gamma'} = \langle u_1, u_2 \,(\multimap)\, v \rangle, \langle w_1, \ldots w_m \,(?)\, u_2 \rangle, \langle s^{\bullet} \rangle^{\Gamma'}, \qquad (4.16)$$

   where, similarly to Equation 4.15, the arity m depends on $\langle s^{\bullet} \rangle^{\Gamma'}$. Now, by inductive hypothesis, we have $\left(\langle s \rangle^{\Theta}\right)^{\bullet} = \langle s^{\bullet} \rangle^{\Theta'}$ for any Θ and some Θ'. Therefore for any choice of $\mathrm{Supp}\left(\left(\langle s \rangle^{\Theta}\right)^{\bullet}\right) \ni s' \in \mathrm{Supp}\left(\langle s^{\bullet} \rangle^{\Theta'}\right)$, it must be the case that $l = m$. Therefore we conclude by taking $\Gamma = \Theta$ and $\Gamma' = \Theta'$. ◆

   (d) $t = (r\,s)$. This inductive case is the really interesting one, since here we can see expansion at work. First of all:

   $$\langle t \rangle^{\Gamma} = \langle w, v \,(\bar{\multimap})\, u \rangle, \langle x \,(!)\, w \rangle, \langle r \rangle^{\Gamma}, \langle s \rangle^{\Gamma}, \qquad (4.17)$$

   where $u \in \mathrm{iface}\left(\langle r \rangle^{\Gamma}\right)$ and $x \in \mathrm{iface}\left(\langle s \rangle^{\Gamma}\right)$, both unique elements with out polarity; and $\langle s \rangle^{\Gamma}$ is enclosed in a new box. Therefore, by definition of expansion:

   $$\left(\langle t \rangle^{\Gamma}\right)^{\bullet} = \sum_{i=0}^{\infty} \langle w, v \,(\bar{\multimap})\, u \rangle, \langle x_1, \ldots, x_i \,(!)\, w \rangle,$$
   $$\left(\langle r \rangle^{\Gamma}\right)^{\bullet}, \left(\langle s_1 \rangle^{\Gamma}\right)^{\bullet}, \ldots \left(\langle s_i \rangle^{\Gamma}\right)^{\bullet}, \qquad (4.18)$$

where x_j is the out vertex in $\text{iface}\left(\left(\langle s_j\rangle^\Gamma\right)^\bullet\right)$, and for any $1 \le j, j' \le i$ we have $\text{iface}\left(\left(\langle s_j\rangle^\Gamma\right)^\bullet\right) = \text{iface}\left(\left(\langle s_{j'}\rangle^\Gamma\right)^\bullet\right)$. Now, looking at the right-hand side of Equation 4.13, we have that

$$t^\bullet = \sum_{i=0}^\infty r^\bullet\left[s_1^\bullet, \ldots, s_i^\bullet\right]. \tag{4.19}$$

So consider the pre-translation with respect to any Γ':

$$\langle t^\bullet\rangle^{\Gamma'} = \sum_{i=0}^\infty \langle w, v\,(\bar{-}\circ)\,u\rangle,\ \langle x_1, \ldots, x_i\,(!)\,w\rangle,$$
$$\langle r^\bullet\rangle^{\Gamma'},\ \langle s_1^\bullet\rangle^{\Gamma'}, \ldots, \langle s_i^\bullet\rangle^{\Gamma'}, \tag{4.20}$$

where u is the unique out conclusion in $\text{iface}\left(\left(r^\bullet\right)^{\Gamma'}\right)$ and for any $1 \le j \le i$ x_j is the unique out conclusion in $\mathrm{I}\left(\left(s_j^\bullet\right)^{\Gamma'}\right)$. Now, let $y_{@h'}$ be one the variable occurrences of $s_j^\bullet$, and assume it to be the expansion of the variable occurrence $y_{@h}$ of s. To prove the claim we need to impose that $(\Gamma(y_{@h}))^\bullet \ni \Gamma'(y_{@h'})$. But this is always possible since, by inductive hypothesis, we have not only that $\left(\langle r\rangle^\Theta\right)^\bullet = \langle r^\bullet\rangle^{\Theta'}$, for any Θ and some Θ' (and vice versa); but also that $\left(\langle s_j\rangle^\Xi\right)^\bullet = \langle s_j^\bullet\rangle^{\Xi'}$, for any Ξ and some Ξ' (and vice versa). $\quad\blacklozenge$

$\square$

2. Now we consider the final steps of translations. $\llbracket t\rrbracket^\Gamma = \left\langle\langle t\rangle^\Gamma\int^\Gamma\right.$. In step 1 we verified that $\left(\langle t\rangle^\Gamma\right)^\bullet = \langle t^\bullet\rangle^{\Gamma'}$, and we showed the existence of Γ, Γ'. Now, let $\mathcal{N}$ be the sum of simple resource nets obtained from $\left(\langle t\rangle^\Gamma\right)^\bullet$ by adding to it, for any variable y of t, a ?-link having as premiss any vertex belonging to $(\Gamma(y_h))^\bullet$, for any occurrence h of y. First of all, we easily verify that, by definition of expansion, $\mathcal{N} = \left(\llbracket t\rrbracket^\Gamma\right)^\bullet$. Then, we observe that in $\llbracket t^\bullet\rrbracket^{\Gamma'}$ the translation similarly adds to $\langle t^\bullet\rangle^{\Gamma'}$ a ?-link, for any variable y' of $t^\bullet$, a ?-link having as premiss any vertices in $\Gamma(y_{h'}')$ for any occurrence h' of y'. Therefore, by definition of Γ, Γ' and by the conclusion of step 1, we have $\llbracket t^\bullet\rrbracket^{\Gamma'} = \mathcal{N}$. Hence the claim. $\quad\square$

$\blacksquare$

4.2.3 Path expansion

Observing the action of expansion on paths, we can identify the expansion of a path π in a proof-net $\mathcal{N}$ as the series of paths which belong to the simple resource nets in the complete expansion of $\mathcal{N}$ and which correspond to π.

Recall the definition of paths in a mMELL proof-net, or in a RIN, and the definition of the static properties which are necessary to ensure their persistence, hence meaningfulness as a description of computation: straightness, maximality, and comprehensiveness (see Definition 2.6, 2.17, 3.9).

Definition 4.4 (Box crossing). Given a proof-net N and $\pi \in P(N)$, its *outermost-box crossing form* is:

$$\text{OBCF}(\pi) \overset{\text{def}}{=} \epsilon_0 :: \beta_1 :: \epsilon_1 :: \ldots :: \beta_n :: \epsilon_n, \tag{4.21}$$

where, for every $1 \leq i \leq n$, β_i is a *box crossing*, that is a maximal subpath belonging to an outermost box $\mathcal{B}$, i.e. $\ell(\mathcal{B}) = 0$. Notice that n may be null and that a crossing is not necessarily a maximal path in $\mathcal{B}$.

Definition 4.5 (Path expansion). Given N a mixed net and $\pi \in P(N)$, let $\overline{\mathcal{B}}_1, \ldots, \overline{\mathcal{B}}_m$ be the outermost bordered boxes of N, let $\text{OBCF}(\pi)$ be as in Equation 4.21, and let $a \in \mathbb{N}^m$. Then the *a-ary outermost expansion* of π in N is the set of paths obtained by the action of the outermost expansion of N according to a.

$$\pi^{N^a} = \begin{cases} \{\pi\} & \text{if } m = 0, \\ \left\{ \epsilon_0 :: \beta_{1_{k_{b(1)}}} :: \epsilon_1 :: \ldots :: \beta_{n_{k_{b(n)}}} :: \epsilon_n \,\middle|\, 1 \leq k_i \leq a(i) \right\} & \text{otherwise,} \end{cases} \tag{4.22}$$

where for each crossing β_j of $\mathcal{B}_{b(j)}$, we write $\beta_{j_{k_{b(j)}}}$ to the denote the image of β_j in the $k_{b(j)}$th copy of the box $\mathcal{B}_{b(j)}$. Notice that the rightmost side of Equation 4.22 is empty whenever there exists $1 \leq i \leq m$ such that $a(i) = 0$; while it is the singleton $\{\epsilon_0\}$ if $n = 0$. The *outermost expansion* of a set of paths Π is the sum of sets of paths generated by the sum of all a-ary outermost expansions, extending its domain to sets of path. The sum of paths satisfies the same properties of the sum of simple resource nets.

$$\Pi^{N^a} \overset{\text{def}}{=} \bigcup_{\pi \in \Pi} \pi^{N^a} \tag{4.23}$$

$$\Pi^{N^\circ} \overset{\text{def}}{=} \begin{cases} \Pi & \text{if } m = 0, \\ \displaystyle\sum_{a \in \mathbb{N}^m} \Pi^{N^a} & \text{otherwise.} \end{cases} \tag{4.24}$$

The *complete expansion* of a sum of sets of paths Π is the fixpoint of the outermost expansion, where we neatly extend its domain to sums of sets of nets.

$$\mathbf{\Pi}^{\mathbf{N}^\circ} \overset{\text{def}}{=} \sum_{\substack{N \in \text{Supp}(\mathbf{N}) \\ \Pi \supseteq \Pi \subseteq P(N)}} \Pi^{N^\circ} \tag{4.25}$$

$$\mathbf{\Pi}^{\mathbf{N}^\bullet} \overset{\text{def}}{=} \left(\mathbf{\Pi}^{\mathbf{N}^\bullet}\right)^{\mathbf{N}^\circ} \tag{4.26}$$

When it does not lead to confusion, we shall ease the notation of $\{\pi\}^{N^\bullet}$ writing $\pi^{N^\bullet}$. The *complete expansion of a vertex* v is the sum of sets of vertices that is naturally obtained by the complete expansion of the atomic path (v):

$$v^{N^\bullet} \overset{\text{def}}{=} \sum_{\Pi \in \text{Supp}((v)^{N^\bullet})} \{v' \mid (v') \in \Pi\}. \tag{4.27}$$

We similarly define the *complete expansion of a link* l in N as the sum of sets of links defined as follows.

$$\langle u \, (\kappa) \, v \rangle^{N^\bullet} \overset{\text{def}}{=} \sum_{\substack{V \in \text{Supp}\left(v^{N^\bullet}\right) \\ U \in \text{Supp}\left(u^{N^\bullet}\right)}} \left\{ l' = \langle u' \, (\kappa') \, v' \rangle \,\middle|\, v' \in V, \; u' \in U, \; l' \in L(N^\bullet) \right\} \tag{4.28}$$

Example 4.3 (Path expansion, a gentle case). Recall from Example 4.1 the proof-net $N = [\![I(\star)]\!]$ for which, as we previously discussed, we have $\text{Bxs}(N) = \{B\}$. Recall also its complete expansion. Let $\pi \in P(N)$ that: starts from the root, crosses the $\multimap$-link from the conclusion to the second premiss, crosses the $\multimapinv$-link from the conclusion to the first premiss, and reaches the $\star$-link (all omitted crossings can be inferred by straightness). Recall from Equation 4.6 that for every $i \in \mathbb{N}$ there is a unique $N_i \in \text{Supp}(N^\bullet)$, i.e. $N[\overline{B}^i/\overline{B}]$. Now, observe that in $\text{Supp}(\pi^{N^\bullet})$ there is a unique set of paths Π_i for every expansion of N. Moreover, since π crosses once the !-link entering B, and since N_i include exactly i copies of B, we have that the cardinality of Π_i is i. This means in particular that $\Pi_0 = \varnothing$. Notice also that $N_i \to_r 0$ for any $i \neq 1$, and that both π and π_1 are persistent paths.

Example 4.4 (Path expansion, two less gentle cases). Look again at Figure 4.2, where the proof-net N and two simple mixed nets M, M' are depicted (from left to right). As we discussed in preceding Example 4.1, let B_I be the box which contains $[\![I]\!]$, and remember that $M, M' \in \text{Supp}(N^\bullet)$.

Let $\phi \in P(N)$ be defined as follows:

$$\phi = (v_1, v_2, v_3, w_1, w_2, w_3$$
$$w_1, v_3, v_2,$$
$$v_4, v_5, v_6, v_3, w_1, w_2, w_3). \tag{4.29}$$

If $M_i \in \text{Supp}(N^\bullet)$ is one of the simple mixed nets that contains i copies of B_I, notice that there exists a unique $\Phi_i \in \text{Supp}(\phi^{N^\bullet})$ that contains any $\phi' \in P(M_i)$ such that $\phi' \in \Phi'_i \in \text{Supp}(\phi^{N^\bullet})$. This will be generalised by Fact 4.3.

Now remark that ϕ enters twice in B_I. In N'' we find 4 expanded paths of ϕ, 9 in M'. The expansion of ϕ essentially computes an assignment of a copy of B to each crossing. Hence $|\Phi_i| = i^2$ if $i > 0$, 0 otherwise. More generally if ϕ had crossed B_I n times, then $|\Phi_i| = i^n$, i.e. the number of n-tuples over $(1, \dots, i)$.

Consider $\pi \in P(N)$ defined as follows:

$$\pi = (v_3, v_6, v_7, v_8). \tag{4.30}$$

Remark that π crosses once both B_f (the box with principal door in v_4) and $B_\star$ (the box with principal door in v_7). Notice that there is only one expanded path in M'', where both boxes are expanded in one copy, i.e. (v_3, v_6, v_7, v_8). But there is again only one expanded path of π even in M', where B_f is expanded twice, i.e. (v_3, v_6, v_7, v_8), since in its second copy $B_\star$ has 0 copies (so there is only a prefix available: (v_3, y_6, y_7)). To see the more general picture, let $M_{i,j_1,\dots,j_i} \in \text{Supp}(N^\bullet)$ be one of the simple mixed nets that contains i copies of B_f, and j_i copies of $B_{\star_i}$, that is the i-th copy of $B_\star$. Let $\Pi_{i,j_1,\dots,j_i}$ be the set of path belonging to the expansions of π inhabiting $M_{i,j_1,\dots,j_i}$. We can easily verify that $|\Pi_{i,j_1,\dots,j_i}| = i \times j_i$. More generally, if π had contained n crossings of the two boxes, we would have had $|\Pi_{i,j_1,\dots,j_i}| = (i \times j_i)^n$.

Path expansion does not expand a path in the sense of increasing its length, since a path in the expansion can be constructed by a vertex-by-vertex procedure. Also, expansion preserves kind and in/out polarities of links, and types of vertices.

Fact 4.1 (Path expansion and length). *The size of a path is preserved in its expansion's addends: for any $\pi \in P(N)$, and any $\pi' \in \pi^{N^\bullet}$, $|\pi| = |\pi'|$.*

Fact 4.2. *For any mMELL proof-net $\mathcal{N}$, let $l = \langle U \ (\kappa) \ v \rangle \in L(\mathcal{N})$ and $l' = \langle U' \ (\kappa') \ v' \rangle \in L' \in \mathrm{Supp}\left(l^{\mathcal{N}^\bullet}\right)$. Then:*

1. *$\kappa = \kappa'$;*

2. *$\mathcal{T}(v) = \mathcal{T}(v')$;*

3. *for any $u \in U$ and $u \in U'$, if $u' \in U'' \in \mathrm{Supp}\left(u^{\mathcal{N}^\bullet}\right)$ then $\mathcal{T}(u) = \mathcal{T}(u')$.*

Given a path π in a mixed net $\mathcal{N}$, the sets of paths which form the expansion of π are in bijection with the simple mixed nets which form the expansion of $\mathcal{N}$. This means that two paths belongs to the same set of paths that appears in the expansion of π if and only if they inhabit the same mixed net. Moreover, if path expansion is considered as a relation between the paths of a simple mixed net and those of another simple mixed net appearing in its complete expansion, then path expansion is injective, or: its inverse relation is a function.

Fact 4.3 (Bijection between addends). *Given $\mathcal{N}$ a mixed net and $\pi \in P(\mathcal{N})$, let $\pi' \in P(\mathcal{N}')$ and $\pi'' \in P(\mathcal{N}'')$ such that $\mathcal{N}', \mathcal{N}'' \in \mathrm{Supp}(\mathcal{N}^\bullet)$. Also, let $\pi' \in \Pi'$ and $\pi'' \in \Pi''$ such that $\Pi', \Pi'' \in \mathrm{Supp}\left(\pi^{\mathcal{N}^\bullet}\right)$. Then $\mathcal{N}' = \mathcal{N}''$ if and only if $\Pi' = \Pi''$.*

Lemma 4.1 (Path injectivity). *Given two mixed nets $\mathcal{N}$ and $\mathcal{N}' \in \mathrm{Supp}(\mathcal{N}^\bullet)$, for any path $\pi' \in P(\mathcal{N}')$:*

1. *there exist $\pi \in P(\mathcal{N})$ and $\Pi \in \mathrm{Supp}\left(\pi^{\mathcal{N}^\bullet}\right)$ such that $\pi' \in \Pi$;*

2. *for any $\gamma \in P(\mathcal{N})$, if $\pi' \in \Gamma \in \mathrm{Supp}\left(\gamma^{\mathcal{N}^\bullet}\right)$ then $\Gamma = \Pi$ and $\gamma = \pi$.*

Proof. By immediate verification against the definition of path expansion. ∎

Remark 4.2 (Non-injectivity w.r.t. simple nets). The expansion functions of paths, of vertices, and of links are not injective with respect to expansions of the containing net. Let us clarify with a counterexample for the case of vertices. Let $v \in V(\mathcal{N})$ for some mixed net $\mathcal{N}$, and consider v', v'' appearing in $v^{\mathcal{N}^\bullet}$, i.e. such that there exist $V' \in v^{\mathcal{N}^\bullet}$ for which $v' \in V''$, and $V'' \in v^{\mathcal{N}^\bullet}$ for which $v'' \in V''$. Now, if $v' \neq v''$, it may be the case that $V' \neq V''$, which means that v', v'' belongs to different addends of $\mathcal{N}^\bullet$, but we may also have $V' = V''$, which implies the converse.

Since path expansion preserve switchingness and acyclicity, we can finally show that expanded nets are sums of simple resource nets.

Lemma 4.2. *Let $\mathcal{N}$ be a mixed net. If a path $\pi' \in P(\mathcal{N}^\bullet)$ is switching (or acyclic), then there exists $\pi \in P(\mathcal{N})$ which is switching (respectively acyclic).*

Proof. We simply use the injectivity property of vertex expansion to show that switchingness and acyclicity cannot be introduced by path expansion. We separately show inheritance of the two properties.

1. Switching. Recall that by Definition 2.9 given any v'_1, v'_2 premises of the same link of kind $\multimap$ or $?$, π' is switching when if π' contains v'_1, then it does not contain v'_2. Now, let $v_1, v_2 \in V(\mathcal{N})$ be such that $v'_1 \in V_1 \in \text{Supp}\left(v_1^{\mathcal{N}\bullet}\right)$ and $v'_2 \in V_2 \in \text{Supp}\left(v_2^{\mathcal{N}\bullet}\right)$, whose existence and unicity is established by item 4.2.3. Now let π such that $\pi' \in \Pi \in \text{Supp}\left(\pi^{\mathcal{N}\bullet}\right)$, and assume that $v'_1 \in \pi'$. Then, by definition of path expansion, it must be the case that $v_1 \in \pi$. Now, for the same reason, if $v'_2 \notin \pi'$ then $v_2 \notin \pi$, hence π is switching. □

2. Cyclic. By Definition 2.9, π' is cyclic if it contains a cycle $\gamma' = (v'_0, v'_1, \ldots, v'_n, v'_0)$, with $n > 0$. Because of Lemma 4.1, there are two unique paths γ, π such that $\gamma' \in \Gamma \in \text{Supp}\left(\gamma^{\mathcal{N}\bullet}\right)$ and $\pi' \in \Pi \in \text{Supp}\left(\pi^{\mathcal{N}\bullet}\right)$. But by definition of path expansion, we know that if $\gamma' \subseteq \pi'$, then also $\gamma \subseteq \pi$. Moreover item 4.2.3 establishes that there exists a unique $v_0 \in V(\mathcal{N})$ such that $v'_0 \in \text{Supp}\left(v_0^{\mathcal{N}\bullet}\right)$, hence γ is a cycle, and π is cyclic. □

■

Proposition 4.2. *The expansion of a mMELL proof-net is a possibly infinite sum of simple resource interaction nets.*

Proof. We directly show that any path in any expansion satisfies Definition 3.4. Let $\pi' \in \mathbb{P}(\mathcal{N}')$, for some resource pre-net $\mathcal{N}' \in \text{Supp}(\mathcal{N}^\bullet)$ and for some proof-net $\mathcal{N}$. Let $\pi \in \mathbb{P}(\mathcal{N})$ such that $\pi' \in \Pi' \in \text{Supp}\left(\pi^{\mathcal{N}\bullet}\right)$, whose existence is guaranteed by Lemma 4.1. Now, by last Lemma 4.2, if π' is switching (or cyclic) then also π is switching (respectively, cyclic). But by Definition 2.10 of mMELL proof-nets, if π is switching then it is not cyclic, which implies that there cannot exist π' switching and cyclic. So we conclude that any $\mathcal{N}'$ satisfies the definition of simple resource net, and consequently that $\mathcal{N}^\bullet$ is a resource net. ■

4.3 Expansion and reduction

Taylor expansion replaces boxes in a mMELL net with a number of copies of its content. Can we reduce all copies of a same redex together, in an infinitary parallel reduction? We define such a reduction relation, which in a sense represent the Taylor expansion of the mMELL cut-elimination, and which we call 'expanded'.

We first define a notion of parallel reduction on resource nets that is a restriction of that employed by Pagani and Tranquilli [2009] to show confluence of resource calculus. In the settings of resource interaction nets a parallel reduction is indeed not hard to formulate, since, oppositely to mMELL proof-nets, λ-terms and resource λ-terms, the RIN rewriting system does not allow overlaps, as mentioned in Proposition 3.5. Our limitation simply aims at tightening it to ease the simulation of mMELL reduction, and is inspired from the following observation, directly following from Fact 4.2.

Fact 4.4. *A mMELL vertex is a cut if and only if its expansion is a sum of sets of cuts.*

Definition 4.6 (Multi-hole contexts and substitution)**.** A *simple multi-hole context* $C[\![\,]\!]$, (resp. a *multi-hole context* $C[\![\,]\!]$) is a simple resource context (resp. a linear combination

of simple resource contexts over the semiring of $\mathbb{N}$) having a finite, possibly null, number of hole links (see Definition 2.14 and recall that !-links are not promotions, nor they have associated boxes). If $C[\ ]$ has holes $h_1, \ldots, h_n$ and $\mathcal{N}$ is a sum of simple resource pre-nets $\mathcal{P}_1, \ldots, \mathcal{P}_m$, then $C[\![\mathcal{N}]\!]$ consists of the (possibly empty) sum of any possible substitution of the m (non-zero) addends of $\mathcal{N}$ into the n holes of $C[\ ]$. Formally, if $\mathbb{N}_k$ denotes $\{1, \ldots, k\}$ (where $\mathbb{N}_0 = \{\}$), and $A \rightarrowtail B$ the set of injections from a set A to a set B:

$$C[\![\mathcal{N}]\!] = \sum_{\substack{c \in \mathbb{N}_n \rightarrowtail \mathbb{N}_m \\ \text{iface}(\mathcal{P}_{c(i)}) = \text{iface}(h_i)}} C[\mathcal{P}_{c(1)}/h_1, \ldots, \mathcal{P}_{c(n)}/h_n]. \tag{4.31}$$

In particular, note that when $n = 0$, one simply get $C[\![\mathcal{N}]\!] = C$. Also, if $m = 0$ and consequently $\mathcal{N} = 0$, then $C[\![\mathcal{N}]\!] = 0$. Given $C[\ ]$ a non-simple resource context, and $\mathcal{N}$ a resource pre-net, $C[\![\mathcal{N}]\!]$ is the substitution of the latter in each addend of the former; i.e.: $0[\![\mathcal{N}]\!] = 0$ and $(C + C)[\![\mathcal{N}]\!] = C[\![\mathcal{N}]\!] + C[\![\mathcal{N}]\!]$.

Definition 4.7 (Parallel and expanded reduction). The closure of the simple reduction relation with respect to multi-hole resource contexts is called *parallel resource reduction* and written $\twoheadrightarrow$. Formally, given a resource net $\mathcal{N} = C[\![\mathcal{P}_1 + \ldots + \mathcal{P}_n]\!]$ where $C[\ ]$ is a multi-hole context and $\mathcal{P}_1, \ldots, \mathcal{P}_n$ are pre-nets, if $\mathcal{P}_1 \to \mathcal{P}_1, \ldots, \mathcal{P}_n \to \mathcal{P}_n$ then $\mathcal{N} \twoheadrightarrow C[\![\mathcal{P}_1 + \ldots + \mathcal{P}_n]\!]$. Given a mMELL proof-net N and a reduction step ρ on a cut $c \in V(N)$, the *expanded reduction of* ρ, written $\rho^\bullet$, is the parallel reduction of the set of any resource redex in $\text{Supp}(N^\bullet)$ for any cut in $\text{Supp}(c^\bullet)$.

Notation 4.1. We extend the domain of resource reduction for paths to sets of paths. For any reduction ρ, and any set of paths Π, we write $\rho(\Pi)$ to denote $\bigcup_{\pi \in \Pi} \rho(\pi)$.

Remark 4.3 (Indirectness of definition of expanded reduction). Even though expanded reduction has been defined using expansion itself but it would be interesting to have instead a direct formulation. Given $\mathcal{N}$ a sum of simple resource nets image of $N^\bullet$, how can one determine, with no knowledge of N, a sum of set of redexes $\mathbf{R}$ such that if $\mathcal{N} = N^\bullet$ then $\mathbf{R} = R^\bullet$? A terse answer has been developed by the author, but requires a restriction undesired by the aim of this thesis: it need to be formalised on proof structures that enjoys connectedness, such as weakening-free proof-nets or λ-terms. In this setting, it is not difficoult to exploit the property of 'uniformity' [Ehrhard and Regnier, 2008], that is the relation of syntactical similarity which intrinsically belongs to all elements within the expansion of the same term/net. In the latter (and more general) case of a weakening-free mMELL proof-net N, one can define a notion of locator of a vertex $v \in V(N)$ as its unique access path, i.e. the unique downward path that goes from the out-conclusion of a net (the root of the corresponding term) to v, without crossing the first premise of any $\multimap$-link. Then two cut are in the same expanded redex whenever their access paths cross the same sequence of kind of links, and for each $\multimap$-link of those they also cross the same vertex (i.e. the conclusion on the left or the first premise on the right).

4.4 Commutativity of reduction and expansion

In this section we show how the Taylor-Ehrhard-Regnier expansion commutes with reduction: reducing and then expanding a mMELL proof-net is equivalent to expanding and then reducing it. The same result is then easily reformulated on paths as well, which

implies that the persistence of a path is characterised by the existence of persistent paths in its expansion.

4.4.1 Commutativity on nets

Theorem 4.1 (Net expansion commutes with reduction step). *For any mMELL net N and any reduction ρ*

$$(\rho(N))^\bullet = \rho^\bullet(N^\bullet) \tag{4.32}$$

Proof. We obtain the equivalence of the two sides of Equation 4.32 by direct, even though technically delicate, manipulations.

1. Firstly we get rid of some irrelevant addends from the two sides of Equation 4.32.

   (a) Let $N = C[R]$, where R is the mMELL redex of ρ for the cut, say c, and C is its context with hole h. By definition of mMELL reduction, $\rho(N) = C[\rho(R)]$, so $(\rho(N))^\bullet = (C[\rho(R)])^\bullet$. Notice that within $C^\bullet$ there are addends which contain one or more copies of h and which are multi-hole simple resource contexts; and others addends where h has been erased, and which are simple resource nets. Let C be the sum of the former addends, and $\overline{N}$ the sum of the latter ones. Hence, $(\rho(N))^\bullet = C[(\rho(R))^\bullet] + \overline{N}$.

   (b) Let, $N^\bullet = C'[R'] + \overline{N}$, where R' is the sum, for any $c' \in c^\bullet$, of the resource redex of c', while C' is its multi-hole resource context, and $\overline{N}$ is again the sum of any non-zero addend in $N^\bullet$ which does not contain any $c' \in c^\bullet$. Hence, $\rho^\bullet(N^\bullet) = C'[\rho^\bullet(R')] + \overline{N}$.

   Therefore, we just reduced the thesis to the equivalence between $\mathrm{Supp}\,(C\,[(\rho(R))^\bullet])$ and $\mathrm{Supp}\,(C'\,[\rho^\bullet(R')])$.

2. We distinguish two cases, depending on the type of the redex of c, which, as per Fact 4.4, is equal to the type of the redex of any $c' \in c^\bullet$.

   (a) ($\multimap$ /$\bar{\multimap}$) Straightforward, since linear implication redexes are identical on both sides.

      i. By definition of expansion, we have that $(\rho(R))^\bullet$ contains exactly one element, which is equal to $\rho(R)$. Therefore, we obtain that $C\,[(\rho(R))^\bullet] = C\,[\rho(R)]$.

      ii. For the same reason above, R' is identical to R, which implies that $\rho^\bullet(R') = \rho(R)$, and also that $C = C'$. We immediately obtain $C'\,[\rho^\bullet(R')] = C\,[\rho(R)]$.

      Hence, $C\,[(\rho(R))^\bullet] = C'\,[\rho^\bullet(R')]$ which implies their respective supports are equal, quod erat demonstrandum. □

   (b) (!/?) This case is quite more involved from the previous one, because we have to deal with the difference between the two notions of redexes. In the mMELL redex we have an erasure or a duplication and a possible relocation of the argument box, while on the resource net, where all of this work has already been performed by the expansion, the redex simply consists of a resource dispatching. Such a complexity requires a deep analysis.

i. We first consider the right side of the equation we want to prove, i.e. $C[\![(\rho(R))^\bullet]\!]$. Suppose

$$R = \langle u_1, \ldots, u_k \ (?) \ c \rangle, \langle v \ (!) \ c \rangle, \langle W \ [\mathcal{B}] \ v \rangle,$$
$$\langle X_1 \ (?) \ y_1 \rangle, \ldots, \langle X_s \ (?) \ y_s \rangle. \tag{4.33}$$

Then, as by Definition 2.15 of mMELL proof-net reduction (in particular, see Figure 2.3b and Equation 2.5),

$$\rho(R) = \langle W_1 \ [\mathcal{B}_1] \ u_1 \rangle, \ldots, \langle W_k \ [\mathcal{B}_k] \ u_k \rangle,$$
$$\langle X'_1 \ (?) \ y_1 \rangle, \ldots, \langle X'_s \ (?) \ y_s \rangle, \tag{4.34}$$

where any X'_i is obtained by replacing any occurrence of a vertex $w \in W$ with the sequence $(w_1, \ldots, w_k)$ such that its elements respectively belongs to $W_1, \ldots, W_k$.

Now observe the reduct and observe that we can write it as follows.

$$\rho(R) = C''[\langle W_1 \ [\mathcal{B}_1] \ u_1 \rangle, \ldots, \langle W_k \ [\mathcal{B}_k] \ u_k \rangle] \tag{4.35}$$
$$C'' = \langle X'_1 \ (?) \ y_1 \rangle, \ldots, \langle X'_s \ (?) \ y_s \rangle,$$
$$\langle W_1 \cup \ldots \cup W_k \ (\square) \ u_1 \ldots u_k \rangle \tag{4.36}$$

where the $\square$-link denotes a hole h', whose in conclusions are written as premisses, and its out conclusions as conclusions. Now, let $C'''[\![ ]\!]$ be the multi-hole context obtained by splitting h' into k distinict holes $h_1, \ldots, h_k$ such that for any $1 \leq i \leq k$, we have $h_i = \langle W_i \ (\square) \ u_i \rangle$. Then, since the substitution of a net in a multi-hole context duplicates it, we can express $\rho(\mathcal{N})$ as $C'''[\![\mathcal{B}]\!]$. Therefore,

$$(\rho(\mathcal{N}))^\bullet = C'''[\![\mathcal{B}]\!]^\bullet.$$

Furthermore, the explicit presence of the box enable us to safely decompose expansion (cf. Definition 4.2), and write:

$$(\rho(\mathcal{N}))^\bullet = (C''')^\bullet[\![\mathcal{B}^\bullet]\!].$$

ii. Now let us analyse the left side of the equation of the claim, that is $C'[\![\rho^\bullet(R')]\!]$. First, we observe that there are addends of R' which reduce to 0, because the arities of the two exponential links differs. Therefore, let R'' be the restriction of R' obtained by depriving it of any $R' \in R'$ such that for some reduction $\rho' \in \rho^\bullet$, we have $\rho'(R') = 0$. But the removed addends are irrelevant, therefore $\rho^\bullet(R') = \rho^\bullet(R'')$, which means $C'[\![\rho^\bullet(R')]\!] = C'[\![\rho^\bullet(R'')]\!]$.

Now, by construction, any $R'' \in \mathrm{Supp}(R'')$ is necessarily made of two exponential links of equal arity $n_{R''}$. Notice that in $C'[\![R'']\!]$, by definition of multi-hole contexts substitution, R'' could be substituted only to a hole-link with equal interface, i.e. having $n_{R''}$ out-vertices and $n_{R''}$ in-vertices of corresponding types. Recall that !-link in R'' has been generated from the expansion of the promotion link of R, which implies that any hole-link compatible with R'' is connected to $n_{R''}$ sub-nets being addends of $\mathcal{B}^\bullet$.

Consider now the sum C'' made of any multi-hole simple resource context obtained by any $C' \in C'$ by removing, in correspondence of any

hole-link, all the copies of $\mathcal{B}$, and consequently extending such a hole. Hence,

$$C'[\![\mathbf{R}']\!] = C''\left[\!\!\left[\sum_{\mathsf{R}'' \in \mathrm{Supp}(\mathbf{R}'')}\left(\mathsf{R}'', \underbrace{\mathcal{B}^\bullet, \ldots, \mathcal{B}^\bullet}_{n_{\mathsf{R}''}}\right)\right]\!\!\right],$$

where $n_{\mathsf{R}''}$ is the arity of links in R''. By Definition 3.8 and 4.7, its reduction corresponds to the replacement of any $\mathsf{R}'' \in \mathbf{R}''$ with a permutation of their vertices. I.e.

$$C'[\![\rho^\bullet(\mathbf{R}')]\!] = C''\left[\!\!\left[\sum_{\substack{\mathsf{R}'' \in \mathrm{Supp}(\mathbf{R}'')\\n_{\mathsf{R}''} \in \mathfrak{S}_{n_{\mathsf{R}''}}}}\left([\sigma n_{\mathsf{R}''}], \underbrace{\mathcal{B}^\bullet, \ldots, \mathcal{B}^\bullet}_{n_{\mathsf{R}''}}\right)\right]\!\!\right],$$

where for any R'', if $u_1, \ldots, u_k$ are premises of the contraction link, and $v_1, \ldots, v_k$ are those of the co-contraction link, then $[\sigma n_{\mathsf{R}''}]$ denotes the vertex equation

$$\left[u'_k \equiv v'_{\sigma n_{\mathsf{R}''}(k)}, \ldots, u'_k \equiv v'_{\sigma n_{\mathsf{R}''}(k)}\right].$$

But the presence of such a permutation is redundant with respect to the support, because substitution in multi-hole contexts already spans over any possible permutation of its arguments. Hence,

$$\mathrm{Supp}\left(\rho^\bullet(\mathcal{N}^\bullet)\right) = \mathrm{Supp}\left(C''\left[\!\!\left[\sum_{\mathsf{R}'' \in \mathbf{R}''}\underbrace{\mathcal{B}^\bullet, \ldots, \mathcal{B}^\bullet}_{n_{\mathsf{R}''}}\right]\!\!\right]\right).$$

In the last expression we could further simplify the substituens by factorising the substituendum. Let C''' be the series of the contexts obtained from any non-zero addend of C'' by partitioning, similarly to what we did before, any hole-link having $n_{\mathsf{R}''}$ out-vertices into $n_{\mathsf{R}''}$ distinguished holes. Therefore, we can use a single expansion of $\mathcal{B}$, and obtain:

$$\mathrm{Supp}\left(\rho^\bullet(\mathcal{N}^\bullet)\right) = \mathrm{Supp}\left(C'''[\![\mathcal{B}^\bullet]\!]\right).$$

In the first step of the exponential case we showed $(\rho(\mathcal{N}))^\bullet = (C''')^\bullet[\![\mathcal{B}^\bullet]\!]$. In the second step we proved that $\mathrm{Supp}\left(\rho^\bullet(\mathcal{N}^\bullet)\right) = \mathrm{Supp}\left(C'''[\![\mathcal{B}^\bullet]\!]\right)$. But by construction $C''' = (C''')^\bullet$, so we conclude. □

∎

Remark 4.4 (Expansion and substitution). Expansion is hardly compatible with reduction. In order to prove Theorem 4.1, we could imagine structuring our reasoning with a sort of substitution lemma stating that "For any proof-net $C[\mathsf{R}]$, where R is a mMELL redex and C is a context, $\mathrm{Supp}\left((C[\mathsf{R}])^\bullet\right) = \mathrm{Supp}\left(C^\bullet[\![\mathsf{R}^\bullet]\!]\right)$". Unfortunately, mMELL reduction has two non-local behaviour in the dynamics of the exponential case, one of which contains an insurmountable obstacle for such approach. The light of the old syntax of Linear Logic allows us to discriminate the two. One non-local behaviour, probably the more evident one, lies in box duplication and erasure, i.e. the action of reducing, respectively, contraction and weakening. This poses no problem to expansion,

since the object box is treated as an atomic whole. The other non-local behaviour comes from box deepening and shallowing, i.e. the action of reducing, respectively, secondary doors and dereliction. This kind of reduction may involve other boxes, thus preventing the compositionality of desire. We consequently prefer to tackle the proof in its entirety.

By composition, we now straightforwardly generalise commutativity between expansion and reduction step, to reduction sequences: it suffices to iteratively apply Theorem 4.1.

Theorem 4.2 (Net expansion commutes with reduction sequence). *For any mMELL net $\mathcal{N}$ and any reduction sequence ρ*

$$(\rho(\mathcal{N}))^{\bullet} = \rho^{\bullet}(\mathcal{N}^{\bullet}) \tag{4.37}$$

Proof. Let $\rho = \rho_1 \ldots \rho_n$. We go by induction on the length n.

1. ($n = 0$). Trivial: the thesis shrinks to the identity $\mathcal{N}^{\bullet} = \mathcal{N}^{\bullet}$.

2. ($n > 0$). By inductive hypothesis:

$$\left(\rho_{n-1}(\ldots \rho_1(\mathcal{N})\ldots)\right)^{\bullet} = \rho_{n-1}^{\bullet}(\ldots \rho_1^{\bullet}(\mathcal{N}^{\bullet})\ldots).$$

Hence, we easily conclude applying Theorem 4.1:

$$\left(\rho_n\left(\rho_{n-1}(\ldots \rho_1(\mathcal{N})\ldots)\right)\right)^{\bullet} = \rho_n^{\bullet}\left(\rho_{n-1}^{\bullet}(\ldots \rho_1^{\bullet}(\mathcal{N}^{\bullet})\ldots)\right).$$

∎

Remark 4.5 (Expanded redexes and Lévy families). Expanded reduction appears to be deeply connected to the notion of parallel reduction on redex families discovered by Lévy [1980], which reduces in a single step all the redexes belonging to a given family, i.e. all the copies that preceding reduction created so far. Similarly, in the expansion of a net $\mathcal{N}$, any simple resource net $\mathcal{N}'$ which does not reduce to 0, has performed all the copies being necessary to its normalisation. For this reason, we believe that between Lévy-parallel reduction on $\mathcal{N}$ equipped with an appropriate strategy (a least efficient one in terms of number of copies), and expanded reduction on $\mathcal{N}'$, there lies a beautiful simulation. We conjecture, indeed, that for every step ρ of the former and any corresponding step of the latter, there is a natural bijection between the sets of simple redexes of the two.

Furthermore, we notice that the very notion of families is at the heart of the duality of the two concepts of main interest of the present thesis: sharing and expansion. They are somehow inverse operations on the use of arguments — the former considers any possible number of usages, the latter minimises their representation with respect to their contexts.

4.4.2   Commutativity on paths

Lastly presented commutation result holds, indeed not surprisingly, not only for reduction on nets, but also on paths. Because of such expectedness, we present an independent

proof that is of particular interest because it enlightens deep internals of the linearisation of Linear Logic exponential modality, explaining when, where and how path persistence can or cannot be preserved along expansion. To better enjoy the analysis, we shall structure it in three steps. Firstly, we show that commutation holds in the core case of a redex crossing and a single reduction step (Lemma 4.3); secondly, we enlarge our view to generic paths (Lemma 4.4); thirdly, we generalise the result to reduction sequences (Theorem 4.3). Lastly we derive the equivalence between mMELL persistence of a path and the existence of persistent path in its expansion (Theorem 4.4).

Lemma 4.3. *Let R be a mMELL redex, ρ be its reduction step, and $\chi \in \mathbb{P}(R)$. Then:*

$$\mathrm{Supp}\left((\rho(\chi))^{(\rho(R))^{\bullet}}\right) = \mathrm{Supp}\left(\rho^{\bullet}\left(\chi^{R^{\bullet}}\right)\right). \tag{4.38}$$

Proof. We distinguish two cases according to the type of redex.

1. $\mathcal{T}(R) = \multimap / \stackrel{-}{\multimap}$. Let $R = \langle v', v\ (\multimap)\ w \rangle, \langle u', u\ (\stackrel{-}{\multimap})\ w \rangle$ as in Figure 2.3a. We immediately notice that, by Definition 4.2, $R^{\bullet} = R$ while, as per Definition 4.5, we have $\chi^{R^{\bullet}} = \{\chi\}$. We consider two sub-cases depending on the persistence of χ to ρ.

   (a) $\rho(\chi) \neq \varnothing$. This implies that, by Definition 3.10 and in particular Equation 3.16, 3.17, that χ is either (v', w, u'), (v, w, u), or the reversal of these. Given that from the analysis of one case the other three can be straightforwardly obtained, let us assume $\chi = (v', w, u')$. By definition of reduction, we have $\rho(\chi) = \{(v')[v' \equiv u']\}$. Moreover, by definition of path expansion, we have that

   $$(\rho(\chi))^{(\rho(R))^{\bullet}} = \{(v')[v' \equiv u']\}^{(\rho(R))^{\bullet}} = \{(v')[v' \equiv u']\}.$$

   Let us now look at $\chi^{R^{\bullet}} = \{\chi\}$, and consider its expanded reduction: $\rho^{\bullet}\left(\chi^{R^{\bullet}}\right) = \rho^{\bullet}(\chi)$. Notice that $\rho^{\bullet}$ contains only one reduction step, which is identical to ρ, because $R^{\bullet} = R$. Hence we conclude:

   $$\rho^{\bullet}\left(\chi^{\mathcal{N}^{\bullet}}\right) = \rho(\chi) = \{(v')[v' \equiv u']\}.$$

   $\blacklozenge$

   (b) $\rho(\chi) = \varnothing$. Then we immediately notice that by definition of path expansion:

   $$(\rho(\chi))^{(\rho(R))^{\bullet}} = \varnothing^{(\rho(R))^{\bullet}} = \varnothing.$$

   Moreover, given that, as we remarked in previous case, we have $\chi^{R^{\bullet}} = \{\chi\}$, and $\rho^{\bullet} = \{\rho\}$, we immediately conclude:

   $$\rho^{\bullet}(\chi^{R^{\bullet}}) = \rho^{\bullet}(\chi) = \rho(\chi) = \varnothing.$$

   $\blacklozenge$

   $\square$

2. $\mathcal{T}(R) = {!}/{?}$. Let R be as in Figure 2.3b:

   $$R = \langle u_1, \ldots, u_k\ (?)\ c \rangle, \langle v\ (!)\ c \rangle, \langle W\ [\mathcal{B}]\ v \rangle, \langle X_1\ (?)\ y_1 \rangle, \ldots, \langle X_s\ (?)\ y_s \rangle. \tag{4.39}$$

   We distinguish three sub-cases depending on the polarities of the extrema vertices of the crossing of $\mathcal{B}$.

(a) out-out. Let $\chi = (u_j, c, v) :: \beta :: (v, c, u_{j'})$. Being the most interesting one, we shall discuss in full details this sub-case, further distinguishing two sub-sub-cases according to the persistence of χ.

i. χ persistent to ρ. Then, by definition of mMELL path reduction (in particular Equation 2.10), $j = j'$ and:

$$\rho(\chi) = \{(u_j) :: \beta_j :: (u_j) \, [u_j \equiv v_j]\},$$

where we kept trivial paths as extrema for the sake of clarity. By Definition 4.5:

$$(\rho(\chi))^{(\rho(R))^\bullet} = \left\{(u_j) :: \beta_j^{\mathcal{B}_j^\bullet} :: (u_j)\right\}$$

Now let us look at the rightmost side of Equation 4.38, and apply the definition of path expansion. If $\mathcal{B}$ is the h-th of the m outermost boxes of R,

$$(\chi)^{R^\bullet} = \sum_{a \in \mathbb{N}^m} \left\{(u_j, c, v_n) :: \beta_n^{\mathcal{B}_n^\bullet} :: (v_n, c, u_j) \,\middle|\, 1 \leq n \leq a(h)\right\},$$

so we can consider its expanded reduction w.r.t. ρ,

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{a \in \mathbb{N}^m} \left\{\rho^\bullet\left((u_j, c, v_n) :: \beta_n^{\mathcal{B}_n^\bullet} :: (v_n, c, u_j)\right) \,\middle|\, 1 \leq n \leq a(h)\right\}.$$

Notice that in any element of any addend of the sum, the two outermost paths are crossings of a resource redex of exponential type. The reduction step $\rho^\bullet$ performs by definition the reduction of any cut $c' \in C' \in \mathrm{Supp}(c^\bullet)$, including c. Therefore, we can apply Definition 3.11 of path reduction and in particular Equation 3.23.

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\substack{a \in \mathbb{N}^m \\ \sigma \in \mathfrak{S}_{a(h)}}} \left\{\rho^\sigma\left((u_j, c, v_n) :: \beta_n^{\mathcal{B}_n^\bullet} :: (v_n, c, u_j)\right) \,\middle|\, 1 \leq n \leq a(h)\right\}$$

$$= \sum_{\substack{a \in \mathbb{N}^m \\ \sigma \in \mathfrak{S}_{a(h)}}} \left\{\rho^\sigma\left((u_j, c, v_n)\right) :: \beta_n^{\mathcal{B}_n^\bullet} :: \rho^\sigma\left((v_n, c, u_j)\right) \,\middle|\, 1 \leq n \leq a(h)\right\}.$$

By definition (cf. Equation 3.21), ρ^σ maps to 0 whenever the arities mismatch, or when the permutation is not respected by all crossing; otherwise it reduces to a trivial path. Hence, recalling that k is the arity of the cut ?-link in R,

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\substack{a \in \mathbb{N}^m \\ \sigma \in \mathfrak{S}_{a(h)}}} \left\{(u_j) :: \beta_n^{\mathcal{B}_n^\bullet} :: (u_j) \, [u_j \equiv v_{\sigma(j)}] \,\middle|\, 1 \leq n \leq a(h), \, k = a(h), \, \sigma(j) = n\right\}.$$

Because of the condition on $a(h)$, we can immediately simplify as:

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\sigma \in \mathfrak{S}_k} \left\{(u_j) :: \beta_n^{\mathcal{B}_n^\bullet} :: (u_j) \, [u_j \equiv v_{\sigma(j)}] \,\middle|\, 1 \leq n \leq k, \, \sigma(j) = n\right\}.$$

Observe that for any $\sigma \in \mathfrak{S}_k$ there exists a unique $1 \leq n \leq k$ such that $\sigma(j) = n$. Therefore, every addend is a singleton of a path:

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\sigma \in \mathfrak{S}_k} \left\{(u_j) :: \beta_{\sigma(j)}^{\mathcal{B}_{\sigma(j)}^\bullet} :: (u_j) \, [u_j \equiv v_{\sigma(j)}]\right\}.$$

Now observe that, for any $\{\chi'\}, \{\chi''\} \in \text{Supp}\left(\rho^\bullet\left(\chi^{R^\bullet}\right)\right)$, we have $\chi' = \chi''$, since they are copies of the same path, within copies of the same sub-substructure. In particular, we have $|\mathfrak{S}_k| = k!$ equal addends, which can be simplified in a single one, since the sum is idempotent, and conclude:

$$\text{Supp}\left(\rho^\bullet\left(\chi^{R^\bullet}\right)\right) = \left\{(u_j) :: \beta_j^{\mathcal{B}_j^\bullet} :: (u_j) \; [u_j \equiv v_j]\right\}$$
$$= \text{Supp}\left((\rho(\chi))^{(\rho(R))^\bullet}\right).$$

$\diamond$

ii. χ not persistent to ρ. Then, as per (2.11), $j \neq j'$ and $\rho(\chi) = \varnothing$. Then, immediately by (4.24): $(\rho(\chi))^{(\rho(R))^\bullet} = 0$. The analysis of $\rho^\bullet\left(\chi^{R^\bullet}\right)$ goes as we previously did, until we reach

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\substack{a \in \mathbb{N}^m \\ \sigma \in \mathfrak{S}_{a(h)}}} \left\{\rho^\sigma\left((u_j, c, v_n)\right) :: \beta_n^{\mathcal{B}_n^\bullet} :: \rho^\sigma\left((v_n, c, u_j')\right) \;\middle|\; 1 \leq n \leq a(h)\right\},$$

where we notice that, since $j \neq j'$, there exists no $\sigma \in \mathfrak{S}_{a(h)}$ such that $\sigma(j) = n = \sigma(j')$. Therefore, by Equation 3.21, in every element of any set of the summation that we are now considering, we have that $\rho^\sigma\left((u_j, c, v_n)\right) = 0$ or $\rho^\sigma\left((v_n, c, u_j)\right) = 0$. Hence, $\rho^\bullet\left(\chi^{R^\bullet}\right) = 0$. $\diamond$

♦

(b) out-in. We follow the same reasoning used in previous sub-case. Let $1 \leq j \leq k$ and $1 \leq l \leq s$, and then let $\chi = (u_j, c, v) :: \beta_i :: (w_l, y_l)$. Then:

$$\rho(\chi) = \left\{(u_j) :: \beta_j :: (w_{l_j}, y_l) \; [u_j \equiv v_j]\right\}.$$

Hence,

$$(\rho(\chi))^{(\rho(R))^\bullet} = \left\{(u_j) :: \beta_j^{\mathcal{B}_j^\bullet} :: (w_{l_j}, y_l)\right\},$$

where $\beta_j^{\mathcal{B}_j^\bullet}$ denotes the expansion of β_j, i.e. the j-th copy of the path β, with respect $\mathcal{B}_j$ i.e. the j-th copy of the box $\mathcal{B}$ containing β. On the other side of Equation 4.38,

$$\chi^{R^\bullet} = \sum_{a \in \mathbb{N}^m} \left\{(u_j, c, v_n) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{l_j}, y_l) \;\middle|\; 1 \leq n \leq a(h)\right\}.$$

In this case, expanded reduction trivially deals with one cut per crossing of the expansion. Therefore simply applying the definition of reduction, and then simplifying by sum's idempotence the k! equal singletons in the summation, we obtain:

$$\rho^\bullet\left(\chi^{R^\bullet}\right) = \sum_{\sigma \in \mathfrak{S}_k} \left\{(u_j) :: \beta_{\sigma(j)}^{\mathcal{B}_{\sigma(j)}^\bullet} :: (w_{l_j}, y_l) \; [u_j \equiv v_{\sigma(j)}]\right\}$$
$$= \left\{(u_j) :: \beta_j^{\mathcal{B}_j^\bullet} :: (w_{l_j}, y_l) \; [u_j \equiv v_j]\right\},$$

whose support is equal to $\text{Supp}\left((\rho(\chi))^{(\rho(R))^\bullet}\right)$, as required. ♦

(c) in-in. Again, along the same line of reasoning, let $1 \leq l, m \leq s$, and let $\chi = (y_l, w_l) :: \beta :: (w_m, y_m)$. This time, differently from previous cases 2a

and 2b, the set that is the reduct of χ may be empty or contain more than one path (cf. Equation 2.13).

$$\rho(\chi) = \left\{ (y_l, w_{l_j}) :: \beta_j :: (w_{m_j}, y_m) \mid 1 \le j \le k \right\}. \qquad (4.40)$$

Hence, its expansion is:

$$(\rho(\chi))^{(\rho(R))^\bullet} = \left\{ (y_l, w_{l_j}) :: \beta_j^{\mathcal{B}_j^\bullet} :: (w_{m_j}, y_m) \,\middle|\, 1 \le j \le k \right\}.$$

On the right side of Equation 4.38, if $\mathcal{B}$ is the h-th of the b outermost boxes of R, we find that

$$(\chi)^{R^\bullet} = \sum_{a \in \mathbb{N}^b} \left\{ (y_l, w_{l_n}) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{m_n}, y_m) \,\middle|\, 1 \le n \le a(h) \right\}.$$

Now, consider its expanded reduction as we previously did:

$$\rho^\bullet \left(\chi^{R^\bullet} \right) = \sum_{\substack{a \in \mathbb{N}^b \\ \sigma \in \mathfrak{S}_{a(h)}}} \left\{ \rho^\sigma \left((y_l, w_{l_n}) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{m_n}, y_m) \right) \,\middle|\, 1 \le n \le a(h) \right\}.$$

Let us consider the persistence w.r.t. $\rho^\bullet$ of a path $\chi' \in X'$, for some addend X' of the sum. By definition, given a redex R' of some cut $c' \in c^\bullet$, χ' is persistent if: $a(h) = k'$, where k' is the arity of the ?-link cut in R' and is equal to k, the arity of the ?-link cut in R; every crossing of R' in χ' preserves $\sigma \in \mathfrak{S}_{a(h)}$. Now, χ' contains no such crossing, so the second condition is always (vacuously) satisfied. Hence, we can simplify by imposing $a(h) = k$ and applying the definition of exponential reduction.

$$= \sum_{\sigma \in \mathfrak{S}_k} \left\{ \rho^\sigma \left((y_l, w_{l_n}) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{m_n}, y_m) \right) \,\middle|\, 1 \le n \le k \right\}$$

$$= \sum_{\sigma \in \mathfrak{S}_k} \left\{ (y_l, w_{l_n}) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{m_n}, y_m) \,\middle|\, 1 \le n \le k \right\}$$

Then, once again, k! equal addends are neutralised by sum's idempotence:

$$= \left\{ (y_l, w_{l_n}) :: \beta_n^{\mathcal{B}_n^\bullet} :: (w_{m_n}, y_m) \,\middle|\, 1 \le n \le k \right\},$$

and we observe that its support is equal to $\mathrm{Supp}\left((\rho(\chi))^{(\rho(R))^\bullet} \right)$. ♦

□

■

Lemma 4.4 (Path expansion commutes with reduction step). *Let N be a mMELL proof-net, let ρ be a reduction step on N, and $\pi \in P_E(N)$. Then*

$$\mathrm{Supp}\left((\rho(\pi))^{(\rho(N))^\bullet} \right) = \mathrm{Supp}\left(\rho^\bullet(\pi^{N^\bullet}) \right). \qquad (4.41)$$

Proof. Let R be the redex of ρ, so that $N = C[R]$. Being an execution path, π is necessarily long enough for R, so let

$$\mathrm{RCF}_R(\pi) = \gamma_0 :: \chi_1 :: \gamma_1 :: \ldots :: \chi_k :: \gamma_k \qquad (4.42)$$

and let γ be a path in C such that

$$\gamma = \gamma_0 :: \theta_1 :: \gamma_1 :: \ldots :: \theta_k :: \gamma_k \tag{4.43}$$

where θ_i for any $1 \le i \le k$ is an atomic path crossing its hole-link. Now let us analyse path expansion. By definition, we can express $\pi^{\mathcal{N}^\bullet}$ as the appropriate substitution of any χ'_i appearing in $\chi_i^{R^\bullet}$ of the corresponding θ'_i appearing in $\gamma^{C^\bullet}$:

$$\pi^{\mathcal{N}^\bullet} = \sum_{\substack{\Gamma \in \mathrm{Supp}(\gamma^{C^\bullet}) \\ \chi'_i \in \mathrm{Supp}(\chi_i^{R^\bullet})}} \{ \gamma'_0 :: \chi'_1 :: \gamma'_1 :: \ldots :: \chi'_k :: \gamma'_k \mid \gamma'_0 :: \theta'_1 :: \gamma'_1 :: \ldots :: \theta'_k :: \gamma'_k \in \Gamma, \ \chi'_i \in X'_i \} \tag{4.44}$$

Now, reduction of mMELL paths (Definition 2.20) acts on redex crossing:

$$\rho(\pi) = \gamma_0 :: \rho(\chi_1) :: \gamma_1 :: \ldots :: \rho(\chi_k) :: \gamma_k,$$

therefore, similarly to what we have done in Equation 4.44, we can write the expansion of $\rho(\pi)$ as the expansion of γ where we substitute any sub-path θ'_i (that is an expansion of θ) with any expansion of $\rho(\chi_i)$.

$$(\rho(\pi))^{(\rho(\mathcal{N}))^\bullet} = \sum_{\substack{\Gamma \in \mathrm{Supp}(\gamma^{C^\bullet}) \\ \chi'_i \in \mathrm{Supp}(\rho(\chi_i)^{(\rho(R))^\bullet})}} \{ \gamma'_0 :: \chi'_1 :: \gamma'_1 :: \ldots :: \chi'_k :: \gamma'_k \mid \gamma'_0 :: \theta'_1 :: \gamma'_1 :: \ldots :: \theta'_k :: \gamma'_k \in \Gamma, \ \chi'_i \in X'_i \} \tag{4.45}$$

Now let us consider the action of expanded reduction on Equation 4.44. For any set of resource redexes R' in the support of the expansion of R, let $\rho^{R'} \subseteq \rho^\bullet$ denote the set of ordinary resource reduction steps acting on every redex in R'. We distinguish two cases according to the type of redex.

1. $\mathcal{T}(R) = \multimap/\overline{\multimap}$. We can directly apply Definition 3.10.

$$\rho^\bullet\left(\pi^{\mathcal{N}^\bullet}\right) = \sum_{\substack{\Gamma \in \mathrm{Supp}(\gamma^{C^\bullet}) \\ R' \in \mathrm{Supp}(R^\bullet) \\ \chi''_i \in \mathrm{Supp}(\rho^{R'}(\chi_i^{R'^\bullet}))}} \{ \gamma'_0 :: \chi''_1 :: \gamma'_1 :: \ldots :: \chi''_k :: \gamma'_k \mid \gamma'_0 :: \theta'_1 :: \gamma''_1 :: \ldots :: \theta'_k :: \gamma''_k \in \Gamma, \ \chi''_i \in X''_i \} \tag{4.46}$$

Therefore, from Equation 4.45 and 4.46, we deduce that it is enough to prove that

$$\mathrm{Supp}\left(\rho(\chi_i)^{(\rho(R))^\bullet}\right) \overset{?}{=} \mathrm{Supp}\left(\sum_{R' \in \mathrm{Supp}(R^\bullet)} \rho^{R'}\left(\chi_i^{R'^\bullet}\right)\right),$$

which is equivalent to the very statement of Lemma 4.3:

$$\mathrm{Supp}\left(\rho(\chi_i)^{(\rho(R))^\bullet}\right) = \mathrm{Supp}\left(\rho^\bullet\left(\chi_i^{R^\bullet}\right)\right).$$

$\square$

2. $\mathcal{T}(R) = !/?$. Similarly to the previous case, we now apply Definition 3.11 of path reduction. This time the sum we obtain spans also over the resource permutation:

$$\rho^\bullet\left(\pi^{\mathcal{N}^\bullet}\right) = \sum_{\substack{\Gamma \in \mathrm{Supp}(\gamma^{C^\bullet}) \\ R' \in \mathrm{Supp}(R^\bullet) \\ \sigma \in \mathfrak{S}_{R'} \\ \chi''_i \in \mathrm{Supp}(\rho^{R'\sigma}(\chi_i^{R'^\bullet}))}} \{ \gamma'_0 :: \chi''_1 :: \gamma'_1 :: \ldots :: \chi''_k :: \gamma'_k \mid \gamma'_0 :: \theta'_1 :: \gamma''_1 :: \ldots :: \theta'_k :: \gamma''_k \in \Gamma, \ \chi''_i \in X''_i \} \tag{4.47}$$

Comparing Equation 4.45 and 4.47, we reduce the statement to:

$$\mathrm{Supp}\left(\rho(\chi_i)^{(\rho(R))^\bullet}\right) \overset{?}{=} \mathrm{Supp}\left(\sum_{\substack{R'\in\mathrm{Supp}(R^\bullet)\\ \sigma\in\mathfrak{S}_{R'}}} \rho^{R'^\sigma}\left(\chi_i^{R'^\bullet}\right)\right),$$

which is ensured again by Lemma 4.3:

$$\mathrm{Supp}\left(\rho(\chi_i)^{(\rho(R))^\bullet}\right) = \mathrm{Supp}\left(\rho^\bullet\left(\chi_i^{R^\bullet}\right)\right).$$

$\square$

$\blacksquare$

Theorem 4.3 (Path expansion commutes with reduction sequence)**.** *Let N be a mMELL proof-net, let ρ be a reduction sequence on N, and $\pi \in P_E(N)$. Then:*

$$\mathrm{Supp}\left((\rho(\pi))^{(\rho(N))^\bullet}\right) = \mathrm{Supp}\left(\rho^\bullet(\pi^{N^\bullet})\right). \tag{4.48}$$

Proof. Using Lemma 4.4, the proof reduces to a simple induction on the number of reduction steps in $\bar\rho$.

1. ($|\rho| = 0$). Trivial: $(\rho(\pi))^{(\rho(N))^\bullet} = \pi^{N^\bullet} = \mathrm{Supp}\left(\rho^\bullet(\pi^{N^\bullet})\right)$.

2. ($|\rho| > 0$). Let $\rho = \rho'\rho''$, where ρ' is a single step. By inductive hypothesis:

$$\mathrm{Supp}\left((\rho''(\pi))^{(\rho''(N))^\bullet}\right) = \mathrm{Supp}\left((\rho'')^\bullet(\pi^{N^\bullet})\right).$$

Hence, we immediately conclude applying Lemma 4.4:

$$\mathrm{Supp}\left((\rho'(\rho''(\pi)))^{(\rho'(\rho''(N)))^\bullet}\right) = \mathrm{Supp}\left((\rho')^\bullet((\rho'')^\bullet(\pi^{N^\bullet}))\right).$$

$\blacksquare$

Theorem 4.4. *For any mMELL proof-net N, and $\pi \in P_E(N)$, π is persistent if and only if there exists $\pi' \in \Pi \in \mathrm{Supp}\left(\pi^{N^\bullet}\right)$ which is persistent.*

Proof. We separately prove necessity and sufficiency.

1. ($\Leftarrow$). By contraposition, suppose π non persistent. This means that there exists a mMELL reduction sequence $\bar\sigma$ such that $\bar\sigma(\pi) = \varnothing$. Hence, $(\bar\sigma(\pi))^{(\bar\sigma(N))^\bullet} = \varnothing$. Therefore, by commutativity of Theorem 4.3, also $\bar\sigma^\bullet(\pi^{N^\bullet}) = \varnothing$. This allows us to conclude — for any $\Pi \in \mathrm{Supp}\left(\pi^{N^\bullet}\right)$ and for every $\pi' \in \Pi$, there exists some resource reduction sequence $\bar\sigma' \subseteq \bar\sigma^\bullet$ such that $\bar\sigma'(\pi') = \varnothing$. $\square$

2. ($\Rightarrow$). Let $\bar\sigma$ be a reduction sequence $N \rightarrow^* \mathrm{NF}(N)$. By hypothesis $\bar\sigma(\pi) \neq \varnothing$, therefore $(\bar\sigma(\pi))^{\mathrm{NF}(N)^\bullet} \neq \varnothing$. By commutativity of expansion with reduction (Theorem 4.3), $\mathrm{Supp}\left((\bar\sigma(\pi))^{\mathrm{NF}(N)^\bullet}\right) = \mathrm{Supp}\left(\bar\sigma^\bullet\left(\pi^{N^\bullet}\right)\right)$, which implies that $\bar\sigma^\bullet\left(\pi^{N^\bullet}\right) \neq \varnothing$. Hence, there exist $\Pi \in \mathrm{Supp}\left(\pi^{N^\bullet}\right)$ and $\pi' \in \Pi$ such that $\bar\sigma^\bullet(\pi') \neq \varnothing$. Now, since $\bar\sigma(N)$ is cut-free, by Fact 4.2 it must be the case that also $\bar\sigma^\bullet(N^\bullet)$ is cut-free, i.e. it is in normal form with respect to resource reduction. Therefore, we conclude by Lemma 3.5 that π' is persistent. $\square$

Remark 4.6 (Untyped future directions). Among results of present Section 4.3, the only point where we used the fact that nets are typed is in the proof of Theorem 4.4, which benefited from the consequent normalisation property. We believe that a generalisation to full λ-calculus is possible using more sophisticated rewriting techniques, allowing obtaining useful properties such as infinitary confluence. Thanks to that, the ($\Rightarrow$) part of the proof could be proved by contraposition: assuming that any $\pi' \in \pi^\bullet$ is not persistent, and supposing $\sigma'_{\pi'}(\pi') = 0$, one can build a sequence of expanded reduction $\sigma^\bullet$ that morally is the completion with respect to expanded reduction of the union of any $\sigma_{\pi'}$. Now $\sigma^\bullet$ would be such that $\sigma^\bullet(\pi^\bullet) = 0$, implying that π not persistent.

4.5 Expansion and execution

Theorem 4.4 showed that persistence of a mMELL execution path π can be characterised by the existence of persistent resource paths in its expansion. But this, in turn, can be characterised by regularity, as established in Theorem 3.4. So we can formulate an expanded variant of the notion of regularity, which characterise persistence of π using weights of its expansion. Moreover, we can formulate a qualitative and expanded variant of the execution formula for a $\star$-typed mMELL proof-net $\mathcal{N}$. We weigh, within the $\mathsf{r}\mathfrak{L}^*$ monoid, any path in any set of the sum obtained by expanding any path in $\mathcal{N}$. Such expansion-execution is invariant with respect to mMELL reduction.

Definition 4.8 (Expansion-regularity and execution). The *expanded weight* of a mMELL path π in a mMELL proof-net $\mathcal{N}$ is the sum of the $\mathsf{r}\mathfrak{L}^*$ weights any of its comprehensive expansions:

$$\mathfrak{w}^\bullet(\pi) = \sum_{\substack{\pi' \in \Pi \in \mathrm{Supp}(\pi^{\mathcal{N}^\bullet}) \\ \pi' \mathrm{compr.}}} \mathfrak{w}(\pi'). \tag{4.49}$$

We call π *expansion-regular* if $\mathfrak{w}^\bullet(\pi) \neq 0$. The *expansion-execution* of $\mathcal{N}$, written $\mathfrak{Ex}^\bullet(\mathcal{N})$, is the sum of the expanded weights of any execution path in $\mathcal{N}$:

$$\mathfrak{Ex}^\bullet(\mathcal{N}) = \sum_{\pi \in \mathcal{P}_E(\mathcal{N})} \mathfrak{w}^\bullet(\pi). \tag{4.50}$$

Last definition is not vacuous, since a comprehensive path always inhabit the expansion of a path, as clarified by the following fact, and could also be located by constructive reasoning.

Fact 4.5 (Comprehensive expansions). *For any persistent path π in a closed mMELL net $\mathcal{N} : \star$, there exists a comprehensive path in $\pi^{\mathcal{N}^\bullet}$.*

Proof. Immediate. By Theorem 4.4 there exists $\pi' \in \Pi \in \mathrm{Supp}(\pi^{\mathcal{N}^\bullet})$ that is persistent and which, as per Lemma 3.3, is necessarily comprehensive. ∎

Remark 4.7 (Construction of a comprehensive expansion of a persistent path). We proved the existence of a comprehensive path within $\mathrm{Supp}(\pi^{\mathcal{N}^\bullet})$ by an indirect and non-constructive argumentation. Nevertheless, it is not hard to sketch an algorithmic

alternative construction by shrinking any redundant expansion of $\mathcal{N}$.

Notice first that $\mathcal{N}^\bullet$ necessarily contains a path π' that is comprehensive. Indeed, for any $\mathcal{N}'' \in \mathcal{N}^\bullet$ and for any $\pi'' \in \pi^{\mathcal{N}''}$, if π'' is not comprehensive then we can find another net $\mathcal{N}'$ where a path π', copy of π'', is comprehensive. Suppose that $\mathcal{B} \in \mathrm{Bxs}\,(\mathcal{N})$ is such that one of its expansions in $\mathcal{N}''$ is not crossed by π''. Then let $\mathrm{Pr}(\mathcal{N}'')$ be the simple resource net obtained from $\mathcal{N}''$ by pruning that missed copy of $\mathcal{B}$. Clearly, there exists a path $\mathrm{Pr}(\pi'') \in \mathbb{P}(\mathrm{Pr}(\mathcal{N}''))$ which is identical to π'' but it lives in a "smaller" net. There evidently exists a fixed point $\mathcal{N}' = \mathrm{Pr}(\ldots \mathrm{Pr}(\mathcal{N}''))$, where $\pi' = \mathrm{Pr}(\ldots \mathrm{Pr}(\pi''))$ crosses each expansion of each $\mathcal{B} \in \mathrm{Bxs}\,(\mathcal{N})$, and it is thus comprehensive.

As expected, the notion of expanded-regularity is equivalent to path persistence in $\star$-typed mMELL proof-nets.

Corollary 4.1. *A path π is persistent if and only if π is expansion-regular.*

Proof. We separately prove necessity and sufficiency. Let $\mathcal{N}$ be the proof-net of π.

1. ($\Leftarrow$). By contraposition, suppose π not persistent. Then, by Theorem 4.4, we have that any $\pi' \in \Pi \in \mathrm{Supp}\,(\pi^{\mathcal{N}^\bullet})$ is not persistent. In particular, it is the case when π' is comprehensive, but Theorem 3.4 established that any such π' is not regular, which means by definition that $\mathbf{w}\,(\pi') = 0$. Ergo, π is not expansion-regular. □

2. ($\Rightarrow$). By contraposition, suppose π not expansion-regular, which by definition means that $\sum_{\pi' \text{compr.} \in \Pi \in \mathrm{Supp}(\pi^{\mathcal{N}^\bullet})} \mathbf{w}\,(\pi') = 0$. Now, since in $(\mathbf{r}\mathcal{L}^*, +)$ the only invertible element is the identity (for any $a, b \in \mathbf{r}\mathcal{L}^*$, if $a + b = 0$ then $a = b = 0$), it must be the case that $\mathbf{w}\,(\pi') = 0$ for any π'. Ergo, by Theorem 3.4, any comprehensive $\pi' \in \pi^{\mathcal{N}^\bullet}$ is not persistent. Now, recall that, by Lemma 3.3, every execution path that is not comprehensive cannot be persistent. Therefore, there exists no $\pi' \in \pi^{\mathcal{N}^\bullet}$ being persistent. Hence, by Theorem 4.4, we conclude that π is not persistent as well. □

■

More generally, the new expanded execution formula is invariant under mMELL reduction for $\star$-typed proof-nets.

Theorem 4.5. *For any mMELL proof-net $\mathcal{N}$ and reduction step ρ*

$$\mathbf{Ex}^\bullet(\mathcal{N}) =_{\mathbf{r}\mathcal{L}^*} \mathbf{Ex}^\bullet(\rho(\mathcal{N})). \tag{4.51}$$

Proof. By definition, we need to show that $\sum_{\pi \in \mathbb{P}_E(\mathcal{N})} \mathbf{w}^\bullet(\pi) = \sum_{\phi \in \mathbb{P}_E(\rho(\mathcal{N}))} \mathbf{w}^\bullet(\phi)$. Since by Lemma 2.2, the step ρ induces a partition on $\mathbb{P}_E(\rho(\mathcal{N}))$, we may express the claim as the equality between

$$\sum_{\pi \in \mathbb{P}_E(\mathcal{N})} \mathbf{w}^\bullet(\pi) \qquad \text{and} \qquad \sum_{\substack{\pi \in \mathbb{P}_E(\mathcal{N}) \\ \phi \in \rho(\pi) \subseteq \mathbb{P}_E(\rho(\mathcal{N}))}} \mathbf{w}^\bullet(\phi).$$

Hence, for any $\pi \in \mathbb{P}_E(\mathcal{N})$, we shall prove that the followings are equal:

$$\mathbf{w}^\bullet(\pi) \qquad \text{and} \qquad \sum_{\phi \in \rho(\pi)} \mathbf{w}^\bullet(\phi).$$

By definition, they are the sum of their comprehensive expansions:

$$\sum_{\substack{\pi'\in\Pi\in\mathrm{Supp}\left(\pi^{\mathcal{N}^{\bullet}}\right)\\ \pi'\mathrm{compr.}}} \mathfrak{w}\left(\pi'\right) \qquad \text{and} \qquad \sum_{\substack{\phi\in\rho(\pi)\\ \phi'\in\Phi\in\mathrm{Supp}\left(\phi^{(\rho(\mathcal{N}))^{\bullet}}\right)\\ \phi'\mathrm{compr.}}} \mathfrak{w}\left(\phi'\right).$$

Exploiting again Lemma 2.2, we can highlight the expansion of $\rho(\pi)$ within the right-most summation, and obtain:

$$\sum_{\substack{\pi'\in\Pi'\in\mathrm{Supp}\left(\pi^{\mathcal{N}^{\bullet}}\right)\\ \pi'\mathrm{compr.}}} \mathfrak{w}\left(\pi'\right) \qquad \text{and} \qquad \sum_{\substack{\pi''\in\Pi''\in\mathrm{Supp}\left((\rho(\pi))^{(\rho(\mathcal{N}))^{\bullet}}\right)\\ \pi''\mathrm{compr.}}} \mathfrak{w}\left(\pi''\right).$$

On the leftmost summation, by invariance theorem of $\mathsf{r}\mathcal{L}^{*}$ (Theorem 3.5), for any reduction step ρ', we have that $\mathfrak{w}\left(\pi'\right) =_{\mathsf{r}\mathcal{L}^{*}} \mathfrak{w}\left(\rho'(\pi')\right)$. Consequently, iterating this argument on any step of $\rho^{\bullet}$, we conclude that $\mathfrak{w}\left(\pi'\right) =_{\mathsf{r}\mathcal{L}^{*}} \mathfrak{w}\left(\rho^{\bullet}(\pi')\right)$. Hence, we can equivalently compare:

$$\sum_{\substack{\pi'\in\Pi\in\mathrm{Supp}\left(\pi^{\mathcal{N}^{\bullet}}\right)\\ \pi'\mathrm{compr.}}} \mathfrak{w}\left(\rho^{\bullet}(\pi')\right) \qquad \text{and} \qquad \sum_{\substack{\pi''\in\Pi\in\mathrm{Supp}\left((\rho(\pi))^{(\rho(\mathcal{N}))^{\bullet}}\right)\\ \pi''\mathrm{compr.}}} \mathfrak{w}\left(\pi''\right).$$

On the rightmost summation, we know from commutativity Lemma 4.4 that $(\rho(\pi))^{(\rho(\mathcal{N}))^{\bullet}}$ and $\rho^{\bullet}\left(\pi^{\mathcal{N}^{\bullet}}\right)$ have the same support. Thus, we obtain:

$$\sum_{\substack{\pi'\in\Pi'\in\mathrm{Supp}\left(\pi^{\mathcal{N}^{\bullet}}\right)\\ \pi'\mathrm{compr.}}} \mathfrak{w}\left(\rho^{\bullet}(\pi')\right) \qquad \text{and} \qquad \sum_{\substack{\pi''\in\Pi''\in\mathrm{Supp}\left(\rho^{\bullet}(\pi^{\mathcal{N}^{\bullet}})\right)\\ \pi''\mathrm{compr.}}} \mathfrak{w}\left(\pi''\right).$$

Recall that $\rho^{\bullet}$ is equivalent to a reduction sequence $\bar{\sigma}$ which includes a step ρ' redex of $\rho^{\bullet}$. Now for any such ρ', we know from Theorem 3.1 that $\rho^{\bullet}$ induces a bijection between execution paths in a RIN and its reduct, provided they persist to ρ'. Moreover, as stated in Fact 3.1, comprehensiveness is preserved by reduction. Thus, $\bar{\rho}$, hence $\rho^{\bullet}$, induces a bijection between execution comprehensive paths $\pi^{\mathcal{N}^{\bullet}}$ and those of $\rho^{\bullet}\left(\pi^{\mathcal{N}^{\bullet}}\right)$. Therefore we can write the rightmost summation identically to the leftmost one. ∎

4.6 Discussion

4.6.1 Related works

In an inspiring paper [Ehrhard and Regnier, 2006b], Taylor expansion was already shown to commute with head-reduction normalisation. In particular, the authors proved that for any ordinary term t, the set of resource terms obtained by expanding the Böhm tree of t is equal to the set of any non-zero normal form of the expansion of t. Their proof is based on a operational view on the correspondence with a modified Krivine's machine Krivine [2007]. For simply typed terms, the aforementioned Theorem 4.3 of this paper generalises such commutation property to any reduction sequence and to any choice of strategy, and is more directly formulated and proved on paths.

4.6.2 Open questions

4.6.2.1 Infinite paths

Is there a nicely-expressible and general notion of infinite paths that allows to capture infinite normal forms? The inspiration may come from Böhm trees and its notion of meaningful infinite head normal form, and such a reduction strategy is already known to be closely related both to dynamics of expansion Ehrhard and Regnier [2006b] and of paths Laurent [2001].

4.6.2.2 Resource abstract machine for the lambda-calculus

Can we design a non-deterministic linear abstract machine inspired by the construction presented here? In order to do so, we need to represent the expansion in a local manner, even if it has a twofold global nature: one in the argument superposition of RC, the other in the superposition of differently expanded terms. Removing the latter, indeed subsumed by the former in a certain sense, the challenge becomes that of formalising a notion of expanded execution which does not explicitly consider the expansion of the proof-net, but which instead first assigns algebraic weights locally, and then computes them in a distributed way. Starting from the approach presented here, the question may be addressed by formalising expansion as a local graph-rewriting system.

4.6.2.3 Combinatorics of path expansion

What are the multiplicity coefficients for path expansion? A study of the non-trivial combinatorial properties would complete our results, providing a tool to study the combinatorics of ordinary paths in λ-terms. We believe it to be connected to the normalisation complexity and the expansion-related part of the framework to be fruitfully extendible at a quantitative level.

Part II

Sharing and efficiency

Chapter 5

Sharing implementation of bounded logics

Contents

5.1 Introduction

Sharing graphs are a fine and versatile implementation technique for graph rewriting systems, such as Λ and MELL. The sharing graphs machinery can be conceptually partitioned into five components, each of which realizes a different type of operation.

Linear component implements the linear fragment of the target calculus, such as the multiplicatives and their cut for MELL, or a linear variant of the β that postpone the duplication of the argument.

Duplication and sharing is the core component, also called the abstract algorithm, which needs a new kind of graph link, called mux, that essentially implements the duplication in a distributed and atomic way. The mux propagation operation corresponds indeed to a sort of commutation of a contraction link over links contained inside another exponential box, or to a step of partial duplication of an argument inside its context. Whilst the contraction link, or the λ binder construct, have only one possible orientation — one input with several outputs — muxes allow also its dual — one output with several inputs, which means that muxes share not only terms, but also contexts.

Indexing management implements the exponential boxes information in a local and asynchronous way. It is also called the oracle, or the bookkeeper, and it can be implemented by mean of explicit hyperlinks (the bracket and the croissant), or by muxes themselves, when all hyperlinks are labelled with indices of their box-nesting level. It corresponds to a sort of commutation of a dereliction link over links contained inside another exponential box.

Readback implements an unrestricted form of duplication in order to remove any sharing in a graph and to obtain an ordinary proof-net or a plain lambda tree. From a practical point of view, it is meant to be used at the end of the main reduction or normalization. To do so, it interacts with other links also when they are not connected through a cut, i.e. with their conclusion.

Garbage collection implements erasure locally, link by link, and it is either represented by zero-ary muxes, or kept distinct by the usage of two new hyperlinks. It corresponds to a sort of commutation of the weakening link over links contained inside another exponential box.

This chapter is devoted to present the sharing implementation of the elementary and light variants of mMELL, recalling its main properties and proving its correctness by syntactical simulations. In Section 5.2 we present the proof-nets of mIELL, that represent a framework of particular convenience because it considerably simplifies the rewriting system, whilst in Section 5.3 we introduce their sharing implementation — the system SG— and its basic properties. A brief review of the most notable qualitative behavioural properties of SG as a Lévy-optimal implementation of mIELL proof-nets is given in Section 5.4. Among these, there is the correctness: any SG-normal-form of a mIELL proof-net $\mathcal{N}$ is the mIELL-normal-form of $\mathcal{N}$. A proof is given in the concluding Section 5.5 (cf. Theorem 5.10), where we employ a simple syntactical simulation of SG in mIELL, exploiting an intermediate rewriting system — the unshared graphs (UG) — that

possess the structure of mIELL proof-nets and a some sharing markers corresponding to SG graphs.

5.2   Elementary and light proof-nets

We present a minimal, propositional and weakening-free fragment of ELL proof-nets, where proof-nets of the corresponding fragment of LLL can also be represented, as well as ELL- and LLL- typed λI- terms.

A ELL proof-net is similar to a MELL proof-net, except from the fact that a vertex can be the secondary door of at most one box. LLL proof-nets, instead, are a bit more complicated, because types/formulæ need to encompass an additional exponential modality $\S$ called paragraph[1], which induces two new kinds ($\S$ and $\bar{\S}$) of binary links with dual polarisations. Hence we have two kind of boxes, associated not only to promotion links, but also to positive paragraph links. !-boxes have at most one secondary door, which is a premiss of a ?-link, $\S$-boxes have instead an arbitrary number of secondary doors, which are premiss of links of kind ? or $\bar{\S}$. These two restrictions tame the amount of duplication possibly needed by the reduction so that, quite interestingly, they are precise representations of, respectively, the complexity classes: Kalmar-elementary time (ELEMENTARY, i.e. iterated exponentials) and polynomial time (PTIME).

Fact 5.1 (Stratification). *Given a LLL or ELL proof-net* $\mathcal{N}$, *let* $v \in V(\mathcal{N})$, $l \in L(\mathcal{N})$, ρ *be reduction step on* $\mathcal{N}$. *If there exist* $\rho(v), \rho(l)$ *residuals of* v, l, *respectively, then* $\ell(v) = \ell(\rho(v))$ *and* $\ell(l) = \ell(\rho(l))$.

Theorem 5.1 (Normalisation and complexity classes [Girard, 1995, Asperti and Roversi, 2002]). *Let* ELEMENTARY *be the set of functions that are computable in time bounded by* $K(h, n)$, *where* n *is the size of the input, and* K *is the Kalmar function defined as follows:*

$$K(0, n) = n, \quad K(i + 1, n) = 2^{K(i,n)}. \tag{5.1}$$

Let PTIME be the set of functions computable in time bounded by a polynomial of the size of the input n.

1. *The set of functions computed by ELL is equal to* ELEMENTARY. *In particular, the length of a ELL reduction on a proof-net* $\mathcal{N}$ *is at most* $K(\ell(\mathcal{N}), |\mathcal{N}|)$.

2. *The set of functions computed by LLL is equal to* PTIME. *In particular, the length of a LLL reduction on a proof-net* $\mathcal{N}$ *is at most a polynomial of degree* $2^{\ell(\mathcal{N})+1}$.

We can now proceed to concretely present our simplified proof-net system and their reduction. We opt to avoid the presence of 0-ary contractions for mere sake of simplicity, since it introduces issues of connectedness that are tedious to address, without any notable change with respect to the complexity aspect that is our primary concern. Classical formulations of ELL and LLL were primarily obtained by mean of more traditional, perhaps orthodox, approaches: sequent-calculi. This is indeed the case both for the original

---

[1]The name most probably comes from an erroneous translation of the French 'paragraphe', i.e. 'section', hence explaining the section sign $\S$ used as its symbol instead of the pilcrow ¶.

presentation by Girard [1995] of LLL and ELL, as well as in the simplifications proposed by Asperti and Roversi [2002], i.e. ILAL and IEAL, where weakening is generalised in its affine form, i.e. the conclusion of such a rule is not required to be a formula/type in the form !T, but can be of any kind. With respect to other proof-nets presented in previous chapters, we still employ the generalised n-ary syntax for negative exponentials, but we drop the polarised [Laurent, 2002] restriction on formulæ/types.

Recall from Section 2.2 the definitions of mMELL pre-nets and, in particular, of boxing function (see Definition 2.4). Our proof-net system is obtained by variation of mMELL proof-nets, where we also remove boxes, — that are useless because of the stratification property — replacing the boxing function by a depth level annotation.

Definition 5.1 (Types). A *type*, or formula, is a word of the following grammar, where $\star$ is the only *ground* type.

$$\mathbb{T} ::= \star \mid \mathbb{T} \multimap \mathbb{T} \mid !\mathbb{T}. \tag{5.2}$$

Definition 5.2 (Elementary boxes). Given a pre-net $\mathcal{P}$ and a !-link $l \in L(\mathcal{P})$, let b be a boxing function. If according to b every vertex is the secondary door of at most one box, then b is an *elementary boxing*. Formally, if $b(l) = \mathcal{B}$, for some sub-pre-net $\mathcal{B} \subseteq \mathcal{P}$, and v is a secondary door of $\mathcal{B}$, then there exists no other !-link $l' \in L(\mathcal{P})$ such that $b(l') = \mathcal{B}'$ and v is a secondary door of $\mathcal{B}'$.

Definition 5.3 (mIELL proof-nets). Given a mMELL proof-net $(\mathcal{P}, b)$, we say $\mathcal{N} = (\mathcal{P}, \ell)$ is a mIELL proof net if: arity of links in $L(\mathcal{P})$ is strictly greater than 0, $\mathcal{P}$ has a typing function in $\mathbb{T}$, b is an elementary boxing, ℓ is the depth according to b. A *box* in $\mathcal{N}$ is the maximal sub-pre-net $\mathcal{P}' \subseteq \mathcal{P}$ such that, for any $v \in \mathcal{B}$ we have $\ell(v) \geq l$, for some $l \in \mathbb{N}$.

Proposition 5.1. *Any minimal weakening-free proof-net of ELL or LLL can be encoded as a mIELL proof-net.*

Proof sketch. The case of IEAL is immediate, since it suffices to remove second order links. The case of ILAL simply additionally requires the replacement of §-links with !-links and §̄-links with ?-links, so that §-boxes become !-boxes. ∎

Definition 5.4 (mELL and mLLL reduction). The rewriting relation $\rightarrow_{\text{mIELL}}$ is the graph-rewriting relation on mLLL or mLLL proof-nets, obtained by the context closures of the union of the following two redexes:

Logics $(\multimap)$, as defined in Figure 5.1;

Duplication (D) as defined in Figure 5.2.

Since there will be no chance of ambiguity, in the present and next chapters of this second part, we shall refer to mIELL proof-nets simply as proof-nets.

5.3   Sharing implementation

In this section we introduce sharing graphs and their reduction, and we recall the most important qualitative behavioural properties, as a rewriting system. About the peculiar

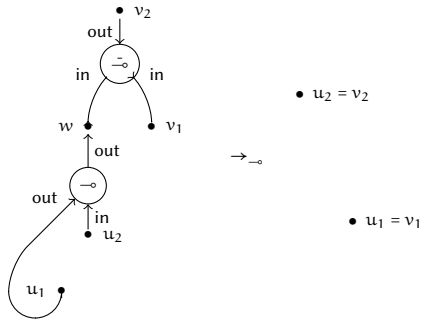**Figure 5.1** Linear implication reduction

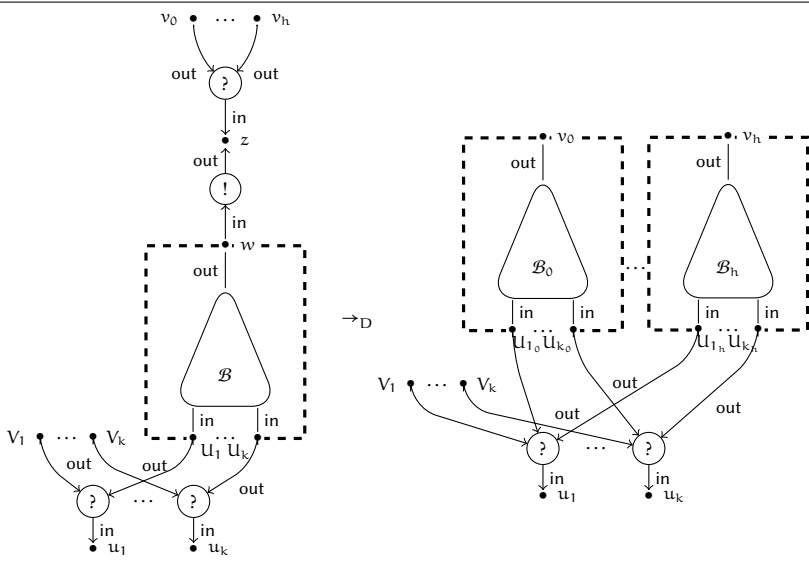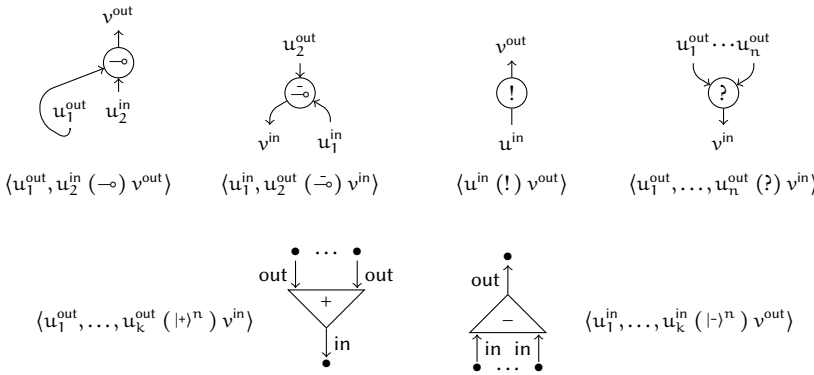
Figure 5.2 Duplication reduction

Figure 5.3 SG links: kind, arity and polarity associated to vertices, in both graphical and textual notations. Above, from left to right: abstraction, application, promotion, contraction (or weakening when $n = 0$); Below, positive and negative mux, with $k > 1$.

$$\langle u_1^{out}, u_2^{in} \; (\multimap) \; v^{out} \rangle \qquad \langle u_1^{in}, u_2^{out} \; (\bar{\multimap}) \; v^{in} \rangle \qquad \langle u^{in} \; (!) \; v^{out} \rangle \qquad \langle u_1^{out}, \ldots, u_n^{out} \; (?) \; v^{in} \rangle$$

$$\langle u_1^{out}, \ldots, u_k^{out} \; (\vdash)^n) \; v^{in} \rangle \qquad \qquad \langle u_1^{in}, \ldots, u_k^{in} \; (\dashv)^n) \; v^{out} \rangle$$

tracts of the present formulation with respect to previous literature, some clarification are in order. First, sharing nodes are n-ary muxes, and not binary fans, and are directly introduced by the reduction itself by a "triggering" rule that replaces the two main links of exponential redex by a mux, and not by means of an initial translation, and they are dually removed by a "merging" rule. Second, since the propagation of a unary mux would be nothing but a redundancy, which will be not quite desirable when in the Chapter 6 the efficiency will be our primary concern, the triggering is allowed only for contractions with arity greater than 2. Third, two unneeded components are excluded by the formulation. The level-management component, or the oracle, which dynamically adjust depth of other links, is made useless by the stratification property. The garbage collector, which erases links, is made useless by the absence of weakening we imposed.

5.3.1 Definition

The core of sharing graphs are multiplexer links, or muxes. From the input-output point of view, the positive mux is close to the contraction ?-link, its conclusion being the input of the link, and its k premisses being its outputs. It makes the sub-term rooted in input available in output contexts. A negative mux conversely has its k premisses as input and its conclusion as output. It shares the output context to the various subterms rooted in its inputs.

To our knowledge, no geometrical correctness criterion has been found, which would allow to discriminate when an arbitrary graph built with logical links and muxes represent correct proof-net without unfolding it. The set of sharing graphs is therefore defined as the transitive closure of the sharing and read-back reductions on proof-net.

Definition 5.5 (SG pre-nets and reductions)**.** A *SG link* is a link having kind in $\{\multimap, \bar{\multimap}, !, ?, \vdash, \dashv\}$ and whose arity and assigned polarities are determined by the link's kind, as shown in Figure 5.3. Links of kind $\vdash, \dashv$ are called *multiplexers*, and have one conclusion and $k > 1$ premisses. The rewriting relation $\to_{SG}$ is the graph-rewriting relation on SG pre-nets, i.e. pre-nets made by SG links, obtained by the context closure of the union of following redexes.

Figure 5.4 Logical redexes.

(a) Linear implication.

(b) Light exponential, when k = 0, and mux triggering, when k > 0.
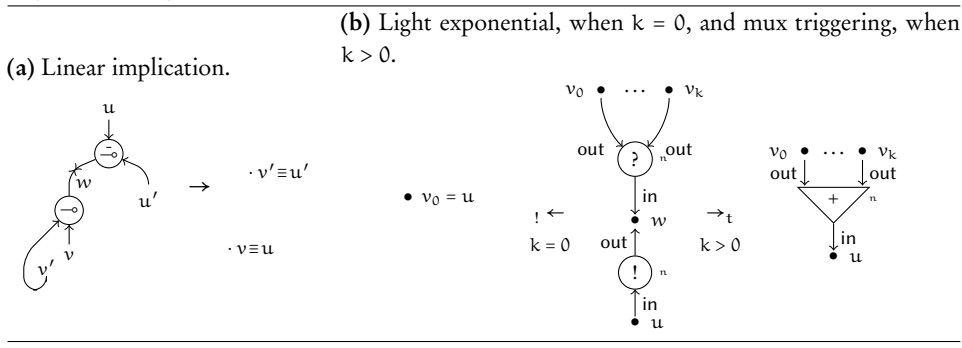
Logics $(\multimap), (!), (t)$, defined in Figure 5.4a and Figure 5.4b.

Duplication $(d \multimap), (d \stackrel{=}{\multimap}), (d!), (d?)$, defined in Figure 5.5a, 5.5b, 5.5c and 5.5d.

Mux interaction $(a), (s)$, defined in Figure 5.6a and 5.6b.

The rewriting relation $\to_{RB}$ is the context closure of the union of the mux interaction rules and

Readback duplication $(r \multimap), (r?), (m)$, as defined in Figure 5.7, 5.8b and 5.8a.

The RB-normal form of a graph $\mathcal{G}$ is called its *read-back* and written R(G). We shall denote as $\to$ SGRB the union of the SG and RB relations.

Definition 5.6 (SG graph). A SG graph is either a mLLL proof-net, or the reduct of a SG graph via SG- or RB-reduction.

Remark 5.1. The merging rule (m) of RB requires that the level n of the mux and the level m of contraction are equal, while $(r?)$ requires that $m > n$. In the more general case of MELL, the main vertex of a (m) redex may be a secondary door of a number of boxes, therefore the definition requires $m \leq n$. We recall that in the case of mIELL that we are considering here, it cannot be the case that $m < n$, which would cause the mux to be deadlocked (see Guerrini [1999], or employ l-contexts that will be introduced in Section 6.4).

5.3.2 Rewriting properties

Sharing graphs with SG-reduction form an orthogonal rewriting system: for every hyperlink h in a given graph there exists at most one reduction involving h, that is the interaction along the unique principal door of h. Indeed, they are interaction nets, therefore they are locally confluent. The RB-reduction includes instead many rules which allows interactions between links that are not connected by their principal vertices. But In this case, as well as in the combined one, the system still enjoys good rewriting properties.
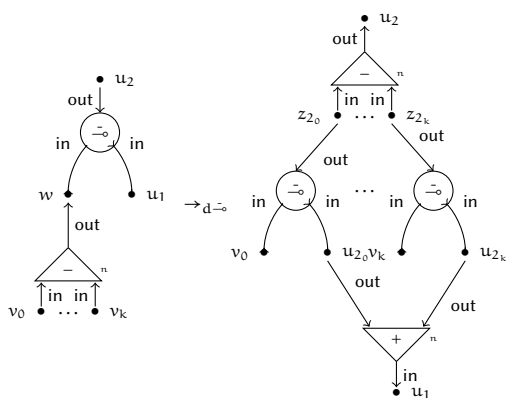
Theorem 5.2 (Confluence and normalisation). *The rewriting systems RB and SGRB are strongly normalising and confluent. Their normal forms are proof-nets.*
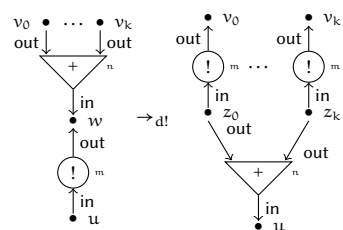
Figure 5.5 Duplication redexes, where m > n.

(a) Positive linear implication link.
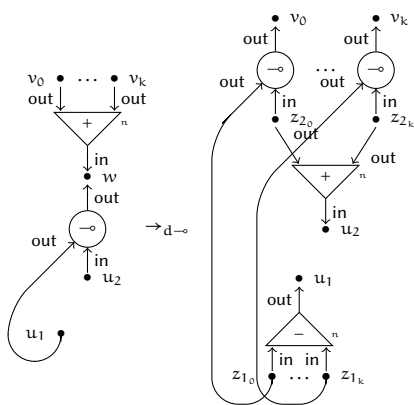
(b) Negative linear implication link.

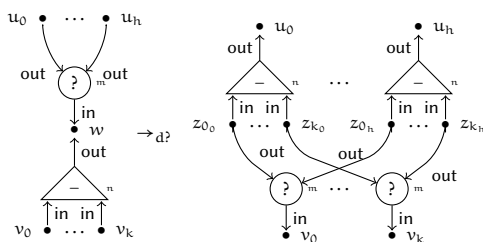**(c)** Elementary promotion

(d) Elementary contraction

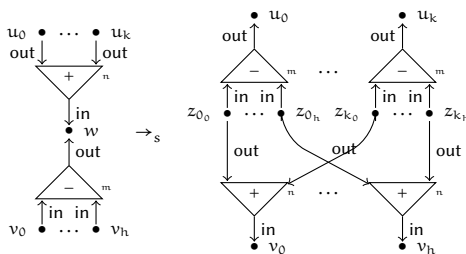**Figure 5.6** Mux interaction redexes.
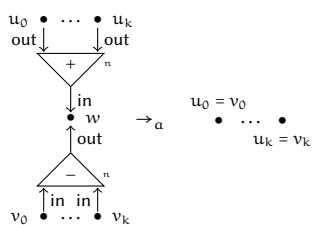
(a) Annihilation.

(b) Swap, where m ≠ n

Figure 5.7 Readback of negative implication, where m > n.
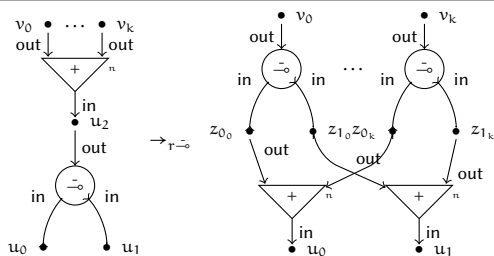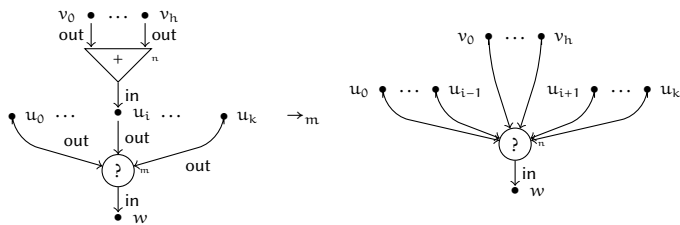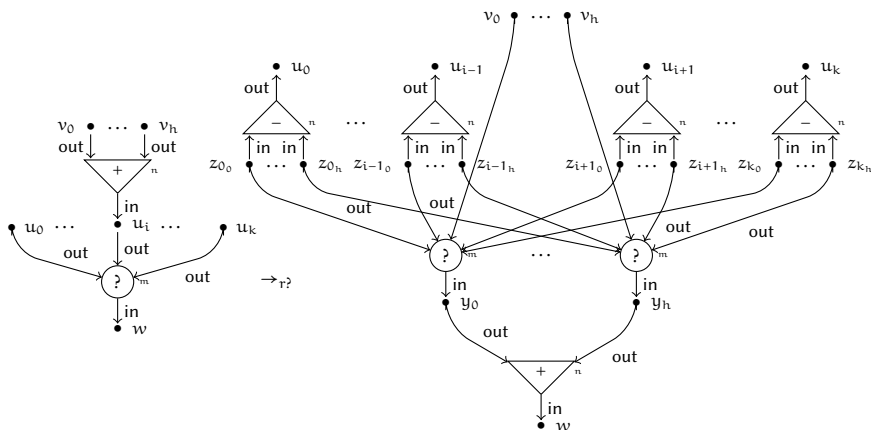
Figure 5.8 Readback redexes on contraction.

(a) Merge into contraction, when $m = n$.

(b) Duplication, when $m > n$.

Proof hint. With respect to the work by Guerrini et al. [2003, Theorems 4, 11.i and 11.ii], we have here only two minor differences: axioms and cut links are implicit, readback rules are restricted to positive muxes only. ∎

Any SGRB sequence can be rearranged in order to postpone the read-back, i.e. the non-optimal duplications, at the end of the sharing normalisation. This is indeed the standard way of employing sharing graphs as an implementation.

Theorem 5.3 (Safety of laziness [Guerrini et al., 2003, Theorem 15])**.** *Let N be a proof-net and $\bar{N}$ its SGRB -normal form. If G is the SG-normal form of N, then $G \to^*_{RB} \bar{N}$.*

5.4 Adequacy properties

5.4.1 Correctness

Sharing graphs are a correct implementation of cut-elimination of proof-nets.

Theorem 5.4 (Correctness)**.** *Let N be a proof-net such that $N \to^*_{SG} G$. Then there exists $N \to^*_{mlELL} R(G)$.*

The correctness of the sharing implementation in the general case of λ-calculus was first established using a semantic analysis of GoI paths in the graphs [Gonthier, Abadi,
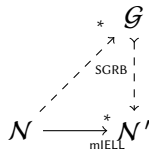
and Lévy, 1992a]. A comprehensive discussion is provided by Asperti [1998]. The first results of correctness obtained by syntactical means has been presented by Guerrini [1999], where sharing graphs implements a generic framework of graph rewriting, and little later by Guerrini et al., who consider [2003, Theorem 13] the case of MELL proof-nets. In a restricted case similar to our interest — IEAL and ILAL proof-nets — correctness was also more recently proven with GoI-like semantics tools [Baillot, Coppola, and Dal Lago, 2011].

We shall use an adaptation of the syntactic approach, based on an intermediate rewriting system called unshared graphs. The stratification property of mIELL allows us to prove correctness by a simpler and neater presentation, provided here in Theorem 5.8.
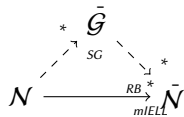
5.4.2 Weak completeness

If we forget optimality, we can consider the rewriting system where all the duplication rules are allowed, i.e. where muxes are allowed to propagate exhaustively and to complete every box duplication process. Then SGRB reduction can simulate any standard mIELL reduction, simply by using a strategy which prioritises RB rules and which thus always performs exhaustive box duplications.

Theorem 5.5 (Suboptimal completeness). *For every proof-net N and every reduction $N \to_{mIELL} N'$ there is a reduction $N \to^+_{SGRB} N'$.*

$$
\begin{array}{ccc}
 & & \mathcal{G} \\
 & {}^*\nearrow & \Big\downarrow {\scriptstyle \text{SGRB}} \\
N & \xrightarrow[\;mIELL\;]{*} & N'
\end{array}
$$

Indeed, the canonical reduction strategy on sharing graphs applies the read-back only at the end of a sharing normalisation, in order to maximise sharing and its efficiency. In this case with two reduction phases, the sharing implementation is complete with respect to normalisation.

Theorem 5.6 (Normalisation completeness). *For every proof-net N that mIELL-normalises $\bar{N}$, there are a sharing reduction and a read-back reduction such that $N \to^*_{SG} \bar{\mathcal{G}} \to^*_{RB} \bar{N}$. Diagrammatically:*

$$
\begin{array}{ccc}
 & \bar{\mathcal{G}} & \\
 {}^*\nearrow {\scriptstyle SG} & & {\scriptstyle RB}\searrow {}^* \\
N & \xrightarrow[\;mIELL\;]{} & \bar{N}
\end{array}
$$

A proof by semantic reasoning for the very similar framework of mLLL proof-nets is given by Baillot et al. [2011, Theorem 4], whilst a syntactical one for λ-calculus is presented in Asperti and Guerrini [1998, Theorem 7.9.3.ii].

5.4.3 Optimality

Finally, we recall that any strategy of SG-reduction which lazily postpones mux rules minimises the number of logical steps.

**Theorem 5.7** (Optimality). *SG-reduction implements Lévy optimal reduction.*

In the setting of MELL proof-nets, a hint to a proof argument is presented by Guerrini et al. [2003, Theorem 14]. In the λ-calculus case, instead, one can find a formal statement and a complete proof, once again, in the comprehensive presentation by Asperti and Guerrini [1998, Theorem 5.6.4].

5.5 Correctness by syntactical simulation

In this section we present unshared graphs and their dynamics. Also, we detail the simulation of sharing reduction by unshared reduction, which allows us easily transfer some qualitative properties from the latter to the former. Since it is trivial to define a simulation of unshared graphs by proof-nets, we obtain by composition also a simulation of the sharing reduction by proof-nets reduction, which enable us to conclude the correctness of the sharing implementation. Such simulation will also be used in page 107 to accomplish the complexity comparison between the two. This essentially constitutes a simplified rephrasing of the syntactical proof of the correctness of SG with respect to MELL [Guerrini, 1999, Guerrini, Martini, and Masini, 2003].

5.5.1 Unshared graphs

From a static point of view, the unshared graphs are a variation of sharing graphs, where muxes are necessarily unary and called *lifts*.
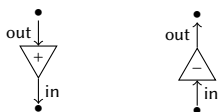
The dynamics of the unshared system is defined similarly to sharing graph's one. We obtain UG-graphs as the reflexive transitive closure of the unshared reduction rules $\rightarrow_{\text{UG}}$ on the class of proof-nets. The only difference in the definition of UG-reduction with respect to SG-reduction is the replacement of the mux triggering rule (t), which could be fired on a cut between a non-unary contraction and a promotion. Such a rule is replaced with a new rule (tD) that is depicted in Figure 5.9. If the ?-link has $h + 1$ premisses, the rule not only duplicates the box creating h new copies of it, very similarly to (D) rule of mIELL-reduction (cf. Figure 5.2), but it also triggers the creation of $h + 1$ lifts, morally where a $h + 1$-ary mux would have been created by a (t) step of SG-reduction (cf. Figure 5.4b).

Definition 5.7 (UG reductions and graphs). The *unshared reduction* is the rewriting relation $\rightarrow_{\text{UG}}$ given by the union of the followings.

Figure 5.9 Duplication and triggering rule (where h > 1).
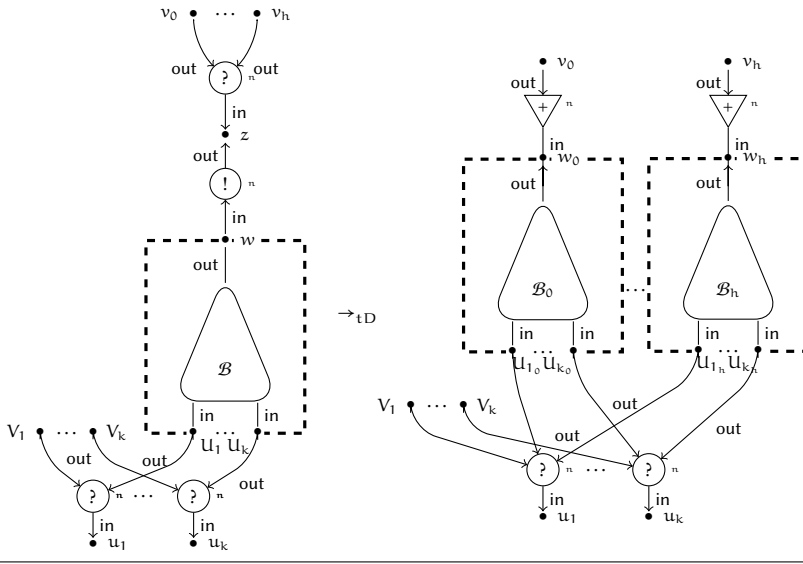
Logics $(\multimap)$, $(!)$, (tD), defined in Figure 5.4a, 5.4b (when k = 0) and 5.9 (when k > 0).

Duplication $(\mathrm{d}\multimap)$, $(\mathrm{d}\overline{\multimap})$, $(\mathrm{d}!)$, $(\mathrm{d}?)$, defined in Figure 5.5a, 5.5b, 5.5c and 5.5d.

Mux interaction (a), (s), defined in Figure 5.6a and 5.6b.

The *unshared read-back* $\rightarrow_{\mathrm{RB}}$ consists of the same rules of the RB-reduction (cf. Definition 5.5). The union of these two reduction relations is written UGRB. A UG graph is either a mIELL proof-net, or the reduct of a UG graph via UG- or URB-reduction.

5.5.2 From sharing graphs to unshared graphs

We aim at a relation between sharing graphs and unshared graphs which relates pairs $(\mathcal{G}, \mathcal{U})$ such that $\mathcal{U}$ represent the unfolding of the proof or term which is compressed in $\mathcal{G}$. The idea is to have a graph morphism relating, when possible, a mux m in $\mathcal{G}$ to the lifts $l_1, \ldots, l_k$ in $\mathcal{U}$ so that the subgraph shared by m corresponds to the subgraph scoped by each of l_i (with $1 \leq k$). This allow us to interpret lifts as sharing markers which demarcate the boundary of shareable sub-proofs. Every SG-sequence has indeed a natural simulation by means of a UG-sequence, while any RB-sequence similarly can be simulated by a URB-sequence. In both cases, the simulation preserves the unfolding relation.
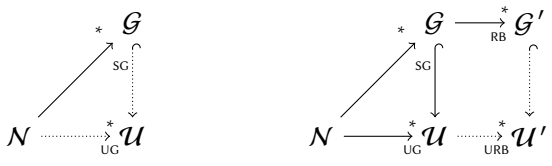
Definition 5.8 (Sharing morphism). A *sharing morphism* $\mathfrak{m}$ is a surjective homomorphism on sharing graphs that preserves the kind, the vertex label, and the level of links. A

sharing graph $\mathcal{G}$ *unfolds* to an unshared graph $\mathcal{U}$, written $\mathcal{G} \hookrightarrow \mathcal{U}$, if there is a sharing morphism $\mathfrak{m}$ such that $\mathfrak{m}(\mathcal{U}) = \mathcal{G}$. We shall use the same notation to relate vertices and links: if $w \in V(\mathcal{G})$ and $W \subseteq V(\mathcal{U})$ we write $w \hookrightarrow W$ to mean $\mathfrak{m}(W) = v$, while if $m \in L(\mathcal{G})$ and $M \subseteq L(\mathcal{U})$ we write $m \hookrightarrow M$ for $\mathfrak{m}(M) = m$.

Fact 5.2. *Let $\mathcal{G} \hookrightarrow N$ respectively be a SG-graph and a mIELL proof-net, and assume that $R \hookrightarrow \mathbf{R}$, for some $R \subseteq \mathcal{G}$ and some set $\mathbf{R}$ of sub-proof-net of N. If R is an SG-redex then $\mathbf{R}$ is a set of UG-redexes. In particular, if R is a (t)-redex, then any $R' \in \mathbf{R}$ is a (tD)-rule, otherwise R, R' are of the same kind.*

Lemma 5.1 (UG simulates SG). *For any proof-net N such that $N \to_{SG}^* \mathcal{G}$ exists a UG-graph $\mathcal{U}'$ such that $N \to_{UG}^* \mathcal{U}$ and $\mathcal{G} \hookrightarrow \mathcal{U}$.*

Proof. We proceed by induction on the SG-reduction sequence $\bar{\sigma}$. The base case is trivial, so let $\bar{\sigma} = \bar{\sigma}'\sigma$ for some SG-sequence $\bar{\sigma}'$ and some SG-step σ; and let r be the redex of σ. By inductive hypothesis (IH), $\mathfrak{m}'(\bar{\mu}'(N)) = \bar{\sigma}'(N)$ for some sharing morphism $\mathfrak{m}'$. Consider the set of redexes $\mathbf{R}$ in $\mathcal{U}$ such that $r \hookrightarrow \mathbf{R}$. Now, any $r' \in \mathbf{R}$ is a redex, as per Fact 5.2, and it is disjoint to any $r'' \in \mathbf{R}$, by orthogonality of the definition of UG-reduction. So let $\bar{\mu}$ be a reduction sequence reducing all and only $r' \in \mathbf{R}$. It is now easy to define $\mathfrak{m}$ from $\mathfrak{m}'$, so that it maps any residual of r' into the residual of r. Hence, $\mathfrak{m}(\bar{\mu}(\bar{\mu}'(N))) = \sigma(\bar{\sigma}'(N))$. ∎

Lemma 5.2 (URB-reduction simulates RB-reduction). *For any SG-graph $\mathcal{G}$ and UG-graph $\mathcal{U}$ such that $\mathcal{G} \hookrightarrow \mathcal{U}$, if $\mathcal{G} \to_{RB}^* \mathcal{G}'$ then $\mathcal{U} \to_{URB}^* \mathcal{U}'$ and $\mathcal{G}' \hookrightarrow \mathcal{U}'$.*

Proof sketch. Almost identical to the proof of previous Lemma 5.1. The only difference worth to mention is the fact that the set of UG-redexes simulating a (m)-step overlaps, but this poses no problem, since they enjoy pair-wise confluence. ∎

Definition 5.9 (Unfolding reduction). Let $\bar{\sigma}$ be a SG- or RB-reduction sequence on a proof-net N, and let $\bar{\mu}$ be a UG- or RB-reduction sequence on $\mathcal{U}$. If $\bar{\sigma}(N) \hookrightarrow \bar{\mu}(N)$, and $\bar{\mu}$ is defined as in the proof of Lemma 5.1, then we call it an *unfolding reduction* of $\bar{\sigma}$, and we write $\sigma \hookrightarrow \mu$.

5.5.3 From unshared graphs to proof-nets

Unshared graphs can be straightforwardly mapped to proof-nets: it suffices to wipe out all the lift, and to recover the lost connections in the hypergraph. Hence, every unshared reduction can be simulated by the proof-net reduction.
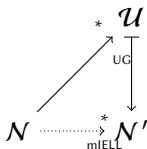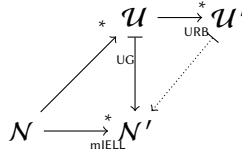
Definition 5.10 (Lift erasure). The *lift erasing* is a function which maps a UG-graphs $\mathcal{U}$ to a proof-net N, written $\mathcal{U} \mapsto N$ or $\mathcal{E}(\mathcal{U}) = N$, by equating any pair of vertices

u, v such that there exists $\langle u \; (\; |* \;) \; \rangle \; v \rangle \in L(\mathcal{U})$. We extend the function $\mapsto$ to vertices and links belonging to some unshared graph.

Lemma 5.3 (mIELL simulates UG). *Let N be a proof-net. If $N \to^*_{UG} \mathcal{U}' \mapsto N'$ then there exists a reduction such that $N \to^*_{mIELL} N'$.*

Proof. By induction on the UG-sequence $\bar{\mu}$ it is easy to build the mIELL-sequence $\bar{\rho}$ containing the matching redex, if present.

1. If a step μ is of type $(\multimap)$ or $(!)$ on some redex R then $R' \leftharpoondown R$ is a redex of the same type. When the step is of type (tD), then corresponding R' is a (D)-redex. In both cases, let ρ be the step reducing R'. It is immediate to note that $\mu(\mathcal{U}) \mapsto \rho(N)$.

2. Otherwise μ is one of $\{(d \multimap), (d \overset{=}{\multimap}), (d!), (d?), (a), (s)\}$ and has no counterpart in mIELL-reduction. In this case let $\bar{\rho}$ be the empty sequence. The claim still holds, because the erasing of the UG-reduct is unchanged.

$\blacksquare$

Lemma 5.4 (Correctness of URB-reduction). *For any UG-graph $\mathcal{U}$ and any proof-net N such that $\mathcal{U} \mapsto N$, if $\mathcal{U} \to^*_{URB} \mathcal{U}'$ then $\mathcal{U}' \mapsto \mathcal{U}$.*

Proof. Identical to the proof of previous Lemma 5.3, in particular to its second case. $\blacksquare$

Definition 5.11 (Erasing reduction). Let $\bar{\mu}$ be a UG- or URB-reduction sequence on a proof-net N, and let $\bar{\rho}$ be a mIELL-reduction sequence on N. If $\bar{\mu}(N) = \mathcal{E}(\bar{\rho}(N))$, and $\bar{\rho}$ is defined as in the proof of Lemma 5.3, then we call it an *erasure reduction* of $\bar{\mu}$, and we write $\sigma \mapsto \mu$.

5.5.4 From sharing graphs to proof-nets

If we erase the lifts from an unshared graph of the unfolding of a sharing graph, we obtain the proof-net that is implemented by it. We show that the sharing and readback reductions are correct with respect to the proof-net reduction and the implementation. Therefore, the read-back normalisation of a SG-graph is equal to the proof-net it implements.
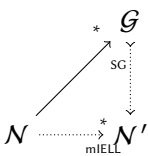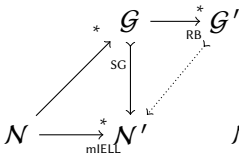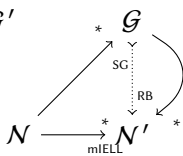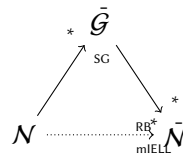
Definition 5.12 (Sharing implementation). The *sharing implementation* is the relation between sharing graphs and proof-nets obtained by composition of erasure and unfolding. A proof-net N is *implemented by* a sharing graph G, written $G \twoheadrightarrow N$, where there is UG-graph $\mathcal{U}$ such that $G \hookrightarrow \mathcal{U} \mapsto N$.

Theorem 5.8 (mIELL-reduction simulates SG-reduction). *Let N be a proof-net. If $N \to^*_{SG}$ G then there exists $N \to^*_{mIELL} N'$ such that $G \twoheadrightarrow N'$.*

Proof. Immediate from Lemma 5.1 and Lemma 5.3. ∎

Theorem 5.9 (Correctness of read-back reduction). *For any sharing graph G and proof-net N' such that $G \twoheadrightarrow N'$ and G, N' have a common proof-net ancestor N, if $G \to^*_{RB} G'$, then $G' \twoheadrightarrow N'$.*

Proof. Immediate from Lemma 5.2 and Lemma 5.4. ∎

Corollary 5.1 (Correctness of read-back normalisation). *For any proof-net N, let N' and G respectively be a proof-net such that $N \to^*_{mIELL} N'$, and a sharing graph such that $N \to^*_{SG} G$. If $G \to^*_{RB} N'$ then $G \twoheadrightarrow N'$.*

Proof. Immediate, by considering normalisation and confluence property of read-back reduction, and the fact that any RB-normal form is a proof-net. ∎

Theorem 5.10 (Normalisation correctness). *For every proof-net N, if it normalises with a sharing reduction and a read-back reduction $N \to^*_{SG} \bar{G} \to^*_{RB} \bar{N}$, then there is a normalisation $N \to_{mIELL} \bar{N}$.*

Proof. Immediate, because the mIELL-reduction is the simulation of a SG-normalisation (cf. Theorem 5.8). ∎

Chapter 6

Efficiency of sharing implementation

This chapter analyses the complexity of SG reductions with respect to mIELL reduction, proving that the former cannot be outperformed by the latter, up to a quadratic factor. Section 6.1 will provide an introductive description of the proof, while outlining the whole chapter.

The results presented here have been developed in collaboration with Guerrini, and previously communicated only in a preliminary form [Guerrini, Leventis, and Solieri, 2012]. The main technique significantly extends an embryonic formulation by the author (2011), that allows to bound the overhead of SG-implementations of IEAL proofnets, or ILAL ones, only with an elementary function, respectively a polynomial one.

Contents

6.1 Introduction

SG and mIELL are two reduction systems that, although equivalent with respect to nor-malisation, and although tightly related by a simulation, present a substantial beha-vioural difference in the way they perform the duplication. The latter has only one big-step duplicating rule, i.e. (D), whose cost cannot be bounded by a constant, since it depends on the size of the duplicated box. The former has instead several small-step duplicating rules, that are the initial (t), plus the various (dκ) and (rκ), and the rest of mux rules. In all these cases, the cost is essentially atomic, because the only variable is the arity of the mux. Because of this, we shall first define two different cost notions for SG and mIELL reductions (Section 6.2).

Let us look at the mIELL-simulation of a SG reduction, and in particular at the UG reduction sequence between them. Given the trivial role of lifts for the UG rewriting system — they simply propagate themselves, commuting with all other links — once we label lifts with unique names preserved under copy, it is possible to identify, for every vertex and link in the graph, the exact set of lifts (or their copies) that eventually will come there along any reduction. In order to do so, given the simplicity of lift dynamics, it's hardly worth bring a GoI approach into its analysis, together with straight paths, which loop back and forth around the graph. It suffices to consider a new kind of paths, called in/out paths, which goes always downward, or upward, with respect to figures, i.e. they always travel from output to input, or vice versa (Section 6.3). Then, on a given path we are able to statically inspect the sequence of lifts and infer their behaviour: positive ones need to be propagated downward, negative ones upward, lifts on the same level erase each other. To formalise this, we do not need the complexity of the dynamic algebra, since it can be algebraically modelled as a language of parentheses called sharing contexts (Section 6.4). The rough intuition behind them is that ( ) = [ ] = 1, while ( ] = [ ) = 0. We shall prove that the sharing contexts assigned to rooted paths cannot be 0 nor begin with a closed parenthesis, i.e. a negative lift (Proposition 6.1). Moreover, as anticipated, we show that sharing contexts morally belong to vertices, rather than paths, because rooted paths reaching the same vertex always carry the same context

(Lemma 6.14).

With the aid of these tools, on the one hand, we can determine the set of sub-graphs of a UG graph $\mathcal{U}$ where a lift propagation will happen. We shall call it the share of $\mathcal{U}$ (Subsection 6.5.1). On the other hand, among a set of lifts that are the unfolding of a mux m with $k + 1$ premises, we can select which lifts are the original instances, i.e. created by the (tD)-step which simulates the (t)-step that creates m, and which are their copies, i.e. duplicated by other (tD)-steps. We shall call the latter ones boundary lifts and show that their number is equal to the arity k of the mux.

We then obtain the key point to precisely transfer (Section 6.5) and compare (Section 6.6) the costs of mIELL and SG reductions in UG reductions. On the mIELL side, they key idea is to count the size of removed share. The (tD)-step can then have a constant cost, because the share grows exactly as the size of the duplicated box. Moreover, when (dκ) and (rκ) rules involve boundary lifts, they shrink the share, so they also have a constant cost. But on the SG side, we already noticed that lift movements are costly only when they are boundary, so the cost of duplications match between the two systems: the SG-cost (beside an additional technical detail) is bounded by a linear function of the mIELL-cost (Lemma 6.20).

What about the other rules (Lemma 6.22)? The logical rules ($\multimap$) and (!) and (t) are obviously the ones where SG outperforms mIELL. Indeed, each redex R in a SG graph $\mathcal{G}$ has exactly one image in the simulating $\mathcal{U}$ that does not belong to the share, so we can say that this 'master copy' is the costly one, from the point of view of SG reduction, whilst all the other shared redexes have no cost. Conversely, these latter images of R (there is at least one) are costly with respect to mIELL reduction, since they remove links and vertices from the share. Even more, each of them has double cost: the instantiation and the elimination. Therefore, for this portion we obtain a loose linear bound for SG-cost.

Last[1] we proceed to discuss annihilations, merges, swaps, which at a first glance they are the most difficult. Once again, we can transfer exactly the SG cost of a step of this kind to the simulating UG sequence:

   i. the rules (a) and (m) are costly when involve boundary lifts;

   ii. the rule (s) is costly when creates new boundary lifts.

Rules of case (i) produce no variation at all in the size of the share, so we are forced to provide a bound by external considerations. We limit the number of boundary annihilations and merges to the cost of other operations — any boundary lift destruction is necessarily preceded by a lift creation, at most equal to the cost of other operations — so we have a linear overhead here also.

Rules of kind (ii), instead, usually produce an atomic decrease to the size of the share, hence they introduce no overhead. But this is not always the case: we have no share decrease if the (s) redex is contained in a share component made by nothing but lifts. The number of such operations grows with the product of the depth level of the graph and the square of the mIELL cost.

---

[1] ... and least!

Hence, given the appropriate cost definitions $\mathbb{C}_{\mathsf{mIELL}}$ and $\mathbb{C}_{\mathsf{SG}}$ (Section 6.2), we will be able to prove that the overhead of an SGRB-reduction, i.e. including the read-back rules, with respect to its mIELL simulation is at most quadratic.

Theorem 6.1 (SG cost bound). *Let N be a proof-net such that $N \xrightarrow{\bar{\sigma}} {}^{*}_{SGRB} G'$ and $N \xrightarrow{\bar{\rho}} {}^{*}_{mIELL} N'$, where $G \twoheadrightarrow N'$. Then there exists a quadratic function $\mathfrak{f}$ such that*

$$\mathbb{C}_{SG}(\bar{\sigma}) \leq \mathfrak{f}\big(\mathbb{C}_{mIELL}(\bar{\rho})\big). \tag{6.1}$$

6.2 Cost measures

We now assign a cost to mIELL and SG reductions. Although a formulation that employs a unique *a priori* notion for both the reductions would look at a first glance more elegant and clean, we introduce some differences between the two. Such constant, and thus morally negligible, initial asymmetry will later repay with symmetries that ease our accounting. The mIELL cost counts for the size, i.e. the number of vertices and links, that are erased and/or added to the graph.

Definition 6.1 (Size). The *size* of a pre-net $\mathcal{P}$, written $\#(\mathcal{P})$, is the sum of the cardinality of the set of its vertices and the sum of the arities of its links:

$$\#(\mathcal{P}) = |V(\mathcal{P})| + \#(L(\mathcal{P})), \tag{6.2}$$

$$\#(L(\mathcal{P})) = \sum_{l \in L(\mathcal{P})} |l|. \tag{6.3}$$

We remark that for a box $\langle u_1, \ldots, u_k \; [\mathcal{B}] \; v \rangle$, all of its doors, primary and secondary ones, belong to the sub-graph $\mathcal{P}$ and are accounted by $\#(\mathcal{P})$.

Definition 6.2 (mIELL-reduction cost). The cost $\mathbb{C}_{\mathsf{mIELL}}(\rho)$ of a mIELL-reduction step ρ on a proof-net N is defined as the size of the symmetric difference, written $\ominus$, between the sets of vertices and links of N and those of $\rho(N)$. The cost of a reduction sequence $\bar{\rho}$ is the sum of the costs of each step which is composed of.

$$\mathbb{C}_{\mathsf{mIELL}}(\rho) = |V(N) \ominus V(\rho(N))| + \#(L(N) \ominus L(\rho(N))); \tag{6.4}$$

$$\mathbb{C}_{\mathsf{mIELL}}(\bar{\rho}) = \sum_{\rho \in \bar{\rho}} \mathbb{C}_{\mathsf{mIELL}}(\rho). \tag{6.5}$$

Lemma 6.1 (mIELL-reduction cost). *Given a mIELL-reduction step ρ, its cost $\mathbb{C}_{mIELL}(\rho)$ is in Table 6.1.*

Proof. By an immediate inspection of the reduction rules, we first see that the linear beta step erases 3 vertices and 2 links of arity 3. In the case of the $k + 1$-ary duplication of an exponential box $\mathcal{B}$, there are k new copies of the box, giving a size augmentation of $k \times \#(\mathcal{B})$. On the other hand, it removes $k + 2$ vertices, a ?-link of arity $k + 2$ and an !-link of arity 2. ∎

The cost for SG and RB is a bit more parsimonious and is motivated by a dynamic interpretation of the mux links. The traditional viewpoint analyses only the number of hyperlinks, and therefore interprets a $(d\kappa)$-step (for some kind κ) between a k-ary

Table 6.1 Cost of classic reduction rules, where in the case of (D), $k + 1$ is the number of premises of the ?-link, and $\mathcal{B}$ is the box enclosed by the !-link.

Rule	$\mathbb{C}_{\mathsf{mIELL}}(\rho)$
$(\multimap)$	9
(D)	$k \times \#(\mathcal{B}) + 2k + 4$

mux m and a hyperlink h as the duplication of h in k copies. In the case of syntax with fans, i.e. binary muxes, such a perspective would allow to assign a unitary cost to every reduction step. We intend, instead, to have a closer and more accurate look. Supposing $m = \langle v_0, \ldots, v_k \, (|\!\star) \,) \, w \rangle$, we interpret any vertex v_i as a duplicate of w, since morally all copies are already available to the rest of the graph. Moreover, if w is the main vertex of a duplication redex on a link $h = \langle z_1 \ldots z_l \, (\kappa) \, w \rangle$, then the rewriting step acts on h (whose size is $l + 1$) and its l premises, not counting w. Therefore, our cost measure roughly correspond to the variation — i.e. the absolute value of the difference — of the size of the graph ignoring conclusions of mux-links. Remark that such measure is slightly smaller than the length of the sequence of binary duplications.

Definition 6.3 (Cost of SG- and RB-reduction). The *cost* of a SG- or a RB-step σ, is given in Table 6.2. Observe that, given a mux with $k + 1$ premiss, the cost of duplication and interaction rule would be null if $k = 0$. This is the reason for which we exclude such a case by forbidding the creation of unary muxes with the rule (!).

Table 6.2 Sharing and readback rules' cost, where $j + 1$, $k + 1$ and $l + 1$ respectively are the number of premises of the ?-link, the first and the second $|\!\star)$-link, possibly involved.

Rule	$\mathbb{C}_{\mathsf{SG}}(\sigma)$
$(\multimap)$	9
$(!)$	6
(t)	$j + 4$
$(d!)$	$3k$
$(d \multimap), (d \bar{\multimap}), (r \bar{\multimap})$	$5k$
$(d?), (r?)$	$(2j + 3) \times k$
$(a), (m)$	k
(s)	$k \times l$

6.3 Input/output paths

We now introduce a new notion of path characterised by the fact that it travels a link only from an in-vertex to a out-link, or vice versa. We call them in/out paths, and show that they are acyclic and enjoys some connectivity properties with respect to $\multimap$-links and boxes. Finally we also define how they are deformed by the reduction.

6.3.1 Statics

Definition 6.4 (in/out paths). A path $\pi : u_0 \sim u_k$ in a pre-net $\mathcal{P}$ is *downward* if for any $0 \le i < k - 1$ the vertex u_i is an out-vertex of l_i, and u_{i+1} is an in-vertex of l_i and

$u_i \notin \mathsf{BVar}(\mathcal{P})$. If $\pi^\dagger$ is downward, then π is an *upward path* from u_k to u_0. A path is in/out when it is either downward or upward. We shall write $u \gg v$ when there is a downward path from u to v, and $v \ll u$ when $u \gg v$.

Lemma 6.2 (Partial order). *For any UG-graph $\mathcal{U}$:*

1. $\gg$ *is a partial order for* $V(\mathcal{U})$;

2. in/out *paths contain no cycles.*

Proof sketch. By a simple inspection of the definition of $\gg$, we verify that is enjoys reflexivity and transitivity, so we pass to asymmetry and acyclicity. Given a in/out path π in $\mathcal{U}$, let π' be the in/out path in $\mathcal{E}(\mathcal{U})$ obtained from π by making it jump over every lift crossed by π. Remark now that by construction π' is a special case of switching path (cf. Definition 2.9), which considers only one of the two admitted switchings on $\multimap$-links (the one on the second premiss), and which imposes the in/out direction. Moreover, mIELL proof-nets do not contain cyclic swtiching paths, hence π' does not contain cycles. But π differ from the presence of lifts that cannot possibly introduce new cycles, thus also π is acyclic. ∎

Lemma 6.3 (Crossing links). *Let π be a downward path in a UG-graph $\mathcal{U}$ and l a link in $\mathcal{U}$. If l is not a $\multimap$-link, then π contains at most 2 vertices of l; otherwise π contains at most 3 vertices, and in such a case one is the lowermost extremum of π.*

Proof. We separately consider the two disjointed propositions of the claim.

1. Suppose that l is not a $\multimap$-link. Then let u, v be two distinct vertices of π, respectively being an in-vertex and an in-vertex of the link l. Then (u, v) is by definition a downward path over l. Now clearly u must precede v in π, since otherwise, as per antisymmetry property of $\gg$ (Lemma 6.2), we would contradict our distinctness hypothesis. Moreover, u, v have to be consecutive in π, because the existence of $\pi' \neq (u, v)$ would imply that (i) there exist two links l, l' both insisting on u with their in-vertex, and similarly that (ii) there exist two links l, l'' both insisting on v with their out-vertex. Indeed, both these facts would be absurd, since at most one in- and one out-vertex connection are allowed by definitions of unshared graphs. So let $\pi = \pi' :: (u, v) :: \pi''$, and by acyclicity we conclude that there cannot be other be other occurrences of u or v within π' or π''.

2. Suppose otherwise that l is a $\multimap$. In particular, let $l = \langle w, v \; (\multimap) \; u \rangle$. Exactly as in the previous case, we have that u, v must be appear consecutively in such order within π, so let $\pi = \pi' :: (u, v) :: \pi''$. Moreover, u, v cannot occur elsewhere in π' nor π''. Now, by definition of downward path there exists no w' such that $w \gg w'$, therefore it must be the last and possibly unique vertex of π''. Hence the claim. ∎

Lemma 6.4 (Connectivity). *Given $\mathcal{U}$ a UG-graph, for any $v \in V(\mathcal{U})$:*

1. $v \ll u$ *where* u *is the root of* $\mathcal{U}$,

2. $v \gg u'$ *for some* $u' \in \mathsf{FBVar}(\mathcal{U})$.

Proof sketch. Immediate by Definition 2.3. ∎

Lemma 6.5 (Lambda-connectivity). *Let* $\langle u, w \; (\multimap) \; z \rangle$ *be a link in a UG-graph. Then* $v \sim w$ *for any* v *such that* $v \gg u$.

Proof sketch. Almost immediate from Lemma 6.4. ∎

Lemma 6.6 (Box connectivity). *Let* v *be the root of a UG-graph* $\mathcal{U}$ *or the premiss of a !-link, and let* u *be the premiss of a ?-link* h. *If* $\ell(u) > \ell(v)$ *and* $v \gg u$, *then for any* w *premiss of* h *there exists* $\pi : v \gg z \gg w$, *such that* z *is the premiss of the principal door of the box of* w.

Proof sketch. Almost immediate from Lemma 6.4. ∎

6.3.2 Dynamics

Remind the notions of crossing and sufficient length for a path with respect to some redex (cf. Definition 2.18).

Definition 6.5 (UG-residuals of downward maximal crossing). Let χ be a downward path in a UG-graph $\mathcal{U}$ being a maximal crossing for a UG-redex R, and let ρ be its reduction. The *residuals of* χ *with respect to* ρ are the set of downward path defined as follows.

1. If R is a $(\multimap)$-step, let it be as in Figure 5.4a. Then:

$$\rho((v_2, w, u_2)) = \{(u_2 = v_2)\}; \tag{6.6}$$
$$\rho((u_1)) = \{(u_1 = v_1)\}; \tag{6.7}$$
$$\rho((v_2, v_1)) = \{(u_2 = v_2) :: \gamma :: (u_1 = v_1) \mid \gamma : u_2 \gg u_1\}. \tag{6.8}$$

   Notice that the rightmost side of (6.8) is the empty set whenever $u_2 \not\gg u_1$. But it cannot be the case, as per Lemma 6.5.

2. If R is a (!)-step, let it be as in Figure 5.4b Then:

$$\rho((v_0, w, u)) = \{(v_0 = u)\}. \tag{6.9}$$

3. If R is a $(d \multimap)$-step, let it be as in Figure 5.5a. Then:

$$\rho((v_0, w, u_2)) = \{(v_0, z_{2_0}, u_2)\}; \tag{6.10}$$
$$\rho((u_1)) = \{(u_1)\}. \tag{6.11}$$

   Remark in the last equation that we cannot write (u_1, z_0), since it would not preserve the extrema.

4. If R is a $(d\multimap)$-step, let it be as in Figure 5.5b. Then:

$$\rho((u_2, w, v_0)) = \{(u_2, z_{2_0}, v_0)\}; \tag{6.12}$$

$$\rho((u_2, u_1)) = \{(u_2, z_{2_0}, u_{2_0}, u_1)\}. \tag{6.13}$$

5. If R is a $(d!)$-step, let it be as in Figure 5.5c. Then:

$$\rho((v_0, w, u)) = \{(v_0, z_0, u)\}. \tag{6.14}$$

6. If R is a $(d?)$-step, let it be as in Figure 5.5d. Then for any $0 \le i \le h$:

$$\rho((u_i, w, v_0)) = \{(u_i, z_{0_i}, v_0)\}. \tag{6.15}$$

7. If R is a (a)-step, let it be as in Figure 5.6a. Then:

$$\rho((u_0, w, v_0)) = \{(u_0 = v_0)\}. \tag{6.16}$$

8. If R is a (s)-step, let it be as in Figure 5.6b. Then:

$$\rho((u_0, w, v_0)) = \{(u_0, z_{0_0}, v_0)\}. \tag{6.17}$$

9. Otherwise R is a (tD)-step. Let it be as in Figure 5.9. Then:

$$\rho((v_i, z, w) :: \gamma :: (u'_j, u_j)) = \{(v_i, w_i) :: \gamma_i :: (u'_{j_i}, u_j)\}, \tag{6.18}$$

where:

- $0 \le i \le h$, and $1 \le j \le k$;
- $u'_j \in U_j$ is a premiss of the ?-link having conclusion in u_j;
- $\gamma : w \gg u'_j$ is a path in the subnet b;
- γ_i is the i-th copy of γ in the subnet b_i.

Definition 6.6 (URB-residuals of downward maximal crossing). Let χ be a downward path in a UG-graph $\mathcal{U}$ being a maximal crossing for a URB-redex R, and let ρ be its reduction. The *residuals of χ with respect to ρ* are the set of downward path defined as follows.

1. If R is a $(r\multimap)$-step, let it be as in Figure 5.7. Then:

$$\rho((v_0, u_2, u_0)) = \{(v_0, z_{0_0}, u_0)\}; \tag{6.19}$$

$$\rho((v_0, u_2, u_1)) = \{(v_0, z_{1_0}, u_1)\}. \tag{6.20}$$

2. If R is a $(r?)$-step, let it be as in Figure 5.8b. Then for any $j \ne i$:

$$\rho((v_0, u_i, w)) = \{(v_0, y_0, w)\}; \tag{6.21}$$

$$\rho((u_j, w)) = \{(u_j, z_{j_0}, z_{0_0}, w)\}. \tag{6.22}$$

3. If R is a (m)-step, let it be as in Figure 5.8a. Then for any $j \ne i$:

$$\rho((v_0, u_i, w)) = \{(v_0, w)\}; \tag{6.23}$$

$$\rho((u_j, w)) = \{(u_j, w)\}. \tag{6.24}$$

Definition 6.7 (Residual of downward paths). Let π be a downward path in a UG-graph $\mathcal{U}$ and let R be a UG- or a URB-redex such that π is long enough for R. If ρ is the reduction step of R, then the *reduction of π with respect to ρ* is a function from π to a set $\rho(\pi)$ of downward paths in $\rho(\mathcal{U})$. Let

$$\pi = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n :: \pi_n, \tag{6.25}$$

where for any $0 \le i \le n$ the sub-path χ_i is a maximal crossing of a redex R. Then:

$$\rho(\pi) = \{\pi_0 :: \chi_1' :: \pi_1 :: \ldots :: \chi_n' :: \pi_n \mid \chi_i' \in \rho(\chi_i)\}, \tag{6.26}$$

where the residuals of a maximal crossing of R are defined in Definition 6.5 and 6.6. When $\rho(\pi)$ is a singleton, we shall often directly refer to the unique $\pi' \in \rho(\pi)$ simply as $\rho(\pi)$.

6.4 Sharing contexts

In order to study the dynamics lifts in unshared graphs, we introduce a labelling of (vertices representing) occurrences of variables in (graphs representing) λ-terms. Over such labels we construct a very simple algebraic monoidal structure. Its objects, called contexts, are assigned to downward paths and models how lifts match each other. We will discover two key facts. First, lifts of a given level are always well-nested, as if they are parentheses (positive and negative lifts respectively correspond to open and close paarenthesis), where one cannot close some that have not been previously opened, or a stack (positive and negative lifts respectively correspond to the push and pop operations), where one cannot pop something that have been previously pushed. Second, whilst a vertex may have more than one access paths connecting it to the root, or to the "root" of a box, i.e. its principal door, all of these access paths have the same configuration of unmatched lifts (proceeding in the metaphor: unclosed parentheses, or elements in the stack).

6.4.1 Variable occurrences and sharing contexts

Definition 6.8 (Variable occurrences). Given V a finite set of symbols of variable names, $\mathcal{V}$ is the set of *variable occurrences*, that are triples in $V \times \mathbb{N} \times \mathbb{N}$, where the last two components respectively select the index of a single occurrence and the maximal index of all occurrences. Then, in a UG graph $\mathcal{U}$, the *labelling of variable occurrences* VarO maps premisses of ?- or $|\star\rangle$-links to $\mathcal{V}$. We write $x_{i:k}$ to denote (x, i, k) and Var for the first projection of VarO, i.e. if $\text{VarO}(v) = x_{i:k}$, then $\text{Var}(v) = x$. In particular, VarO satisfies the followings properties.

1. Injective on ?-links: if $v \ne w$ are two distinct premisses of ?-links in $\mathcal{U}$, then $\text{VarO}(v) \ne \text{VarO}(w)$.

2. Order-preserving and surjective on ?-links: if $\mathcal{U}$ contains $\langle v_0, \ldots, v_k \,(?)\, w\rangle$, then there exists $x \in V$ such that, for any $0 \le i \le k$, $\text{VarO}(v_i) = x_{i:k} \in \mathcal{V}$.

3. Coherent in the creation of $|*$)-links: if v is a premiss of the main ?-link in a (tD) redex R, and $\mu(v)$ is the premiss of a $|+$)-link, residual of v with respect to the reduction μ of R, then $\text{VarO}(v) = \text{VarO}(\mu(v))$.

4. Coherent in the copy of $|*$)-links: if v is a premiss of a $|*$)-link l and $\mu(v)$ is the premiss of a $|+$)-link residual of v with respect to a reduction μ, then $\text{VarO}(v) = \text{VarO}(\mu(v))$.

Definition 6.9 (Sharing context monoid). The *sharing context monoid* $\mathfrak{S}^*$ is generated over $\mathcal{V} \cup \{0, 1\}$ by a binary operator $\cdot$, called concatenation, and a unary operator $\overline{}$, said negation. Concatenation forms a monoid, having 1 as identity and 0 as absorbing element. If $a, b, c \in \mathfrak{S}^*$, then:

$$a \cdot (b \cdot c) = (a \cdot b) \cdot c \tag{6.27}$$
$$a \cdot 1 = 1 \cdot a = a \tag{6.28}$$
$$a \cdot 0 = 0 \cdot a = 0 \tag{6.29}$$

Negation is involutive and distributes over concatenation reversing its order. If $a, b \in \mathcal{V}$, then:

$$\overline{\overline{a}} = a \tag{6.30}$$
$$\overline{a \cdot b} = \overline{b} \cdot \overline{a} \tag{6.31}$$

Moreover, concatenation of an object and its negation is either neutral or annihilating. If $a, b \in \mathcal{V}$ with $a \neq 0$ and $a \neq b$, then:

$$a \cdot \overline{a} = 1 \tag{6.32}$$
$$a \cdot \overline{b} = 0 \tag{6.33}$$

When we need to distinguish $\mathfrak{S}^*$-equality from syntactic equality, we write $=_{\mathfrak{S}^*}$ for the former and $=$ for the latter.

Definition 6.10 (Levelled sharing contexts). The *levelled sharing contexts*, or simply l-contexts, are sharing contexts whose syntax is enriched by a *lifting* operator $!$ whose n-th iterate is denoted by $!_n$. An l-context γ naturally induces a map from $\mathbb{N}$ to $\mathfrak{S}^*$: for any $a \in \mathfrak{S}^*$, we define $a(n+1) = 1$ and $a(0) = a$, whilst $!\gamma(n+1) = \gamma(n)$ and $!\gamma(0) = 1$. Moreover, l-contexts are uniformly null: if for some $n \in \mathbb{N}$ we have $\gamma(n) = 0$, then $\gamma(m) = 0$ for any $m \in \mathbb{N}$.
Also, we write $(\gamma)|_n$ to denote the *restriction* of γ on n. Namely, if $m \neq n$ then $(\gamma)|_n(m) = \gamma(m)$; otherwise, if $\gamma(m) \neq 0$, then $(\gamma)|_n(m) = 1$. Finally, we shall write $\gamma =_n \gamma'$ to mean that $\gamma(m) = \gamma(m)$ for any $m \geq n$, whilst $\gamma =^n \gamma'$ dually means that $\gamma(m) = \gamma'(m)$ for any $m < n$. Also, $\gamma =_n^{n'} \gamma'$ denotes the fact that $\gamma(m) = \gamma'(m)$ for any $n \leq m < n'$, so that $\gamma =_n^n \gamma'$ means $\gamma(n) = \gamma'(n)$.

Definition 6.11 (Positivity and stable forms). An object $c \in \mathfrak{S}^*$ is *positive*, denoted by $c \geq 1$, if $c = 1$ or $c = a \cdot b$, where $a \in \mathcal{V}$ and $b \geq 1$. Dually, c is *negative*, written $c \leq 1$, if $c = 1$ or $c = \overline{a} \cdot b$ with $a \in \mathcal{V}$ and $b \leq 1$. Observe that the intersection between the sets of positive and negative objects is $1 = \overline{1}$, while the complement of their union is 0. A *stable form* of $c \in \mathfrak{S}^*$ is any $a \cdot b = c$ such that $a \leq 1$ and $b \geq 1$.
Similarly, given an l-context $\gamma \in \mathbb{N} \mapsto \mathfrak{S}^*$, we say that γ is n-*positive* and write that $\gamma \geq_n 1$ if for any $m \geq n$ we have $\gamma(m) \geq 1$. In particular, when $n = 0$, we simply call γ positive, written $\gamma \geq 1$.

Lemma 6.7 (Stability or nullity). *Every* $c \in \mathfrak{S}^*$ *is either equal to* 0 *or to a unique stable form.*

Proof sketch. By orienting $\mathfrak{S}^*$-equations from left to right we obtain a rewriting system. It is terminating — consider as a strictly decreasing measure the sum of: the number of atomic objects, the number of negated concatenation, the number of negated negations. Moreover it is locally confluent. ∎

Lemma 6.8 (Co-neutrality). *If* $a, b \in \mathfrak{S}^*$ *such that* $a \cdot b = a \neq 0$, *then there exists* $c, d \in \mathfrak{S}^*$ *such that* $a = d \cdot c$ *and* $b = \bar{c} \cdot c$, *with* $c \geq 1$.

Proof. We first observe that $b \neq 0$. Indeed, if we suppose otherwise we would obtain that $a \cdot b = 0$. But $a \cdot b = a$, therefore we would have $a = 0$ contradicting our hypothesis. So a, b are not null, hence they have a stable form.

$$a = d' \cdot d \qquad \text{(stable form)} \qquad (6.34)$$
$$b = c' \cdot c \qquad \text{(stable form)} \qquad (6.35)$$
$$a \cdot b = d' \cdot d \cdot c' \cdot c \qquad (6.34), (6.35) \qquad (6.36)$$

Now, $d \cdot c'$ cannot be null, so they have a stable form.

$$d' \cdot c = e' \cdot e \qquad \text{(stable form)} \qquad (6.37)$$
$$a \cdot b = d' \cdot e' \cdot e \cdot c \qquad (6.37), (6.36) \qquad (6.38)$$
$$d' \cdot e' \cdot e \cdot c = d' \cdot d \qquad \text{(hypothesis)}, (6.38) \qquad (6.39)$$
$$e' \cdot e \cdot c = d \qquad (6.39) \qquad (6.40)$$

Observe that by definition $d > 0$, while $e' \leq 1$. Therefore $e' \geq 1$, i.e.

$$e' = 1 \qquad (6.41)$$
$$e \cdot c = d \qquad (6.41), (6.40) \qquad (6.42)$$
$$a = d' \cdot e \cdot c \qquad (6.34), (6.42) \qquad (6.43)$$

which is our first claim. Moreover,

$$d' \cdot e \cdot c = d' \cdot e \cdot c \cdot c' \cdot c \qquad \text{(hypothesis)}, (6.35), (6.43) \qquad (6.44)$$
$$1 = c \cdot c' \qquad (6.44) \qquad (6.45)$$
$$c' = \bar{c} \qquad (6.45) \qquad (6.46)$$
$$b = \bar{c} \cdot c \qquad (6.35), (6.46) \qquad (6.47)$$

that is the second claim. ∎

Definition 6.12 (Levelled sharing context assignment). Given a downward path $\pi : u \gg v$ between two vertices u, v in a UG-graph, the *sharing l-context* of π is defined as follows.

$$\mathfrak{s}\left(()\right) = 1 \qquad (6.48)$$

$$\mathfrak{s}\left(\pi :: (u, v)\right) = \begin{cases} \mathfrak{s}(\pi) \cdot !_n a & \text{if there is } l = \langle u \left(|+ \right)^n \rangle v \rangle \text{ s.t. VarO}(l) = a \\ \mathfrak{s}(\pi) \cdot !_n \bar{a} & \text{if there is } l = \langle v \left(|- \right)^n \rangle u \rangle \text{ s.t. VarO}(l) = a \\ (\mathfrak{s}(\pi))|_n & \text{if there is } \langle u \left(?^n \right) v \rangle \\ \mathfrak{s}(\pi) & \text{if } u, v \text{ belong to a link of kind in } \{-\circ, \bar{-\circ}, !\} \end{cases} \qquad (6.49)$$

6.4.2 Positivity

This subsection is devoted to prove two fundamental properties of sharing l-contexts. The first is that positive and negative lifts are placed along downward paths as matching parentheses, which algebraically means that contexts are always positive (after some level) if they start from the root of the graph (or a box within it). The second is that any reduction different from the $(d \multimap)$-rule does not affect contexts of downward paths.

Proposition 6.1 (Positivity). *Let π be a downward path in a UG-graph $\mathcal{U}$ having maximum in v, being the root of $\mathcal{U}$ or the premiss of a !-link. Then $\mathfrak{s}(\pi) \geq_{\ell(v)} 1$.*

The proof requires four additional properties.

Lemma 6.9 (Long Invariance). *If π is a downward path in a UG-graph $\mathcal{U}$ long enough for the redex of a reduction step ρ, then $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$ for any $\pi' \in \rho(\pi)$.*

Lemma 6.10 (Lambda-compatibility). *Let $\pi : u \gg w$ and $\pi' : w \gg v$ be two downward paths in a UG-graph $\mathcal{U}$ containing $\langle v, w (\multimap) u' \rangle$. If $\mathcal{U}$ is the root of $\mathcal{U}$ or the premiss of a !-link, then $\mathfrak{s}(\pi) =_{\ell(u)} \mathfrak{s}(\pi :: \pi')$.*

Lemma 6.11 (Box compatibility). *Let $\pi : u \gg v$ and $\pi' : u \gg v'$ be two downward paths in a UG-graph $\mathcal{U}$ such that: u is the root of $\mathcal{U}$ or a premiss of a !-link, and v, v' are premisses of the same ?-link, which implies that $\ell(v) = \ell(v')$. Then $\mathfrak{s}(\pi) =_{\ell(u)}^{\ell(v)} \mathfrak{s}(\pi')$.*

Lemma 6.12 (Unary contraction). *Let $\pi : u \gg v$ be a downward path in a UG-graph $\mathcal{U}$, where u is a !-premiss, v is a ?-premiss. If u, v belong to a box $\mathcal{B}$, then $\mathfrak{s}(v)(\ell(v)) = 1$.*

The four lemmas and the positivity proposition are so tightly related one with the other that their proof will require a mutually recursive approach: to prove a statement we will use another as inductive hypothesis.

Proof of Lemma 6.9 (Long Invariance). Let R be the redex of ρ. Since π is long enough for R, let $\pi = \pi_0 :: \chi_1 :: \pi_1 :: \ldots :: \chi_n :: \pi_n$, where χ_c is a maximal crossing of R, for any $1 \leq c \leq n$. Recall from Lemma 6.3 that a link l could be crossed by a downward path at most: twice, if l is not a $\multimap$-link, three times, if it is. But a redex contains two connected links, therefore, if R does not contain a $\multimap$-link then $n = 1$, otherwise $n \leq 2$. Now, if $n = 0$ then $\rho(\pi) = \{\pi\}$, and both claims trivially hold. So we assume otherwise that $n \geq 1$, and proceed by a case analysis on the kind of rule employed by ρ'.

1. Rule $(\multimap)$ and $n \leq 2$. Let R be as in Figure 5.4a. Recall from Definition 6.5 that χ can be of three kinds, among which (u_i) is quite peculiar, since by Lemma 6.3, it does not allow right concatenation, and is the only one that can succeed another maximal crossing of the same redex R, namely (v_2, w, u_2). For these reasons, and by our hypotheses on π, we can simplify our analysis to the following four cases.

   (a) Suppose that π crosses (v_2, w, u_2). Then by (6.6), we have $\rho(\chi) = \{(v_2 = u_2)\}$, and $\rho(\pi) = \{\pi_0 :: (v_2 = u_2) :: \pi_1\}$. So let $\pi' = \pi_0 :: \pi_1$ be the only element of $\rho(\pi)$. Now, by definition of l-context assignment, $\mathfrak{s}(\chi) = \mathfrak{s}(\rho(\chi)) = 1$. Therefore, $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$.

(b) Assume that π crosses (v_2, v_1). This is one of the most interesting cases of this proof. By (6.8) and (6.26), we have

$$\rho(\chi) = \{(u_2 = v_2) :: \gamma :: (u_1 = v_1) \mid \gamma : u_2 \gg u_1\} \tag{6.50}$$

$$= \{\gamma \mid \gamma : u_2 \gg u_1\}, \tag{6.51}$$

$$\rho(\pi) = \rho(\pi_0 :: \chi :: \pi_1) \tag{6.52}$$

$$= \{\pi_0 :: \gamma :: \pi_1 \mid \gamma : u_2 \gg u_1\}. \tag{6.53}$$

Now consider $\pi' = \pi_0 :: \gamma :: \pi_1 \in \rho(\pi)$, for some $\gamma : u_2 \gg u_1$. By definition of context,

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: \gamma :: \pi_1) \tag{6.54}$$

$$= \mathfrak{s}(\pi_0) \cdot \mathfrak{s}(\gamma) \cdot \mathfrak{s}(\pi_1). \tag{6.55}$$

Now we can safely apply Lemma 6.10 on γ and obtain invariance.

$$= \mathfrak{s}(\pi_0) \cdot \mathfrak{s}(\pi_1) \tag{6.56}$$

$$= \mathfrak{s}(\pi_0) \cdot 1 \cdot \mathfrak{s}(\pi_1) \tag{6.57}$$

$$= \mathfrak{s}(\pi_0) \cdot \mathfrak{s}(\chi) \cdot \mathfrak{s}(\pi_1) \tag{6.58}$$

$$= \mathfrak{s}(\pi) \tag{6.59}$$

(c) If π crosses (u_1), we immediately observe that $\rho(\pi) = \{\pi\}$, which trivially means that $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$, for any $\pi' \in \rho(\pi)$.

(d) If π crosses R first in (v_2, w, u_2) and then in (u_1), exactly as we noticed in case 1a, we have a unique $\pi' \in \rho(\pi)$, that is $\pi_0 :: \pi_1$. Thus, $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$.

2. Rule (!) and $n = 1$. Let R be as in the leftmost part of Figure 5.4b, let m be the depth of the !-link, and let also $\chi_1 = (v_0, w, u)$, that is the only possible maximal crossing of R. By definition, its l-context is as follows:

$$\mathfrak{s}(\pi) = (\mathfrak{s}(\pi_0))|_m \cdot \mathfrak{s}(\pi_1). \tag{6.60}$$

By definition of residual, as per (6.9):

$$\rho(\pi) = \rho(\pi_0 :: (v_0, w, u) :: \pi_1) \tag{6.61}$$

$$= \{\pi_0 :: (v_0 = u) :: \pi_1\} \tag{6.62}$$

$$= \{\pi_0 :: \pi_1\}. \tag{6.63}$$

Let $\pi' \in \rho(\pi)$. Its l-context is

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0) \cdot \mathfrak{s}(\pi_1). \tag{6.64}$$

So, let $m' \geq \ell(v)$.

(a) If $m' \neq m$, then $(\mathfrak{s}(\pi_0))|_m(m') = \mathfrak{s}(\pi_0)(m')$, as by Definition 6.10 of restriction operator. Hence, $\mathfrak{s}(\pi')(m') = \mathfrak{s}(\pi)(m')$. Since by inductive hypothesis (IH) we have that $\mathfrak{s}(\pi)(m') \geq 1$, we also have $\mathfrak{s}(\pi')(m') \geq 1$.

(b) Otherwise, $m' = m$. Observe that, by definition of proof-nets, v_0 belong to a box $\mathcal{B}$, so let $\pi_0 = \eta :: \beta$, where β is the maximal suffix of π_0, whose vertices are in $\mathcal{B}$. Hence by definition:

$$\mathfrak{s}(\pi)(m) = \mathfrak{s}(\eta)(m) \cdot (\mathfrak{s}(\beta))|_m(m) \cdot \mathfrak{s}(\pi_1)(m)$$

$$= \mathfrak{s}(\eta)(m) \cdot 1 \cdot \mathfrak{s}(\pi_1)(m). \tag{6.65}$$

Now, by construction, the maximum of β is an !-premiss of $\mathcal{B}$, and v_0 is the only ?-premiss of $\mathcal{B}$, therefore we can apply Lemma 6.12, to obtain that $\mathfrak{s}(\beta)(\mathfrak{m}) = 1$.

$$\mathfrak{s}(\pi')(\mathfrak{m}) = \mathfrak{s}(\eta)(\mathfrak{m}) \cdot \mathfrak{s}(\beta)(\mathfrak{m}) \cdot \mathfrak{s}(\pi_1)(\mathfrak{m})$$
$$\mathfrak{s}(\pi')(\mathfrak{m}) = \mathfrak{s}(\eta)(\mathfrak{m}) \cdot 1 \cdot \mathfrak{s}(\pi_1)(\mathfrak{m}). \tag{6.66}$$

Therefore $\mathfrak{s}(\pi)(\mathfrak{m}) = \mathfrak{s}(\pi)(\mathfrak{m})$.

3. Rule (tD) and $\mathfrak{n} = 1$. Let R be as in Figure 5.9, and if $\mathfrak{l}'$ is the ?-link in R, then let $\mathfrak{m} = \ell(\mathfrak{l}')$ and $\mathfrak{a} = \text{VarO}(\mathfrak{l}')$. Observe that by definition, χ_1 is in the form $(v_j, z, w) :: \gamma :: (U_{j'}, u_{j'})$, where $0 \le j \le h$ with $h > 0$, $1 \le j' \le k$, and γ is a downward path in the box $\mathfrak{b}$. Remember that by definition of residual, and in particular by (6.18),

$$\rho(\pi) = \rho(\pi_0 :: (v_j, z, w) :: \gamma :: (U_{j'}, u_{j'}) :: \pi_1)$$
$$= \left\{ \pi_0 :: (v_j, w_i) :: \gamma_j :: (U_{j'_j}, u_j) :: \pi_1 \right\}. \tag{6.67}$$

Its l-context is by definition:

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0 :: (v_j, z, w) :: \gamma :: (U_{j'}, u_{j'}) :: \pi_1)$$
$$= (\mathfrak{s}(\pi_0))|_{\mathfrak{m}} \cdot (\mathfrak{s}(\gamma))|_{\mathfrak{m}} \cdot \mathfrak{s}(\pi_1) \tag{6.68}$$

Let π' be the unique element of $\rho(\pi)$, and observe it crosses the j-th lift introduced by the reduction, say $\mathfrak{l}''$. By definition of reduction and that of variable occurrences, $\ell(\mathfrak{l}'') = \ell(\mathfrak{l}') = \mathfrak{m}$ and $\text{VarO}(\mathfrak{l}'') = \text{VarO}(\mathfrak{l}') = \mathfrak{a}$. Therefore,

$$\mathfrak{s}(\pi') = \mathfrak{s}\left(\pi_0 :: (v_j, w_j) :: \gamma_j :: (U_{j'_j}, u_j) :: \pi_1 \right)$$
$$= (\mathfrak{s}(\pi_0) \cdot !_{\mathfrak{m}} \mathfrak{a} \cdot \mathfrak{s}(\gamma))|_{\mathfrak{m}} \cdot \mathfrak{s}(\pi_1). \tag{6.69}$$

Let $\mathfrak{m}' \ge \ell(v)$ and consider its equality with respect to $\mathfrak{m}$.

(a) If $\mathfrak{m}' \ne \mathfrak{m}$, we easily conclude. Indeed, by definitions of restriction and lifting we obtain the claim from (6.68) and (6.69).

$$\mathfrak{s}(\pi)(\mathfrak{m}') = (\mathfrak{s}(\pi_0) \cdot \mathfrak{s}(\gamma) \cdot \mathfrak{s}(\pi_1))(\mathfrak{m}')$$
$$= \mathfrak{s}(\pi')(\mathfrak{m}'). \tag{6.70}$$

(b) Otherwise, $\mathfrak{m}' = \mathfrak{m}$. Consider again (6.68) and (6.69), and recall that, applying Proposition 6.1 of positivity as IH, $\mathfrak{s}(\pi_0) \ge_{\ell(v)} 1$ and $\mathfrak{s}(\gamma) \ge_{\mathfrak{m}} 1$. In particular, these imply that respectively $\mathfrak{s}(\pi_0)(\mathfrak{m}) \ge 1$ and $\mathfrak{s}(\gamma)(\mathfrak{m}) \ge 1$. Now, by definition of restriction and lifting operators, we easily rewrite (6.68) into:

$$\mathfrak{s}(\pi)(\mathfrak{m}) = 1 \cdot 1 \cdot (\mathfrak{s}(\pi_1))(\mathfrak{m}) \tag{6.71}$$
$$= \mathfrak{s}(\pi_1)(\mathfrak{m}). \tag{6.72}$$

Now, in (6.69) we can exploit again positivity as IH to obtain $\mathfrak{a} \cdot \mathfrak{s}(\gamma)(\mathfrak{m}) \ge 1$. Therefore we conclude.

$$\mathfrak{s}(\pi')(\mathfrak{m}) = 1 \cdot (\mathfrak{s}(\pi_1))(\mathfrak{m}) \tag{6.73}$$
$$= \mathfrak{s}(\pi)(\mathfrak{m}). \tag{6.74}$$

4. Rule (d!) and $n = 1$. Assume R is as in the left of Figure 5.5c, where $\chi_1 = (v_0, w, u)$. Also, if l is the $|+\rangle$-link in R, let n be its level, and let $a = \text{VarO}(l)$ for some $a \in \mathfrak{S}^*$. By definition of l-context assignment,

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0 :: (v_0, w, u) :: \pi_1) \tag{6.75}$$

$$= \mathfrak{s}(\pi_0) \cdot !_n a \cdot 1 \cdot \mathfrak{s}(\pi_1) \tag{6.76}$$

By definition of residual, and in particular by (6.14):

$$\rho(\pi) = \rho(\pi_0 :: (v_0, w, u) :: \pi_1) \tag{6.77}$$

$$= \{\pi_0 :: (v_0, z_0, u) :: \pi_0\} \tag{6.78}$$

So let π' be the unique path in $\rho(\pi)$ and consider its l-context assignment.

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: (v_0, z_0, u) :: \pi_1) \tag{6.79}$$

$$= \mathfrak{s}(\pi_0) \cdot 1 \cdot !_n a \cdot \mathfrak{s}(\pi_1) \tag{6.80}$$

$$= \mathfrak{s}(\pi) \tag{6.81}$$

5. Rule (d?) and $n = 1$. Let R be as in the right of Figure 5.5d, and let $\chi_1 = (u_i, w, v_0)$, for some $0 \leq i \leq h$. Also, if l is the $|+\rangle$-link in R, let n be its level, and let $a = \text{VarO}(l)$ for some $a \in \mathfrak{S}^*$. Finally, let $m > n$ be the level of the ?-link. By definition of l-context assignment,

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0 :: (u_i, w, v_0) :: \pi_1) \tag{6.82}$$

$$= (\mathfrak{s}(\pi_0))|_m \cdot !_n a \cdot \mathfrak{s}(\pi_1) \tag{6.83}$$

By definition of residual, and in particular by (6.15):

$$\rho(\pi) = \rho(\pi_0 :: (u_i, w, v_0) :: \pi_1) \tag{6.84}$$

$$= \{\pi_0 :: (u_i, z_{0_i}, v_0) :: \pi_0\} \tag{6.85}$$

So let π' be the unique path in $\rho(\pi)$ and consider its l-context assignment, and observe it is equivalent by definition of the restriction operator.

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: (u_i, z_{0_i}, v_0) :: \pi_1) \tag{6.86}$$

$$= (\mathfrak{s}(\pi_0) \cdot !_n a)|_m \cdot \mathfrak{s}(\pi_1) \tag{6.87}$$

$$= \mathfrak{s}(\pi) \tag{6.88}$$

6. Rule (d $\multimap$) and $1 \leq n \leq 2$. Let R be as in Figure 5.5a, and if h is the lift within R, then let $a = \text{VarO}(h)$ and let m be its level.

   (a) Suppose that π contains only the maximal crossing (v_0, w, u_2). By definition of l-context assignment we have that

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0 :: (v_0, w, u_2) :: \pi_1)) \tag{6.89}$$

$$= \mathfrak{s}(\pi_0 :: (v_0, w) :: (w, u_2) :: \pi_1) \tag{6.90}$$

$$= \mathfrak{s}(\pi_0) \cdot !_m a \cdot 1 \cdot \mathfrak{s}(\pi_1). \tag{6.91}$$

   Now, by definition of downward path residual, in particular (6.10),

$$\rho(\pi) = \rho(\pi_0 :: (v_0, w, u_2) :: \pi_1) \tag{6.92}$$

$$= \{\pi_0 :: (v_0, z_{2_0}, u_2) :: \pi_1\}, \tag{6.93}$$

So if we consider the l-context of $\pi' \in \rho(\pi)$, we easily obtain that is equal to that of π.

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: (\nu_0, z_{2_0}, u_2) :: \pi_1) \tag{6.94}$$

$$= \mathfrak{s}(\pi_0) \cdot \mathfrak{s}((\nu_0, z_{2_0}) \cdot \mathfrak{s}((z_{2_0}, u_2)) \cdot \mathfrak{s}(\pi_1) \tag{6.95}$$

$$= \mathfrak{s}(\pi_0) \cdot 1 \cdot !_m a \cdot \mathfrak{s}(\pi_1) \tag{6.96}$$

$$= \mathfrak{s}(\pi). \tag{6.97}$$

(b) Assume that π crosses R only in (u_1). By definition of l-context assignment, and of residual we have:

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0 :: (u_1)) = \mathfrak{s}(\pi_0), \tag{6.98}$$

$$\rho(\pi) = \rho(\pi_0 :: (u_1)) = \{\pi_0 :: (u_1)\}. \tag{6.99}$$

So, we immediately conclude: let $\pi' \in \rho(\pi)$ and consider its context.

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: (u_1)) = \mathfrak{s}(\pi). \tag{6.100}$$

(c) Otherwise π crosses R twice, and it must be the case that $\chi_1 = (\nu_0, w, u_2)$, while $\chi_2 = (u_1)$. From previous cases 6a and 6b we have: $\mathfrak{s}(\chi_1) = \mathfrak{s}(\chi_1')$ for any $\chi_1' \in \rho(\chi_1)$, and $\mathfrak{s}(\chi_2) = \mathfrak{s}(\chi_2')$ for any $\chi_2' \in \rho(\chi_2)$. Hence the claim.

7. Rule $(\text{d}\multimap)$ and $n = 1$. Let R be as in the rightmost part of Figure 5.5b, and if h is the lift within R, then let $a = \text{VarO}(h)$ and $m = \ell(h)$. We distinguish two cases, according to the two possible maximal crossing of R.

(a) If $\chi_1 = (u_2, w, \nu_0)$, then by definition of residual, and in particular by (6.12),

$$\rho(\pi) = \rho(\pi_0 :: (u_2, w, \nu_0) :: \pi_1) \tag{6.101}$$

$$= \{\pi_0 :: (u_2, z_{2_0}, \nu_1) :: \pi_1\} \tag{6.102}$$

Let $\pi' \in \rho(\pi)$. By simply inspecting the definition, we immediately verify that the l-context is unvaried.

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0) \cdot 1 \cdot !_m \overline{a} \cdot \mathfrak{s}(\pi_1) \tag{6.103}$$

$$= \mathfrak{s}(\pi_0) \cdot !_m \overline{a} \cdot \mathfrak{s}(\pi_1) \tag{6.104}$$

$$= \mathfrak{s}(\pi_0) \cdot !_m \overline{a} \cdot 1 \cdot \mathfrak{s}(\pi_1) \tag{6.105}$$

$$= \mathfrak{s}(\pi') \tag{6.106}$$

(b) Otherwise $\chi_1 = (u_2, u_1)$. Its residual is by (6.13),

$$\rho(\pi) = \rho(\pi_0 :: (u_2, u_1) :: \pi_1) \tag{6.107}$$

$$= \{\pi_0 :: (u_2, z_{2_0}, u_{2_0}, u_1) :: \pi_1\}. \tag{6.108}$$

While the l-context of π and that of $\pi' \in \rho(\pi)$ are:

$$\mathfrak{s}(\pi) = \mathfrak{s}(\pi_0) \cdot 1 \cdot \mathfrak{s}(\pi_1) \tag{6.109}$$

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0) \cdot !_m \overline{a} \cdot !_n a \cdot \mathfrak{s}(\pi_1). \tag{6.110}$$

Now, let $m' \geq l$ and consider its equality with respect to m.

i. If $m' \neq m$, then $\mathfrak{s}(\pi)(m) = \mathfrak{s}(\pi')(m)$.

ii. Otherwise, $m' = m$. By IH we have that $\mathfrak{s}\,(\pi_0 :: (u_2, u_1)) \geq_{\ell(v)} 1$, which means that in particular $\mathfrak{s}\,(\pi_0)\,(m) \cdot \overline{a} \geq 1$. Therefore, it must be the case that $\mathfrak{s}\,(\pi_0)\,(m) = b \cdot a$, for some $b \in \mathbb{S}^*$. Substituting it in (6.109) and (6.110), we obtain

$$\mathfrak{s}\,(\pi)\,(m) = b \cdot a \cdot \mathfrak{s}\,(\pi_1)\,(m) \tag{6.111}$$

$$\mathfrak{s}\,(\pi')\,(m) = b \cdot a \cdot \overline{a} \cdot a \cdot \mathfrak{s}\,(\pi_1)\,(m) \tag{6.112}$$

$$= b \cdot a \cdot \mathfrak{s}\,(\pi_1)\,(m). \tag{6.113}$$

8. Rule (r?) and $n = 1$. Let R be as in Figure 5.8b, where h is the ⊢-link and h′ is the ?-link involved R. Let $m = \ell(h)$, $a = \mathsf{VarO}(h)$. We have two kinds of downward crossings, which we analyse separately.

(a) Assume first $\chi_1 = (v_0, u_i, w)$. By definition of residual and in particular by (6.21),

$$\rho(\pi) = \rho(\pi_0 :: (v_0, u_i, w) :: \pi_1) \tag{6.114}$$

$$= \{\pi_0 :: (v_0, y_0, w) :: \pi_1\}. \tag{6.115}$$

Let π' be the unique path in $\rho(\pi)$ and inspect the definition of l-context assignment. We immediately verify it is unvaried.

$$\mathfrak{s}\,(\pi) = \mathfrak{s}\,(\pi_0) \cdot !_m \overline{a} \cdot 1 \cdot \mathfrak{s}\,(\pi_1) \tag{6.116}$$

$$= \mathfrak{s}\,(\pi_0) \cdot 1 \cdot !_m \overline{a} \cdot \mathfrak{s}\,(\pi_1) \tag{6.117}$$

$$= \mathfrak{s}\,(\pi') \tag{6.118}$$

(b) Otherwise, hypothesise that $\chi_1 = (u_j, w)$ where $i \neq j$. By definition and in particular by (6.22):

$$\rho(\pi) = \rho(\pi_0 :: (u_j, w) :: \pi_1) \tag{6.119}$$

$$= \{\pi_0 :: (u_j, z_{j_0}, z_{0_0}, w) :: \pi_1\}. \tag{6.120}$$

Now, let $m' \geq \ell(v)$ and consider its equality with respect to m.

i. If $m' \neq m$, then by definition of the lifting operator we immediately obtain that the l-context is invariant and that IH is consequently preserved.

$$\mathfrak{s}\,(\pi)\,(m') = (\mathfrak{s}\,(\pi_0) \cdot \mathfrak{s}\,(\pi_1))\,(m') \tag{6.121}$$

$$= \mathfrak{s}\,(\pi')\,(m') \tag{6.122}$$

ii. Otherwise, we have that:

$$\mathfrak{s}\,(\pi)\,(m) = (\mathfrak{s}\,(\pi_0) \cdot \mathfrak{s}\,(\pi_1))(m), \tag{6.123}$$

$$\mathfrak{s}\,(\pi')\,(m) = (\mathfrak{s}\,(\pi_0) \cdot !_m \overline{a} \cdot !_m a \cdot \mathfrak{s}\,(\pi_1))(m). \tag{6.124}$$

Let π_0' be a downward path whose first vertex is the same as that of π_0, and whose last vertex is the premiss u_i of h. Its existence is guaranteed by Lemma 6.6. As observed in previous case 8a, let $\mathfrak{s}\,(\pi_0')\,(m) = \alpha \cdot \overline{a}$, for some $\alpha \in \mathbb{S}^*$. By Lemma 6.11, we know that $\mathfrak{s}\,(\pi_0) = _{\ell(v)}^{\ell(h')} \mathfrak{s}\,(\pi_0')$, therefore we obtain the claim:

$$\mathfrak{s}\,(\pi')\,(m) = \alpha \cdot a \cdot \overline{a} \cdot a \cdot \mathfrak{s}\,(\pi_1) \tag{6.125}$$

$$= \alpha \cdot a \cdot \mathfrak{s}\,(\pi_1)\,(m) \tag{6.126}$$

$$= \mathfrak{s}\,(\pi)\,(m). \tag{6.127}$$

9. Rule ($\mathsf{r}\!\!-\!\!\circ$). Omitted: an inspection of definitions shows that for each of the two possible downward crossing of R the situation is identical to what is described in case 4 (rule (d!)).

10. Rule (s) and $n = 1$. Let R be as in the right of Figure 5.6b, and $\chi_1 = (u_0, w, v_0)$. Also, if l (resp. l') is the $\vert+$)-link (resp. $\vert-$)) in R, let n (resp. m) be its level, and let $a = \mathsf{VarO}(l)$ for some $a \in \mathcal{S}^*$ (resp. b). By definition of l-context assignment,

$$\mathfrak{s}\left(\pi\right) = \mathfrak{s}\left(\pi_0 :: (u_0, w, v_0) :: \pi_1\right) \tag{6.128}$$
$$= \mathfrak{s}\left(\pi_0\right) \cdot !_n a \cdot !_m \overline{b} \cdot \mathfrak{s}\left(\pi_1\right) \tag{6.129}$$

By definition of residual, and in particular by (6.17):

$$\rho(\pi) = \rho(\pi_0 :: (u_0, w, v_0) :: \pi_1) \tag{6.130}$$
$$= \{\pi_0 :: (u_0, z_{0_0}, v_0) :: \pi_0\} \tag{6.131}$$

So let π' be the unique path in $\rho(\pi)$. If we consider its l-context assignment, we easily conclude by definition of lifting:

$$\mathfrak{s}\left(\pi'\right) = \mathfrak{s}\left(\pi_0 :: (u_0, z_{0_0}, v_0) :: \pi_1\right) \tag{6.132}$$
$$= \mathfrak{s}\left(\pi_0\right) \cdot !_m \overline{b} \cdot !_n a \cdot \mathfrak{s}\left(\pi_1\right) \tag{6.133}$$
$$= \mathfrak{s}\left(\pi\right). \tag{6.134}$$

11. Rule (a) and $n = 1$. Let R be as in left of Figure 5.6a. If h, h' are the lifts within R, then let: $m = \ell(h) = \ell(h')$, $a = \mathsf{VarO}(h)$, and $a' = \mathsf{VarO}(h')$. By definition of l-context assignment,

$$\mathfrak{s}\left(\pi\right) = \mathfrak{s}\left(\pi_0 :: (u_0, w, v_0) :: \pi_1\right) \tag{6.135}$$
$$= \mathfrak{s}\left(\pi_0\right) \cdot !_m a \cdot !_m \overline{a'} \cdot \mathfrak{s}\left(\pi_1\right). \tag{6.136}$$

But as per Proposition 6.1 we know that $\mathfrak{s}\left(\pi\right) \geq_{\ell(v)} 1$. Therefore, it must be the case that $\mathfrak{s}\left(\pi_0\right) \geq_{\ell(v)} 1$ and $a \cdot \overline{a'} = 1$, which means that $a = a'$. Hence,

$$= \mathfrak{s}\left(\pi_0\right) :: \mathfrak{s}\left(\pi_1\right). \tag{6.137}$$

Now, by definition of residual (cf. (6.16)), we have that

$$\rho(\pi) = \rho(\pi_0 :: (u_0, w, v_0) :: \pi_1) \tag{6.138}$$
$$= \{\pi_0 :: (u_0 = v_0) :: \pi_1\}. \tag{6.139}$$

Let $\pi' \in \rho(\pi)$ and observe that l-context of π' is equivalent to that of π.

$$\mathfrak{s}\left(\pi'\right) = \mathfrak{s}\left(\pi_0\right) :: \mathfrak{s}\left(\pi_1\right). \tag{6.140}$$

12. Rule (m) and $n = 1$. Assume R as in the right of Figure 5.8a, and let $\chi_1 = (v_0, u_i, w)$, for some $0 \leq i \leq h$. Also, let n be the level of the ?-link and of the $\vert+$)-link in R, say l. Finally, let $a = \mathsf{VarO}(l)$ for some $a \in \mathcal{S}^*$. By definition of l-context assignment, and by definition of the restriction operator:

$$\mathfrak{s}\left(\pi\right) = \mathfrak{s}\left(\pi_0 :: (v_0, u_i, w) :: \pi_1\right) \tag{6.141}$$
$$= \left(\mathfrak{s}\left(\pi_0\right) \cdot !_n a\right)\big|_n \cdot \mathfrak{s}\left(\pi_1\right) \tag{6.142}$$
$$= \left(\mathfrak{s}\left(\pi_0\right)\right)\big|_n \cdot \mathfrak{s}\left(\pi_1\right). \tag{6.143}$$

By definition of residual, and in particular by (6.23):

$$\rho(\pi) = \rho(\pi_0 :: (v_0, u_i, w) :: \pi_1) \tag{6.144}$$

$$= \{\pi_0 :: (v_0, w) :: \pi_0\}. \tag{6.145}$$

Consider π' is the unique path in $\rho(\pi)$, and look at its l-context assignment. We conclude.

$$\mathfrak{s}(\pi') = \mathfrak{s}(\pi_0 :: (v_0, w) :: \pi_1) \tag{6.146}$$

$$= (\mathfrak{s}(\pi_0) \cdot !_n a)|_m \cdot \mathfrak{s}(\pi_1) \tag{6.147}$$

$$= \mathfrak{s}(\pi) \tag{6.148}$$

∎

Proof of Proposition 6.1 (Positivity). Let $\mathcal{U}$ be a UG-graph, and $\pi : a \gg b$ be a path such that a is a !-premiss or the root of $\mathcal{U}$. If $\mathcal{U}$ is a proof-net, then the claim holds trivially, since $\mathfrak{s}(\pi) = 1 \geq 1$. So suppose otherwise that $\mathcal{U} = \rho(\mathcal{U}')$, for some UG-graph $\mathcal{U}'$ and some UG- or URB-reduction step ρ. Also, let R be the redex of ρ, and let $\rho(R)$ be the residuals of R.

1. If a, b do not belong to int(R), then there exists $\pi' : a \gg b$ path of $\mathcal{U}$ that is long enough for R and such that $\pi = \rho(\pi')$. Now a is a !-premiss or the root of $\mathcal{U}'$, so per IH $\mathfrak{s}(\pi') \geq_{\ell(a)} 1$. But by invariance Lemma 6.9 we have $\mathfrak{s}(\pi') = \mathfrak{s}(\pi)$, hence $\mathfrak{s}(\pi) \geq_{\ell(a)} 1$.

2. If a or b belong to int(R), we distinguish some cases depending on the rule of R. When needed to avoid confusion between paths, we shall precise as a subscript the graph to which they belong, e.g. $\pi_{\mathcal{U}}$ or $\pi'_{\mathcal{U}'}$.

   a. Rules $(!), (\multimap), (a), (m)$. Absurd: by inspection of the definition we verify that int(R) = ∅.

   b. Rule (d!). Let R be as in Figure 5.5c, and observe that int(R) = $\{z_0\}$.

      i. If $a = z_0$, then $\pi = (z_0, u)_{\mathcal{U}} :: \eta$ for some downward path η. We notice immediately that $\mathfrak{s}((z_0, u)_{\mathcal{U}'}) \geq 1$, by definition of contexts, as well as $\mathfrak{s}(\eta) \geq_{\ell(a)} 1$, by inductive hypothesis. Hence $\mathfrak{s}(\pi) \geq_{\ell(a)} 1$.

      ii. If $b = z_0$, then $\pi = \eta :: (v_0, z_0)_{\mathcal{U}}$ for some downward path η. But $\mathfrak{s}(\eta) \geq_{\ell(z_0)} 1$, while $\mathfrak{s}((v_0, z_0)) = 1 \geq 1$. Therefore, $\mathfrak{s}(\pi) \geq_{\ell(a)} 1$.

   c. Rule (d?). Let R be as in Figure 5.5d, and observe that int(R) = $\{z_{0_0}, \ldots, z_{0_k}\}$, so let us consider z_{0_i} for some $0 \leq i \leq k$.

      i. If $a = z_{0_i}$, then we absurdly contradict the hypothesis of a being a !-premiss or the root of $\mathcal{U}$.

      ii. If $b = z_{0_i}$, then $\pi = \eta :: (u_i, z_{0_i})_{\mathcal{U}}$ for some downward path η. Inspecting ρ we observe that $\mathfrak{s}((u_i, z_{0_i})_{\mathcal{U}}) = \mathfrak{s}((u_i, w, v_0)_{\mathcal{U}'})$. Also, by IH we have that $\mathfrak{s}(\eta :: (u_i, w, v_0)) \geq_{\ell(a)} 1$, hence $\mathfrak{s}(\pi) \geq_{\ell(a)} 1$.

   d. Rules (d $\multimap$). Let R be as in Figure 5.5a, and notice that int(R) = $\{z_{1_0}, z_{2_0}\}$. Also, since $\rho(R)$ does not include a !-link, $a \notin$ int(R).
      Now, if $b = z_{2_0}$, then the situation is exactly as in the previously discussed case 2.b.ii. So assume $b = z_{1_0}$, and let $\chi_1 = (v_0, x_{2_0}, u_2)$, and $\chi_2 = (u_1, z_{1_0})$. Clearly, χ_2 is a suffix of π. What about χ_1?

i. If $\chi_1 \not\subset \pi$, then observe that $\ell(a) > \ell(z_{1_0})$. Let $\pi = \pi_0 :: \chi_2$, Now, by definition $\mathfrak{s}\left((u_1, z_{1_0})\right) =_{\ell(a)} 1$, and $\pi_0 \mathcal{U}' \geq_{\ell(a)} 1$ per inductive hypothesis. Hence the claim.

ii. If $\chi_1 \subset \pi$, then let $\pi = \pi_0 :: \chi_1 :: \pi_1 :: \chi_2$, and consider $\pi' = \pi_0 \mathcal{U}' :: \chi_1' :: \pi_{1\mathcal{U}'}$, where $\chi_1' = (v_0, w, u_2)$. Also, let n be the level of the $|*)$-links crossed in χ_1, χ_2, and consider the l-context of π at level $m \in \mathbb{N}$.

A. If $m \neq n$, as in last sub-case we immediately obtain that $\mathfrak{s}(\pi)(m) = \mathfrak{s}(\pi')(m)$. But by IH $\mathfrak{s}(\pi')(m) \geq 1$, thus $\mathfrak{s}(\pi)(m) \geq 1$.

B. Otherwise $m = n$. In $\mathcal{U}'$ we know by IH that $\mathfrak{s}(\pi_0 :: \chi_1') =_{\ell(a)} \mathfrak{s}(\pi_0 :: \chi_1' :: \pi_1)$, hence $\mathfrak{s}(\pi_0 :: \chi_1')(n) = \mathfrak{s}(\pi_0 :: \chi_1' :: \pi_1)(n)$. Because of this, we can apply Lemma 6.8, to obtain that, for any $m' \in \mathbb{N}$ (hence in particular for $m' = m$), there exist $c, d \in \mathfrak{S}^*$ such that $\mathfrak{s}(\pi_0 :: \chi_1')(m') = c \cdot d$ and $\mathfrak{s}(\pi_1)(m') = \overline{d} \cdot d$, where d is positive.

Let $e = \mathfrak{s}(\chi_1')(m)$, and observe that $\mathfrak{s}(\chi_1) = \mathfrak{s}(\chi_1')$. Also, notice that e has to be the rightmost object of the stable form of $\mathfrak{s}(\pi_0 :: \chi_1')(m')$.

I. If $d = 1$, then $\mathfrak{s}(\pi_0 :: \chi_1')(m') = c' \cdot e$, and $\mathfrak{s}(\pi_1)(m') = 1$. So we can write the l-context of π at n as follows.

$$\mathfrak{s}(\pi)(n) = \mathfrak{s}(\pi_0 :: \chi_1 :: \pi_1 :: \chi_2) \tag{6.149}$$
$$= \mathfrak{s}(\pi_0 :: \chi_1)(n) \cdot \mathfrak{s}(\pi_1)(n) \cdot \mathfrak{s}(\chi_2)(n) \tag{6.150}$$
$$= c' \cdot e \cdot 1 \cdot \overline{e} \tag{6.151}$$
$$= c' \tag{6.152}$$
$$= \mathfrak{s}(\pi_0). \tag{6.153}$$

II. If $d \neq 1$, then $\mathfrak{s}(\pi_0 :: \chi_1')(m') = c \cdot d' \cdot e$. and $\mathfrak{s}(\pi_1)(m') = \overline{e} \cdot \overline{d'} \cdot d' \cdot e$. Therefore, the l-context at level n of π is:

$$\mathfrak{s}(\pi)(n) = \mathfrak{s}(\pi_0 :: \chi_1 :: \pi_1 :: \chi_2) \tag{6.154}$$
$$= \mathfrak{s}(\pi_0 :: \chi_1)(n) \cdot \mathfrak{s}(\pi_1)(n) \cdot \mathfrak{s}(\chi_2)(n) \tag{6.155}$$
$$= c \cdot d' \cdot e \cdot \overline{d' \cdot e} \cdot d' \cdot e \cdot \overline{e} \tag{6.156}$$
$$= c \cdot d' \cdot e \cdot \overline{e} \cdot \overline{d'} \cdot d' \cdot e \cdot \overline{e} \tag{6.157}$$
$$= c' \cdot d' \cdot e \tag{6.158}$$
$$= \mathfrak{s}(\pi_0). \tag{6.159}$$

But by IH $\mathfrak{s}(\pi_0)(n) \geq 1$, thus $\mathfrak{s}(\pi)(n) \geq 1$.

Therefore the claim.

e. Rules $(d\overset{-}{\multimap}), (r?), (r\overset{-}{\multimap}), (s)$. Since $\rho(R)$ does not contain an !-link, $a \notin \operatorname{int}(R)$, so it must be the case that $b \in \operatorname{int}(R)$. Then we can follow both arguments used in cases 2.b.ii and 2.c.ii.

f. Rule (tD). Let R be as in Figure 5.9, and observe that in this case we have $\operatorname{int}(R) = \bigcup_{0 \leq i \leq h} V(b_i)$, where we recall that $V(b)$ is the set of vertices of the i-th copy of the box b.

i. If $a, b \in \bigcup_{0 \leq i \leq h} V(b_i)$, then observe that any vertex of π has a unique anti-residual in $\mathcal{U}'$. So let $\pi' : a' \gg b'$ be the path built by these anti-residuals following the same order of π. Unsurprisingly, $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$. But $\mathfrak{s}(\pi') \geq_{\ell(a')} 1$ by IH, and $\ell(a') = \ell(a)$, therefore $\mathfrak{s}(\pi) \geq_{\ell(a)} 1$.

ii. If $a \in \bigcup_{0 \le i \le h} V(b_{j_i}) \not\ni b$, then let $\pi = \gamma :: (U_{j_i}, u_j) :: \eta$, for some $0 \le i \le h$ and $1 \le j \le k$. Now, consider $\pi' = \gamma :: (U_{j_i}, u_j) :: \eta$. Similarly to the previous case 2.f.i, we have $\mathfrak{s}(\pi') = \mathfrak{s}(\pi)$, and $\mathfrak{s}(\pi') \ge_{\ell(a)} 1$, hence the claim.

iii. If $a \notin \bigcup_{0 \le i \le h} V(b_i) \ni b$, then let $\pi_{\mathcal{U}} = \pi_0 :: (v_j, w_j) :: \gamma_j$ for some $1 \le j \le k$, and consider $\pi'_{\mathcal{U}'} = \pi_0 :: (v_j, z, w) :: \gamma$. Now, firstly observe that $\mathfrak{s}(\pi_{0\mathcal{U}}) = \mathfrak{s}(\pi_{0\mathcal{U}'})$, which is positive by IH. Secondly, we have $\mathfrak{s}((v_j, z, w)) \ge 1$ by definition. Finally, we notice that $\mathfrak{s}(\gamma_j) = \mathfrak{s}(\gamma)$, which is positive by IH. (More details are comprehensively explained in case 3 of the proof of Lemma 6.9.) Therefore π has a positive weight.

∎

Proof of Lemma 6.10 (–∘-compatibility). Let $\mathcal{U}$ be a UG-graph, $\pi : a \gg c$ and $\phi : c \gg d$ be two paths such that a is a !-premiss or the root of $\mathcal{U}$, and there exists $\langle d, c \, (-\!\circ) \, b \rangle$. If $\mathcal{U}$ is a proof-net, then the claim immediately holds, since $\mathfrak{s}(\pi) = \mathfrak{s}(\phi) = 1 = \mathfrak{s}(\pi :: \phi)$. Suppose otherwise that $\mathcal{U} = \rho(\mathcal{U}')$, for some UG-graph $\mathcal{U}'$ and some UG- or URB-reduction step ρ. Also, let R be the redex of ρ, and let $\rho(R)$ be the residual of R.

1. If a, c, and d do not belong to int(R), then there exist $\pi' : a \gg c$ and $\phi' : c \gg d$ paths of $\mathcal{U}$ that are long enough for R. Now, a is a !-premiss the root of $\mathcal{U}'$, so per IH $\mathfrak{s}(\pi') =_{\ell(a)} \mathfrak{s}(\pi' :: \phi')$. But by invariance Lemma 6.9, $\mathfrak{s}(\pi') = \mathfrak{s}(\pi)$ and $\mathfrak{s}(\phi') = \mathfrak{s}(\phi)$. Hence $\mathfrak{s}(\pi) \ge_{\ell(a)} 1$.

2. If a, c, or d belongs to int(R), we distinguish some cases depending on the rule of R. Subscripts of paths denotes the graph to which they belong, e.g. $\pi_{\mathcal{U}}, \pi'_{\mathcal{U}'}$.

   a. Rules $(!), (-\!\circ), (a), (m)$. Absurd: by inspection of the definition we verify that int(R) = ∅.

   b. Rules $(d?), (d\bar{-}\!\circ), (r\bar{-}\!\circ), (r?), (s)$. Then $\rho(R)$ does not contain any link of kind ! or $-\!\circ$, so a, c, and d cannot be in int(R). Absurd.

   c. Rule $(d!)$. Let R be as in Figure 5.5c, and observe that int(R) = $\{z_0\}$. Since we assumed c, d being respectively the second and the first premiss of a $-\!\circ$-link, it must be the case that $c, d \ne z_0$, which implies that $a = z_0$. So, let $\pi = (z_0, u) :: \eta$ for some downward path η. By IH, we know that $\mathfrak{s}(\eta) =_{\ell(a)} \mathfrak{s}(\eta :: \phi)$. Hence $\mathfrak{s}(\pi) =_{\ell(a)} \mathfrak{s}(\pi :: \phi)$.

   d. Rule $(d -\!\circ)$. Let R be as in Figure 5.5a. We preliminarily notice that int(R) = $\{z_{1_0}, z_{2_0}\}$. Also, since $\rho(R)$ does not include a !-link, $a \notin$ int(R). Moreover, it must the case that both $c, d \in$ int(R). because $\rho(R)$ contains both the premisses of a $-\!\circ$-link. In particular, this mean that $c = z_{2_0}, d = z_{1_0}$. Then the claim has been proven in the proof of Proposition 6.1, case 2.d.ii (in particular Equation 6.153 and 6.159).

   e. Rule (tD). Let R be as in Figure 5.9, where int(R) = $\bigcup_{0 \le i \le h} V(\mathcal{B}_i)$. By hypothesis on c, d we have that c belongs to int(R) if and only if d does so.

      i. If $a, c, d \in \bigcup_{0 \le i \le h} V(b_i)$, then observe that any vertex of π has a unique anti-residual in $\mathcal{U}'$. So let $\pi' : a' \gg c'$ and $\phi' : c' \gg d'$ be the path built by these anti-residuals following the same order of π and ϕ, respectively.

Then $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$ and $\mathfrak{s}(\phi) = \mathfrak{s}(\phi')$. By IH we have $\mathfrak{s}(\pi') =_a \mathfrak{s}(\pi' :: \phi')$, therefore $\mathfrak{s}(\pi) =_a \mathfrak{s}(\pi :: \phi)$.

  ii. If $a \in \bigcup_{0 \leq i \leq h} V(b_i) \not\ni c, d$, then let $\pi = \gamma :: (U_{j_i}, u_j) :: \eta$ for some $0 \leq i \leq h$ and $1 \leq j \leq k$. Now, consider $\pi' = \gamma :: (U_{j_i}, u_j) :: \eta$, and observe that $\mathfrak{s}(\pi') = \mathfrak{s}(\pi)$. Now, by IH we know $\mathfrak{s}(\pi') =_a \mathfrak{s}(\pi' :: \phi)$, so we conclude that $\mathfrak{s}(\pi) =_a \mathfrak{s}(\pi :: \phi)$, quod erat demonstrandum.

  iii. If $a \notin \bigcup_{0 \leq i \leq h} V(b_i) \ni c, d$, then let $\pi = \pi_0 :: (v_j, w_j)\mathcal{U} :: \gamma_j$, for some $1 \leq j \leq k$. Now, consider $\pi' = \pi_0 :: (v_j, z, w)\mathcal{U}' :: \gamma$. Let m be the level of the !-link in R. Now, firstly observe that $\mathfrak{s}(\pi_0\mathcal{U}) = \mathfrak{s}(\pi_0\mathcal{U}')$ and $\mathfrak{s}(\gamma_j) = \mathfrak{s}(\gamma)$. Moreover, by definition of l-context assignment, $\mathfrak{s}(\pi_0)(m) = 1$. Therefore $\mathfrak{s}(\pi')(m) = \mathfrak{s}(\gamma)(m)$. But $\mathfrak{s}(\pi')(m) = \mathfrak{s}(\pi' :: \phi)(m)$, thus $\mathfrak{s}(\gamma)(m) = \mathfrak{s}(\gamma :: \phi)(m)$. Secondly, we have $\mathfrak{s}((v_j, w_j)) \geq 1$ per definition. Thirdly, applying positivity Proposition 6.1 as IH, we obtain that $\mathfrak{s}(\gamma) \geq_{\ell(a)} 1$, hence $\mathfrak{s}(\gamma_j) \geq_{\ell(a)} 1$. Hence the claim. (See also case 3 of the proof of Lemma 6.9.)

$\blacksquare$

Proof sketch of Lemma 6.11 (Box-compatibility). To avoid pedantry, we provide only an outline of the argumentation, since it is very similar to that of Lemma 6.10. One goes by induction on the length of the reduction sequence from a proof-net, where the statement holds trivially, to the given UG-graph. Given a step ρ, Lemma 6.9 allows to obtain the claim by IH in the case of paths whose counter-image with respect to ρ is long enough for the redex of ρ. Otherwise, one proceeds with an inspection of possible redexes whose residuals are crossed by the paths $\pi : u \gg v$ and $\pi' : u \gg v'$ of interest.

- One interesting case is that of the (tD). Its behaviour is irrelevant at levels strictly smaller than $\ell(u)$ and greater or than $\ell(v)$, which are out of the statements' scope. In between such interval, the rule adds a lift in front of one of the copies of a box $\mathcal{B}$, which carry a $\mathfrak{S}^*$ object to the paths crossing it. It is sufficient to observe that for any box $\mathcal{B}'$ at level strictly smaller than $\ell(v)$, $v \in \mathcal{B}'$ if and only if $v' \in \mathcal{B}'$ (such property comes from a crucial feature of mIELL: a ?-premiss belongs at most to one box); and to recall the box connectivity Lemma 6.6. This allow to observe that any new lift generated by a (tD)- step on $\mathcal{B}$ necessarily introduces a $\mathfrak{S}^*$ object which belongs to both the contexts of π, π'

- The only other notable case is $(r?)$. Assume it happens on v and involve a lift having variable occurrence c at level n, which therefore is the only one to be considered. By IH, both the n-th contexts of the anti-residuals of v, v' have suffix c. Therefore while c is not the suffix of the n-th l-context of v, the other premiss v' has $c \cdot c'$, so they are equivalent.

$\blacksquare$

Proof sketch of Lemma 6.12 (Unary contraction). Let $\pi : u \gg v$ be a downward path in a UG-graph $\mathcal{U}$, where $\mathcal{U}$ is a !-premiss, v is a ?-premiss, and both belong to a box $\mathcal{B}$. If $\mathcal{U}$ is a proof-net, then the claim holds trivially, since $\mathfrak{s}(\pi) = 1 \geq 1$. So suppose otherwise

that $\mathcal{U} = \rho(\mathcal{U}')$, for some UG-graph $\mathcal{U}'$ and some UG- or URB-reduction step ρ. Let R be the redex of ρ, and $\rho(R)$ its residual.

1. If u and v do not belong to $\text{int}(R)$, then there exists $\pi' : u \gg v$ path of $\mathcal{U}$ that is long enough for R and such that $\pi = \rho(\pi')$. Now, a is a !-premiss or the root of $\mathcal{U}'$, so per IH $\mathfrak{s}(\pi) = 1$. But by invariance Lemma 6.9 we have $\mathfrak{s}(\pi') = \mathfrak{s}(\pi)$, and in particular $\mathfrak{s}(\pi')(\ell(u)) = \mathfrak{s}(\pi)(\ell(u))$ hence $\mathfrak{s}(\pi')(\ell(u)) = 1$.

2. Otherwise u or v belongs to $\text{int}(R)$.

   (a) Rules $(!), (\multimap), (d \multimap), (d \overset{=}{\multimap}), (a), (s), (r \overset{=}{\multimap})$. Absurd: by inspection of the definition we verify that $\text{int}(R) = \varnothing$.

   (b) Rules $(d!)$. Let $\pi = \chi :: \eta$, and $\pi' = \chi' :: \eta$ such that π' is long enough for R and $\rho(\pi') = \pi$.
   Now, by definition of UG-graph, the level of the involved lift has to be strictly smaller than n. Therefore $\mathfrak{s}(\chi')(\ell(u)) = \mathfrak{s}(\chi)(\ell(u))$. But by IH $\mathfrak{s}(\eta)(\ell(u))$, hence we conclude.

   (c) Rules $(d?), (r?)$. These cases are duals of 2b, and omitted.

   (d) Rule (m). Let $\pi = \eta :: \chi$, and $\pi' = \eta :: \chi'$ such that π' is long enough for R and $\rho(\pi') = \pi$. Consider the $|+)$-link l in R, and observe that by definition, $\mathfrak{s}(\chi')(\ell(u)) = \text{VarO}(l) \neq 1$. Hence $\mathfrak{s}(\pi')(\ell(u)) =\neq 1$. But this is the negation of IH. Absurd.

   (e) Rule (tD). Let R be as in Figure 5.9, and say the box of R is named C, and the copies of C are named $C_0 \ldots C_h$, for some $h > 0$. In this case $\text{int}(R) = \bigcup_{0 \leq i \leq h} V(C_i)$. We distinguish three cases depending on the level n of the box C.

      i. If $n > \ell(u)$, then in $\mathcal{U}$ we have that $C_i \subset \mathcal{B}$ for any $0 \leq i \leq h$. This implies that there exists a downward path π' in $\mathcal{U}'$ such that $\rho(\pi') = \pi$. But this contradicts our assumption that u or v belong to $\text{int}(R)$. Absurd.

      ii. If $n = \ell(u)$, then there exists $0 \leq i \leq h$ such that $C_h = \mathcal{B}$. Look again at Figure 5.9 and observe that $v = U_{j_i}$ for some $1 \leq j \leq k$, that is the residual of the ?-premiss U_j in $\mathcal{U}'$. But U_j has a total of $h + 1$ residual vertices in $\mathcal{U}$, all being premiss of the same ?-link l. But this contradicts our hypothesis that v is the only premiss of l. Thus there is nothing to be proven here.

      iii. If $n < \ell(u)$, then there exists $0 \leq i \leq h$ such that $C_i \supset \mathcal{B}$. Notice that π is one of the $h + 1$ copies of some path π' of $\mathcal{U}'$, i.e. $\pi \in \rho(\pi)$, for some downward path π of $\mathcal{U}'$. Thus, $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$, from which the claim.

∎

6.4.3 Path irrelevance

We show that the contexts of rooted paths to a given vertex does not depend on the choice of the path, so we can generalise the definition of sharing levelled context as a notion about vertices.

Lemma 6.13 (Path irrelevance). *Let* $\pi : u \gg v$ *and* $\pi' : u \gg v$ *be two downward paths in a UG-graph* $\mathcal{U}$ *where* u *is the root of* $\mathcal{U}$ *or a premiss of a* $!$-*link. Then* $\mathfrak{s}(\pi_1) =_{\ell(u)} \mathfrak{s}(\pi_2)$.

Proof. Let $\mathcal{U}$ be a UG-graph, and let $\phi, \phi' : w \gg v$ be two paths in $\mathcal{U}$. We go by induction on the number e of vertices belonging to both paths.

1. If $e = 1$, then $\phi = \phi'$ and the claim is trivially verified.

2. Otherwise $e > 1$, so let $\phi = \gamma :: \pi$ and $\phi' = \gamma' :: \pi'$ such that: $\gamma, \gamma' : w \gg u$ and $\pi, \pi' : u \gg v$, where $u \neq v$. Since by definition $\mathfrak{s}(\phi) = \mathfrak{s}(\gamma) \cdot \mathfrak{s}(\pi)$ and $\mathfrak{s}(\phi') = \mathfrak{s}(\gamma') \cdot \mathfrak{s}(\pi')$, where per IH we have $\mathfrak{s}(\gamma) = \mathfrak{s}(\gamma')$, we have just reduced the claim to prove $\mathfrak{s}(\pi) = \mathfrak{s}(\pi')$.

   By construction, v has to be the out vertex of a link having at least two in vertices. There are only two links of such kind: $\multimap$ and $?$. But the first premiss of a $\multimap$-link cannot be reached from its second premiss with an upward path. Therefore v is the conclusion of a $?$-link $\mathfrak{l}$.

   Let z, z' respectively be the two premisses of $\mathfrak{l}$ such that $\pi = \delta :: (z, v)$ and $\pi' = \delta' :: (z', v)$. By Lemma 6.11, $\mathfrak{s}(\delta) =_{\ell(u)}^{\ell(w)} \mathfrak{s}(\delta')$. And by definition of $\mathfrak{l}$-context assignment, $\mathfrak{s}(\pi) = (\mathfrak{s}(\delta))|_{\ell(w)}$ and $\mathfrak{s}(\pi') = (\mathfrak{s}(\delta'))|_{\ell(w)}$. Hence the claim.

   ∎

Definition 6.13 (Sharing levelled context of a vertex). The *sharing levelled context of a vertex* $v \in V(\mathcal{U})$, for some UG-graph $\mathcal{U}$ whose root is r, is $\mathfrak{s}(\pi)$ for any downward path $\pi : r \gg v$.

Recall from Definition 2.3 that the interior of a pre-net $\mathcal{P}$ is the set of vertices of $\mathcal{P}$ that are not its conclusions. Now we observe that the $\mathfrak{l}$-context of a given vertex is preserved under reduction if it does not belong to the interior of the involved redex. If otherwise it is not the case, then the $\mathfrak{l}$-context may change. Most notably, this happens for any vertex v belonging to a box $\mathcal{B}$ that is duplicated under a (tD)-step, where any rooted path reaching a copy of v crosses a different, and newly introduced, lift.

Lemma 6.14. *Given a UG-graph* $\mathcal{U}$ *and a reduction step* ρ *on a redex* R, *if* $v \notin \text{int}(R)$, *then* $\mathfrak{s}(v) = \mathfrak{s}(v')$, *for any residual* v' *of* v.

Proof. By hypothesis any downward path π from the root r of $\mathcal{U}$ to v is long enough for R. By definition, there exists $\pi' \in \rho(\pi)$ such that $\pi' : r \gg v'$. Moreover, by long invariance Lemma 6.9, we have $\mathfrak{s}(\pi) = \mathfrak{s}(\pi') = \mathfrak{s}(v')$. ∎

6.5 Unshared cost of reductions

Thanks to sharing contexts, we can identify a set of sub-graphs in a UG graph whose links and vertices are morally waiting for a lift propagation: its share. As anticipated in Section 6.1, we introduce some additional precisions and tools that allow us to transfer $\mathbb{C}_{\text{mIELL}}$ and $\mathbb{C}_{\text{SG}}$ in unshared reductions. More precisely, we define two distinct complexity measures for $\rightarrow_{\text{UG}}$, that are $\widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}$ and $\mathbb{C}_{\text{UG}}^{\text{SG}}$. We show that for any simulation they are indeed equivalent to their counterparts.

6.5.1 Share

Every time that in a UG-graph $\mathcal{U}$ we perform a (tD)-step on a $k + 1$-ary contraction against a box $\mathcal{B}$, we distinguish 1 old copy of $\mathcal{B}$, say $\mathcal{B}_0$ and call it the master copy of $\mathcal{B}$, and k new copies, say $\mathcal{B}_1, \ldots \mathcal{B}_k$, that are shared by the mux introduced in the simulating (t). So, if we call 'master' the 0th lifts and 'sharing' all the others, we are able to discriminate all the vertices of $\mathcal{U}$ whose sharing l-context is neutral or master, from those whose sharing l-context is shared. These former are the share of $\mathcal{U}$, which is partitioned in connected components.

Definition 6.14 (Share and master). A variable occurrence $x_{i:m}$ is called *master* if $i = 0$. Given a lift $\langle u \, (\mid\!\!* \,) \, v \rangle$, if $\mathrm{VarO}(u)$ is master then it is a *master lift*, and it will be drawn in solid black, otherwise it is a *sharing* lift. A context a is *master*, written $a \approx 1$ if $a = 1$, or $a = x_{0:m}$, or $a = \bar{b}$ and $b \approx 1$, or $a = b \cdot c$ and $b, c \approx 1$. Otherwise, we write $a \not\approx 1$ and say that a is *shared*. Similarly, a l-context α is master if $\alpha(n) \not\approx 1$ for any $n \in \mathbb{N}$, otherwise it is shared. A vertex is shared if its l-context is so, otherwise is master; a link is shared if it has at least one shared vertex, otherwise is master.

The *share* of a UG-graph $\mathcal{U}$ is the subgraph $\mathrm{Sh}\,(\mathcal{U}) = (V', L')$, where $V' \subset V(\mathcal{U})$ contains any shared vertex, and $L' \subset L(\mathcal{U})$ contains any shared link. A *share component* is a non-empty, maximal, and connected subset of the share, i.e. $C \subseteq \mathrm{Sh}\,(\mathcal{U})$ such that for any $u \in C$ if $v \in \mathrm{Sh}\,(\mathcal{U})$ is linked to u then $v \in C$. The set of share components of $\mathcal{U}$ is denoted as $\mathrm{ShC}\,(\mathcal{U})$.

Sometimes (tD)-steps acts on subgraphs that overlap with the share, so we further distinguish lifts that get duplicated: intuitively, if a lift l is in the scope of a sharing lift l', then l is 'interior' and the latter is a 'boundary'. Also, we want a way to measure the size of the share ignoring all vertices whose presence is due to lifts, and not to mIELL links. For a sequence of lifts containing a boundary lift, we say all of its vertices are 'boundary', while vertices of other sequences without boundary lifts are 'pseudo-boundary'. All other shared vertices are then those of our primary interest, since they morally resist to lift-erasing. But we may have a strange kind of share components that are made of lifts only, so that none of its vertices are interior; we shall call them 'boundary' as well.

Definition 6.15 (Share boundary and interior). Let $l = \langle u \, (\mid\!\!* \,) \, v \rangle$ in a UG-graph $\mathcal{U}$. l is a *boundary lift* if l is sharing and $u \notin \mathrm{Sh}\,(\mathcal{U}), v \in \mathrm{Sh}\,(\mathcal{U})$, while l is a an *interior lift* when $u, v \in \mathrm{Sh}\,(\mathcal{U})$. The set of lifts of the former kind is denoted as $\mathrm{bdLft}\,(\mathcal{U})$, that of the latter as $\mathrm{intLft}\,(\mathcal{U})$.

A vertex $v \in \mathrm{Sh}\,(\mathcal{U})$ is a *boundary vertex* if there is a boundary lift $\langle u \, (\mid\!\!* \,) \, v \rangle$, or there is a lift $\langle w \, (\mid\!\!* \,) \, v \rangle$ or $\langle v \, (\mid\!\!* \,) \, w \rangle$ such that w is a boundary vertex. A boundary vertex is *boundary-limit* if it is linked to a shared link. The set of boundary vertices is denoted as $\mathrm{bdSh}\,(\mathcal{U})$, its subset of limit boundary vertices as $\mathrm{bdLimSh}\,(\mathcal{U})$. Given $v \in \mathrm{bdLimSh}\,(\mathcal{U})$, the *boundary lift chain* of v is the longest sequence of lifts L that induces a path $v \sim u$, where u is the conclusion of a boundary lift.

If $\mathrm{Sh}\,(\mathcal{U}) \ni v \notin \mathrm{bdSh}\,(\mathcal{U})$ and v is a in-vertex of a lift, then v is a *pseudo-boundary vertex*, the set of which is denoted as $\mathrm{pbSh}\,(\mathcal{U})$. If $\mathrm{Sh}\,(\mathcal{U}) \ni v \notin \mathrm{bdSh}\,(\mathcal{U}) \cup \mathrm{pbSh}\,(\mathcal{U})$, then v is an *interior vertex*, the set of which is $\mathrm{intSh}\,(\mathcal{U})$. A share component having no interior vertices is a *boundary component*, and $\mathrm{bdShC}\,(\mathcal{U})$ denotes the set of such components.

Fact 6.1. *In any UG-graph, the l-context at level n of the first premiss (i.e. the one indexed as 0) of a $?^n$-link is master.*

In any UG-graph the number of boundary limit vertices cannot be greater than the size of the interior share.

Fact 6.2. *In any UG-graph $\mathcal{U}$, $\#(\text{bdLimSh}(\mathcal{U})) \leq \#(\text{intSh}(\mathcal{U}))$.*

Fact 6.3. *Any UGRB-normalisation of a boundary share component include at least one (α)-step.*

6.5.2 Unshared cost of mIELL reduction

We are now able to introduce the notion of cost on UG-reduction which we expect to be equivalent to the $\mathbb{C}_{\text{mIELL}}$ for simulating reductions. The main intuition is that, roughly speaking, in any UG graph $\mathcal{U}$, the size of the internal share corresponds to the number of lift propagations that can be performed, or that must be performed to reach the normal form. Therefore we can make use of the variations in the size of the internal share to distribute the non-constantly bounded cost of a (D)-step over a single (tD) step that grows the share, and a number of $(d\kappa)$- or $(r\kappa)$-steps, for some kind κ, that shrink the share.

Definition 6.16 (mIELL-cost and bdShC-cost of unshared reduction)**.** Given a UG-reduction $\mu : \mathcal{U} \to \mathcal{U}'$, the *partial mIELL-cost* of μ, denoted as $\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu)$, is defined as the difference between: the $\mathbb{C}_{\text{mIELL}}$-cost of ρ such that $\rho \hookleftarrow \mu$, and the variation in the size of interior share caused by μ:

$$\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) = \mathbb{C}_{\text{mIELL}}(\rho) - \Delta\text{intSh}(\mu); \tag{6.160}$$

$$\Delta\text{intSh}(\mu) = \#(\text{intSh}(\mathcal{U}')) - \#(\text{intSh}(\mathcal{U})). \tag{6.161}$$

The *full mIELL-cost* of μ is the sum of its partial cost and the size of internal share of its residual:

$$\widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\mu) = \mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) + \#(\text{intSh}(\mathcal{U})). \tag{6.162}$$

Lemma 6.15 (Correctness of $\widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}$)**.** *Let $\bar{\mu}$ be a UG-reduction $\mathcal{N} \to^* \mathcal{U}'$ and let $\bar{\rho}$ be its mIELL-simulation $\mathcal{N} \to^* \mathcal{N}'$. Then $\mathbb{C}_{\text{mIELL}}(\bar{\rho}) = \widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\bar{\mu})$.*

Proof. Immediate by Definition 6.16. ∎

To allow the comparison of $\mathbb{C}_{\text{UG}}^{\text{mIELL}}$ with respect to $\mathbb{C}_{\text{UG}}^{\text{SG}}$, i.e. the other cost measure of UG reduction that we shall introduce in Subsection 6.5.3, we first carry out a systematic[2] inspection of the former cost measure. In particular, it is worth noticing that, for any given kind of redex, the sum of the variations in the size of the interior share and in the number of boundary share component is constant. Also, we see that the accounting of logical rules ($\multimap$) and ($\S$) is essentially left unchanged with respect to $\mathbb{C}_{\text{mIELL}}$, with a notable exception. If it involves interior shared vertices, then such vertices has to be accounted twice, because morally the reduction need to pay not only for the elimination cost, but also for the unpaid instantiation cost.

---

[2] Tedious.

Table 6.3 Metrics of UG and URB reduction (cf. Lemma 6.16): variation in the size of interior share, mIELL-cost, and variation in the number of boundary share components. Notations: μ is the reduction step, $\mathcal{U}$ is the net containing the redex, $d\kappa, r\kappa$ is a generic duplication rule, $h + 1$ is the number of premises of the ?-link, $\mathcal{B}$ is the box subnet, l, l' are the lifts.

Rule	Proviso	$\Delta\text{intSh}(\mu)$	$\mathbb{C}^{\text{mIELL}}_{\text{UG}}(\mu)$	$\mathbb{C}^{\text{bdShC}}_{\text{UG}}(\mu)$
$(\multimap)$	master	0	9	0
	shared	$-9 + \Delta\text{bdShC}(\mu)$	$18 - \Delta\text{bdShC}(\mu)$	$[0, 2]$
$(!)$	master	0	6	0
	shared	-6	12	$[0, 1]$
$(\text{t}D)$	master	$h \times \#(\mathcal{E}(\mathcal{B})) - h$	$3h + 4$	$\{0, h\}$
	shared	$h \times \#(\mathcal{E}(\mathcal{B})) - 3h - 6$	$5h + 10$	0
$(\text{d}!)$	$l \in \text{bdLft}(\mathcal{U})$	$-3 + \Delta\text{bdShC}(\mu)$	$3 - \Delta\text{bdShC}(\mu)$	$[0, 1]$
$(\text{d}\multimap)$	$l \in \text{bdLft}(\mathcal{U})$	$-5 + \Delta\text{bdShC}(\mu)$	$5 - \Delta\text{bdShC}(\mu)$	$[0, 2]$
$(\text{d}?), (\text{r}?)$	$l \in \text{bdLft}(\mathcal{U})$	$-2h - 3 + \Delta\text{bdShC}(\mu)$	$2h + 3 - \Delta\text{bdShC}(\mu)$	$[0, h + 1]$
$(\text{d}\overset{\bullet}{\multimap}), (\text{r}\overset{\bullet}{\multimap})$	$l \in \text{bdLft}(\mathcal{U})$	$-5 + \Delta\text{bdShC}(\mu)$	$5 - \Delta\text{bdShC}(\mu)$	$[0, 2]$
$(\text{d}\kappa), (\text{r}\kappa)$	$l \notin \text{bdLft}(\mathcal{U})$	0	0	0
(a)	$l, l' \in \text{bdLft}(\mathcal{U})$	0	0	-1
	otherwise	0	0	0
(s)	$l, l' \notin \text{bdLft}(\mathcal{U})$ and $l, l' \in \text{bdLft}(\mu(\mathcal{U}))$	$-1 + \Delta\text{bdShC}(\mu)$	$1 - \Delta\text{bdShC}(\mu)$	$[0, 1]$
	otherwise	0	0	0
(m)	$l \in \text{bdLft}(\mathcal{U})$	0	0	-1
	otherwise	0	0	0

Definition 6.17 (Boundary share component cost). The *boundary-share-components cost* of μ is the absolute value of the variation in the number of boundary share components caused by μ.

$$\mathbb{C}^{\text{bdShC}}_{\text{UG}}(\mu) = |\Delta\text{bdShC}(\mu)|; \tag{6.163}$$

$$\Delta\text{bdShC}(\mu) = \#(\text{bdShC}(\mathcal{U}')) - \#(\text{bdShC}(\mathcal{U})). \tag{6.164}$$

Lemma 6.16 (Metrics on UG-reduction). *Given a UG or a URB reduction step, the possible values of the variation in the size of interior share, of the mIELL-cost, and of the variation in the number of boundary share components and lifts are in Table 6.3.*

Proof. Given a UG-graph $\mathcal{U}$, let R be a redex in $\mathcal{U}$, and let μ be the reduction step on R. We proceed with a case analysis depending on the kind of redex.

1. Rule $(\multimap)$. Let R be as in Figure 5.1.

   (a) If $w \notin \text{Sh}(\mathcal{U})$, then $v_1, v_2, w, u_1, u_2 \notin \text{Sh}(\mathcal{U})$ and $v_1, v_2 \notin \text{Sh}(\mu(\mathcal{U}))$. No boundary share components are changed by μ. Hence, trivially:

   $$\Delta\text{intSh}(\mu) = 0; \tag{6.165}$$

   $$\mathbb{C}^{\text{mIELL}}_{\text{UG}}(\mu) = 9 - 0 = 9; \tag{6.166}$$

   $$\Delta\text{bdShC}(\mu) = 0. \tag{6.167}$$

   $$\tag{6.168}$$

   (b) If $w \in \text{Sh}(\mathcal{U})$, we separately consider three portions of the redex.

      - Consider w and the two links, which all belong to $\text{intSh}(\mathcal{U})$, so they contributes with $\Delta\text{intSh}(\mu) = -1 - 2 \times 3 = -7$.

- Consider u_1, v_1, whose residual is $u_1 = v_1$. Observe that $v_1 \notin \text{pbSh}(\mathcal{U})$. We now distinguish six sub-cases.

    i. If $u_1, v_1 \in \text{intSh}(\mathcal{U})$, then $u_1 \in \text{intSh}(\mu(\mathcal{U}))$. Therefore $\Delta\text{intSh}(\mu) = -1$ and $\Delta\text{bdShC}(\mu) = 0$.

    ii. If $u_1 \in \text{pbSh}(\mathcal{U})$ and $v_1 \in \text{intSh}(\mathcal{U})$, then $u_1 \in \text{pbSh}(\mu(\mathcal{U}))$. Therefore $\Delta\text{intSh}(\mu) = -1$ and $\Delta\text{bdShC}(\mu) = 0$.

    iii. If $u_1 \in \text{intSh}(\mathcal{U})$ and $v_1 \in \text{bdSh}(\mathcal{U})$, then $u_1 \in \text{bdSh}(\mu(\mathcal{U}))$. Hence, $\Delta\text{intSh}(\mu) = -1$ and $\Delta\text{bdShC}(\mu) = 0$.

    iv. If $u_1 \in \text{pbSh}(\mathcal{U})$ and $v_1 \in \text{bdSh}(\mathcal{U})$, then by definition of pseudo-boundary, there exists a path $\pi_i : x_i \gg u_i$ such that $x_i \in \text{intSh}(\mathcal{U})$ while $x_i' \in \text{pbSh}(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. In the residual we have not only $u_1 \in \text{bdSh}(\mu(\mathcal{U}))$, but also $x_i, x_i' \in \text{bdSh}(\mu(\mathcal{U}))$. Hence $\Delta\text{intSh}(\mu) = -1$ and $\Delta\text{bdShC}(\mu) = 0$.

    v. If $u_1 \in \text{bdSh}(\mathcal{U})$ and $v_1 \in \text{bdSh}(\mathcal{U})$, then $u_1 \in \text{bdSh}(\mu(\mathcal{U}))$. Therefore $\Delta\text{intSh}(\mathcal{U}) = 0$. Moreover $v_1 = u_1$ form a new boundary share component, so $\Delta\text{bdShC}(\mu) = 1$.

    vi. If $u_1 \in \text{bdSh}(\mathcal{U})$ and $v_1 \in \text{intSh}(\mathcal{U})$, then $u_1 \in \text{bdSh}(\mathcal{U})$. Thus, $\Delta\text{intSh}(\mathcal{U}) = -1$. and $\Delta\text{bdShC}(\mu) = 0$.

- Consider u_2, v_2, whose residual is $u_2 = v_2$. The analysis is identical to the previous case, where u_2 plays the role of v_1 and v_2 that of u_1.

Summing up, we obtain what follows.

$$\Delta\text{intSh}(\mu) = -9 + \Delta\text{bdShC}(\mu); \tag{6.169}$$

$$\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) = 9 - \left(-9 + \Delta\text{bdShC}(\mu)\right)$$

$$= 18 - \Delta\text{bdShC}(\mu); \tag{6.170}$$

$$\Delta\text{bdShC}(\mu) \in [0, 2]. \tag{6.171}$$

2. Rule (!). Let R be as in Figure 5.4b.

    (a) Assume that $w \notin \text{Sh}(\mathcal{U})$. We immediately notice that $u \notin \text{Sh}(\mathcal{U})$, but we can easily observe also that $v_0 \notin \text{Sh}(\mathcal{U})$. Suppose otherwise that v_0 is shared and let $m \in \mathbb{N}$ such that $\mathfrak{s}(v_0)(m) \not\equiv 1$.

    i. If $m \neq \ell(u)$ then, by definition of context assignment, we would have also $\mathfrak{s}(w)(m) \not\equiv 1$, contradicting our hypothesis.

    ii. Otherwise, $m = \ell(u)$, and let $\mathfrak{s}(v_0) = x_{i:m'}$ with $m' \geq i > 0$. Now, this would absurdly imply that the ?-link of R has at least $i + 1$ premisses, while it has only 1 (i.e. v_0).

    Therefore $v_0 \notin \text{Sh}(\mathcal{U})$, which implies:

$$\Delta\text{intSh}(\mu) = 0; \tag{6.172}$$

$$\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) = 6 - 0 = 6; \tag{6.173}$$

$$\Delta\text{bdShC}(\mu) = 0. \tag{6.174}$$

    (b) Otherwise, $w \in \text{Sh}(\mathcal{U})$ and the analysis follows almost identical to previous sub-case 1b.

$$\Delta\text{intSh}(\mu) = -6; \tag{6.175}$$

$$\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) = 6 - (-6) = 12; \tag{6.176}$$

$$\Delta\text{bdShC}(\mu) \in [0, 1]. \tag{6.177}$$

3. Rule (tD). Let R be as in Figure 5.9.

(a) Suppose first that $z \notin \text{Sh}(\mathcal{U})$.

The vertices $v_0, \ldots, v_h, V_1 \ldots, V_k, u_1, \ldots, u_k$ are irrelevant with respect to the interior share, since none of them cannot belong to $\text{Sh}(\mu(\mathcal{U}))$. The same goes for z, since $w \notin \mu(\mathcal{U})$.

$\mathcal{B}_0$ is irrelevant as well. Indeed, the newly introduced master lift cannot affect $\Delta\text{intSh}(\mu)$ nor $\Delta\text{bdShC}(\mu)$. Moreover, by Fact 6.1 and by our hypothesis on z, we also have $v_0 \notin \text{Sh}(\mathcal{U})$.

Now let $1 \le i \le h$, and let y_i be a vertex in $\mathcal{B}_i \subset \mu(\mathcal{U})$ residual of y in $\mathcal{B} \subset \mathcal{U}$. We observe that $y_i \in \text{intSh}(\mu(\mathcal{U}))$ if $y \ne w$ and $y \in \mathcal{E}(\mathcal{U})$. Therefore:

$$\Delta\text{intSh}(\mu) = h \times (\#(\mathcal{E}(\mathcal{B})) - 1); \qquad (6.178)$$

$$\mathbb{C}_{UG}^{\text{mIELL}}(\mu) = h \times \#(\mathcal{E}(\mathcal{B})) + 2h + 4 - \left(h \times \#(\mathcal{E}(\mathcal{B})) - h\right)$$

$$= 3h + 4. \qquad (6.179)$$

Now let us consider share boundaries. Clearly we have h new lifts of this kind. What about the variation of the number of share components? Notice for any $c \in \text{bdShC}(\mathcal{B})$ and any c_i copy of c, we have c_i is not a boundary share component, but a set of pseudo-boundary share vertices. So the only way for μ to create a new boundary share component involve the new lifts, which are boundary because we assumed $w \notin \text{Sh}(\mathcal{U})$. Since by Proposition 6.1 in $\mathcal{B}$ there cannot be $\langle s \, (\vdash)^n \rangle t \rangle$ such that $w \gg t$, the only other way to have a boundary share component is that $V(\mathcal{B}) = \{w\}$. So if this is case we have $\Delta\text{bdShC}(\mu) = h$, otherwise $\Delta\text{bdShC}(\mu) = 0$:

$$\Delta\text{bdShC}(\mu) = \{0, h\}. \qquad (6.180)$$

(b) Otherwise $z \in \text{Sh}(\mathcal{U})$. We separately discuss subsets of vertices of the redex.

- Consider z and the two main links. By this last hypothesis $z \in \text{intSh}(\mathcal{U})$, while $z \notin \mu(\mathcal{U})$, so in this portion of the redex we have $\Delta\text{intSh}(\mu) = -1 - (h + 2) - 2 = -h - 5$.

- Consider $V(\mathcal{B}) \smallsetminus w$, and let $x \in V(\mathcal{B}) \smallsetminus w$ and $x_i \in V(\mathcal{B}_i) \smallsetminus w_i$ for some $0 \le i \le h$. If $i = 0$ then nothing changes with respect to the share positioning: x belongs to $\text{Sh}(\mathcal{B})$, $\text{intSh}(\mathcal{B})$, $\text{pbSh}(\mathcal{B})$, or $\text{bdSh}(\mathcal{B})$ if and only if x_0 respectively belongs to $\text{Sh}(\mathcal{B}_0)$, $\text{intSh}(\mathcal{B}_0)$, $\text{pbSh}(\mathcal{B}_0)$, or $\text{bdSh}(\mathcal{B}_0)$. The same goes for boundary lifts: $l \in \text{bdLft}(\mathcal{B})$ if and only if $l' \in \mathcal{B}_0$, where l' is the residual of l. If instead $i > 0$, we first observe that if $x \in \text{bdSh}(\mathcal{U})$, then $x_i \in \text{pbSh}(\mu(\mathcal{U}))$, and that any lifts in $L(\mathcal{B}_i)$ are interior. Hence, in both cases, in this portion of the redex we have that $\Delta\text{intSh}(\mu) = h \times (\#(\mathcal{E}(\mathcal{B})) - 1)$, while $\Delta\text{bdShC}(\mu) = 0$.

- Consider w and v_i with $0 \le i \le h$. Since $z \in \text{Sh}(\mathcal{U})$, it must be the case that $w, v_i \in \text{Sh}(\mathcal{U})$. Also, notice that, by definition of pseudo-boundaries, $w \notin \text{pbSh}(\mathcal{U})$, and by Proposition 6.1, $w \notin \text{bdSh}(\mathcal{U})$. Hence $w \in \text{intSh}(\mathcal{U})$.

  i. If $v_i \in \text{bdSh}(\mathcal{U})$, then both $w_i, v_i \in \text{bdSh}(\mu(\mathcal{U}))$, and boundary share components unaffected.

  ii. If $v_i \in \text{intSh}(\mathcal{U})$, then $v_i \in \text{intSh}(\mu(\mathcal{U}))$ and $w_i \in \text{pbSh}(\mu(\mathcal{U}))$, while no change affects boundary share components.

iii. If $v_i \in \text{pbSh}(\mathcal{U})$, then $v_i, w_i \in \text{pbSh}(\mu(\mathcal{U}))$, while no change affects boundary share components.

Hence, here we observe $\Delta\text{intSh}(\mu) = -(h+1)$ and $\Delta\text{bdShC}(\mu) = 0$.

We conclude summing up the variations for the three portions of R so far considered:

$$\Delta\text{intSh}(\mu) = -h - 5 + h \times (\#(\mathcal{E}(\mathcal{B})) - 1) - (h+1)$$
$$= h \times \#(\mathcal{E}(\mathcal{B})) - 3h - 6; \tag{6.181}$$
$$\mathbb{C}^{\text{mIELL}}_{\text{UG}}(\mu) = h \times \#(\mathcal{E}(\mathcal{B})) + 2h + 4 +$$
$$- \left(h \times \#(\mathcal{E}(\mathcal{B})) - 3h - 6\right)$$
$$= 5h + 10; \tag{6.182}$$
$$\Delta\text{bdShC}(\mu) = 0. \tag{6.183}$$

4. Rule (d!). Let R be as in Figure 5.5c, and let $l, \mu(l)$ respectively be the lift of R and its residual. Recall that by definition u cannot belong to $\text{pbSh}(\mathcal{U})$.

   (a) If $l \in \text{bdLft}(\mathcal{U})$, then by definition also $\mu(l) \in \text{bdLft}(\mu)$. Moreover, we have $v_0 \notin \text{Sh}(\mathcal{U})$ and $w \in \text{bdSh}(\mathcal{U})$. We separate two sub-cases about u.

      i. If $u \in \text{intSh}(\mathcal{U})$, then in the reduct we notice that $w \notin \text{Sh}(\mu(\mathcal{U}))$ and $u \in \text{bdSh}(\mu(\mathcal{U}))$. Therefore $\Delta\text{intSh}(\mu) = -1 - 2 = -3$, and $\Delta\text{bdShC}(\mu) = 0$.

      ii. If $u \in \text{bdSh}(\mathcal{U})$, then $w \notin \text{Sh}(\mu(\mathcal{U}))$ and $u \in \text{bdSh}(\mu(\mathcal{U}))$. Observe that u belongs to a new boundary share component of $\mu(\mathcal{U})$. Therefore $\Delta\text{intSh}(\mu) = 0 - 2 = -2$, and $\Delta\text{bdShC}(\mu) = 1$.

   Summarising:

   $$\Delta\text{intSh}(\mu) = -3 + \Delta\text{bdShC}(\mu); \tag{6.184}$$
   $$\mathbb{C}^{\text{mIELL}}_{\text{UG}}(\mu) = 0 - (-3 + \Delta\text{bdShC}(\mu)) = 3 - \Delta\text{bdShC}(\mu); \tag{6.185}$$
   $$\Delta\text{bdShC}(\mu) \in [0, 1]. \tag{6.186}$$

   (b) If $l \notin \text{bdLft}(\mathcal{U})$, then also $\mu(l) \notin \text{bdLft}(\mu(\mathcal{U}))$. Moreover, $w \notin \text{intSh}(\mathcal{U})$. There are then only three sub-cases we need to consider about w.

      i. If $w \notin \text{Sh}(\mathcal{U})$, which means that l is master, then $v_0, u \notin \text{Sh}(\mathcal{U})$. Hence trivially, $v_0, u \notin \text{Sh}(\mu(\mathcal{U}))$.

      ii. If $w \in \text{pbSh}(\mathcal{U})$, then by definition of pseudo-boundary, there exists a path $\pi_i : x_i \gg w$ (where it may be the case that $x_i = v_0$) such that $x_i \in \text{intSh}(\mathcal{U})$ while $x_i' \in \text{pbSh}(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. In the residual we have x_i, x_i' unchanged with respect to the share. Moreover, $w \in \text{intSh}(\mu(\mathcal{U}))$, while $u \in \text{bdSh}(\mu(\mathcal{U}))$.

      iii. If $w \in \text{bdSh}(\mathcal{U})$, then $w \in \text{bdSh}(\mu(\mathcal{U}))$. We need to distinguish two sub-cases about u, which by definition cannot belong to $\text{pbSh}(\mathcal{U})$.

         A. If $u \in \text{intSh}(\mathcal{U})$, then in the residual we observe that $w \in \text{intSh}(\mu(\mathcal{U}))$, while $u \in \text{pbSh}(\mu(\mathcal{U}))$.

         B. If $u \in \text{bdSh}(\mathcal{U})$, then trivially $u \in \text{bdSh}(\mu(\mathcal{U}))$.

   In all the three sub-cases, we accounted no variation in the metrics.

   $$\Delta\text{intSh}(\mu) = 0; \tag{6.187}$$
   $$\mathbb{C}^{\text{mIELL}}_{\text{UG}}(\mu) = 0; \tag{6.188}$$
   $$\Delta\text{bdShC}(\mu) = 0. \tag{6.189}$$

5. Rules $(d \multimap), (d \multimap), (d?), (r \multimap), (r?)$. The analysis of $\Delta \text{intSh}(\mu)$, $\mathbb{C}_{UG}^{mIELL}(\mu)$ and $\Delta \text{bdShC}(\mu)$ is very similar to that of case 4, thus omitted to avoid pedantry.

6. Rule (a). Let R be as in Figure 5.6a, and let l, l' be the two lifts.

   (a) If $l, l' \in \text{bdLft}(\mathcal{U})$, then by definition, we have $u_0, v_0 \notin \text{Sh}(\mathcal{U})$, while $w \in \text{bdSh}(\mathcal{U})$. After the reduction, the only residual vertex is $u_0 \notin \text{Sh}(\mu(\mathcal{U}))$. Hence,

$$\Delta \text{intSh}(\mu) = 0; \qquad (6.190)$$
$$\mathbb{C}_{UG}^{mIELL}(\mu) = 0; \qquad (6.191)$$
$$\Delta \text{bdShC}(\mu) = -1. \qquad (6.192)$$

   (b) Otherwise $l, l' \notin \text{bdLft}(\mathcal{U})$, and we consider three sub-cases depending on v_0, which by definition cannot belong to $\text{intSh}(\mathcal{U})$.

      i. If $v_0 \notin \text{Sh}(\mathcal{U})$, then also $u_0 \notin \text{Sh}(\mathcal{U})$, which means that l, l' are master lifts. Trivially, we have $u_0 \notin \text{Sh}(\mu(\mathcal{U}))$.

      ii. If $v_0 \in \text{bdSh}(\mathcal{U})$, then it must be the case that $u_0 \in \text{bdSh}(\mathcal{U})$ as well. Therefore $u_0 \in \text{bdSh}(\mu(\mathcal{U}))$. Also, there exists $c \in \text{bdShC}(\mathcal{U})$ such that $u_0, w, v_0 \in c$ if and only if there exists $c' \in \text{bdShC}(\mathcal{U})$ such that $u_0 \in c$.

      iii. If $v_0 \in \text{pbSh}(\mathcal{U})$, then by definition of pseudo-boundary, there exists a path $\pi_i : x_i \gg w$ (where it may be the case that $x_i = u_0$) such that $x_i \in \text{intSh}(\mathcal{U})$ while $x_i' \in \text{pbSh}(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. In the residual we have x_i, x_i' unchanged with respect to the share positioning.

   Therefore,

$$\Delta \text{intSh}(\mu) = 0; \qquad (6.193)$$
$$\mathbb{C}_{UG}^{mIELL}(\mu) = 0; \qquad (6.194)$$
$$\Delta \text{bdShC}(\mu) = 0. \qquad (6.195)$$

7. Rule (s). Let R be as in Figure 5.6b, and let l, l' be the two lifts.

   (a) If $l, l' \notin \text{bdLft}(\mathcal{U})$ but $l, l' \in \text{bdLft}(\mu(\mathcal{U}))$, then w necessarily belongs to $\text{Sh}(\mathcal{U})$, but not to $\text{intSh}(\mathcal{U})$. Before considering two sub-cases about w, we can already remark that, by hypotesis, one new positive boundary lift is introduced by μ.

      i. If $w \in \text{pbSh}(\mathcal{U})$, then there exists $\pi_i : x_i \gg u_0$ (where possibly $x_i = u_0$) such that $x_i \in \text{intSh}(\mathcal{U})$ while $x_i' \in \text{pbSh}(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. In the reduct we have $x_i, x_i' \in \text{bdSh}(\mu(\mathcal{U}))$, while $z_{0_0} \notin \text{Sh}(\mu(\mathcal{U}))$.

      ii. Otherwise $w \in \text{bdSh}(\mathcal{U})$, which implies that both $u_0, v_0 \in \text{bdSh}(\mathcal{U})$. In the reduct we still have $u_0, v_0 \in \text{bdSh}(\mu(\mathcal{U}))$, while $z_{0_0} \notin \text{Sh}(\mu(\mathcal{U}))$. Moreover, while $u_0, v_0 \in c \in \text{bdShC}(\mathcal{U})$, after the reduction $u_0 \in c$ and $v_0 \in d$ where $c, d \in \text{bdShC}(\mu(\mathcal{U}))$ and $c \neq d$.

   Hence:

$$\Delta \text{intSh}(\mu) = -1 + \Delta \text{bdShC}(\mu); \qquad (6.196)$$
$$\mathbb{C}_{UG}^{mIELL}(\mu) = 1 - \Delta \text{bdShC}(\mu); \qquad (6.197)$$
$$\Delta \text{bdShC}(\mu) \in [0, 1]. \qquad (6.198)$$

(b) Otherwise, observe that if l, l' cannot both belong to $bdLft\,(\mathcal{U})$, since that would mean that $\ell(l) = \ell(l')$, which would imply that R is an (s)-redex, contradicting our hypothesis. Moreover, by definition we cannot have one interior and one boundary lift. Therefore we have only three possible cases.

 i. If $w \notin Sh\,(\mathcal{U})$, which means that l, l' are both master lifts, then we trivially have $u_0, w, v_0 \notin Sh\,(\mathcal{U})$ and $u_0, z_{0_0}, v_0 \notin Sh\,(\mu(\mathcal{U}))$.

 ii. If $w \in pbSh\,(\mathcal{U})$, which means that l, l' are both interior lifts, then we have $v_0 \in pbSh\,(\mathcal{U})$. Moreover, there exists $\pi_i : x_i \gg u_0$ (where possibly $x_i = u_0$) such that $x_i \in intSh\,(\mathcal{U})$ while $x_i' \in pbSh\,(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. In the reduct we have $x_i', z_{0_0} \in pbSh\,(\mu(\mathcal{U}))$, and $x_i \in intSh\,(\mu(\mathcal{U}))$, while no change can affect boundary share components.

 iii. If $w \in Sh\,(\mathcal{U})$, which means that one lift, say the positive l, is boundary, and the other, say the negative l', is master. The dual case is omitted for the sake of conciceness. By such assumption and by definition, we then have $u_0 \notin Sh\,(\mathcal{U})$, while $w, v_0 \in bdSh\,(\mathcal{U})$. After the reduction, we obtain $u_0, z_{0_0} \notin Sh\,(\mathcal{U})$, while $v_0 \in bdSh\,(\mathcal{U})$. Moreover, the number of boundary share components is unchanged.

We conclude by summarising:

$$\Delta intSh\,(\mu) = 0; \tag{6.199}$$
$$\mathbb{C}^{mIELL}_{UG}(\mu) = 0; \tag{6.200}$$
$$\Delta bdShC\,(\mu) = 0. \tag{6.201}$$

8. Rule (m). Let R be as in Figure 5.8a, and l its lift.

(a) If $l \in bdLft\,(\mathcal{U})$, then $u_i \in bdSh\,(\mathcal{U})$, while $v_0, w \notin Sh\,(\mathcal{U})$. After μ, we obtain no change with respect to the share positioning. On the other hand, $\{u_i\} \in bdShC\,(\mathcal{U})$, which is erased by μ. Therefore,

$$\Delta intSh\,(\mu) = 0; \tag{6.202}$$
$$\mathbb{C}^{mIELL}_{UG}(\mu) = 0; \tag{6.203}$$
$$\Delta bdShC\,(\mu) = -1. \tag{6.204}$$

(b) Otherwise, we distinguish two sub-cases about u_0, which by our assumption cannot belong to $intSh\,(\mathcal{U})$, nor to $bdSh\,(\mathcal{U})$.

 i. If $u_i \notin Sh\,(\mathcal{U})$, then trivially also $v_0, w \notin Sh\,(\mathcal{U})$. In the reduction u_i is erased, and we have no change about internal share or its boundaries.

 ii. If $u_i \in pbSh\,(\mathcal{U})$, then $w \notin Sh\,(\mathcal{U})$. Also, there exists $\pi_i : x_i \gg v_0$ (where it may be the case that $x_i = v_0$) such that $x_i \in intSh\,(\mathcal{U})$ while $x_i' \in pbSh\,(\mathcal{U})$ for every $x_i' \neq x_i$ in π_i. Now, in the residual we have that x_i, x_i' are unchanged with respect to the share positioning. Hence, again there is no change about internal share or its boundaries.

$$\Delta intSh\,(\mu) = 0; \tag{6.205}$$
$$\mathbb{C}^{mIELL}_{UG}(\mu) = 0; \tag{6.206}$$
$$\Delta bdShC\,(\mu) = 0. \tag{6.207}$$

■

6.5.3 Unshared cost of SG reduction

We now define $\mathbb{C}_{UG}^{SG}$, a second notion of cost on unshared graphs reduction that, symmetrically with respect to what we did with $\mathbb{C}_{UG}^{mIELL}$, allows to transfer the $\mathbb{C}_{SG}$-cost of any sharing reduction simulated by UG graphs. Looking at a simulation of such kind, the first inspiring idea is that, in a UG-graph $\mathcal{U}$, all reductions inside $Sh(\mathcal{U})$ have no cost. Indeed, shared vertices, shared links and shared redexes are copies of master copies, and only these are represented in the SG-graph that unfold in $\mathcal{U}$. Hence, we shall put costs only to redexes outside the share. The second ingredient is the role played by lifts, which mimic muxes' dynamics while preserving coherence in the complexity account. For every mux m in $\mathcal{G}$, we distribute the cost of a propagation against another kind of link, or of an interaction with another mux, over the set of boundary lifts in the unfolding of m. These considerations lead us to the following definition.

Definition 6.18 (Cost on unshared graph). Given a UG or RB reduction step μ, the *SG-cost* of μ, written $\mathbb{C}_{UG}^{SG}(\mu_i)$, is defined in Table 6.4. The SG-cost of a reduction sequence is the sum of the costs of its steps.

Table 6.4 Shared cost of an unshared reduction step $\mu_i : \mathcal{U} \to \mathcal{U}'$. When present, l, l' are lifts, and $k + 1$ is the number of premisses of the ?-link.

Rule	Provisos	$\mathbb{C}_{UG}^{SG}(\mu)$
$(\multimap)$	$\mu \notin Sh(\mathcal{U})$	9
$(!)$	$\mu \notin Sh(\mathcal{U})$	6
(tD)	$\mu \notin Sh(\mathcal{U})$	$j + 4$
$(\multimap), (!), (tD)$	$\mu \in Sh(\mathcal{U})$	0
$(d!)$	$l \in bdLft(\mathcal{U})$	3
$(d \multimap), (d\bar{\multimap}), (r\bar{\multimap})$	$l \in bdLft(\mathcal{U})$	5
$(d?), (r?)$	$l \in bdLft(\mathcal{U})$	$2k + 3$
$(d\kappa), (r\kappa)$	$l \notin bdLft(\mathcal{U})$	0
$(a), (m)$	$l, l' \in bdLft(\mathcal{U})$	1
	otherwise	0
(s)	$bdLft(\mathcal{U}) \ni l, l' \in bdLft(\mu(\mathcal{U}))$	1
	otherwise	0

Is such a definition correct? To verify that, we deep into the unfolding relation and the simulation, looking at their effects on sharing contexts in the unshared graph. We find a strong invariance. Consider a set V of vertices in a UG graph such that they all belong to the unfolding of a same vertex in a SG graph. If in the l-context of $v \in V$ appears $x_{i:k}$, then we know that for any $0 \le j \le k$ there exists a unique $v' \in V$ whose l-context differs from that of v only for $x_{i:k}$, where instead appears $x_{j:k}$. This essentially means that the set V is in a bijection with all the possible variations to indices that one can perform on their sharing l-context (Proposition 6.2). Thanks to this crucial proposition, we then easily subsume the fact that every vertex in a SG graph has a unique master vertex in its unfolding (Lemma 6.17), and similarly that, for every $k + 1$-ary mux in a SG graph, there are exactly k boundary lifts in its unfolding (Lemma 6.18). These two facts pave the way to prove that the unshared cost of a UG-reduction sequence is always equal to the cost of the SG-sequence it simulates (Lemma 6.19).

Proposition 6.2 (Unfolding and context permutations). *Let $\sigma \hookrightarrow \mu$ be a pair of simulating reductions such that $\mathcal{N} \xrightarrow{\sigma}_{SG}^{*} \mathcal{G}$, and $\mathcal{N} \xrightarrow{\mu}_{UG}^{*} \mathcal{U}$, where $\mathcal{G} \hookrightarrow \mathcal{U}$. Let $u \leftrightarrow v$ for some $u \in V(\mathcal{U}), v \in V(\mathcal{G})$ such that $a \cdot b \cdot c = \mathfrak{s}(u)(n)$ stable, for some $a, b, c \in \mathfrak{S}^{*}$. If $b = x_{i:k}$ (or $\overline{x_{i:k}}$), then for any $0 \leq i' \leq k$ there is a unique $u' \leftrightarrow v$ such that:*

1. *$\mathfrak{s}(u')(n) = a \cdot b' \cdot c$, with $b' = x_{i':k}$ (or $\overline{x_{i':k}}$, respectively); and*

2. *$\mathfrak{s}(u')(n') = \mathfrak{s}(u)(n')$, for any $n' \neq n$.*

Proof. We proceed by induction on the length of the SG-reduction $\bar{\sigma}$ such that $\mathcal{G} = \bar{\sigma}(\mathcal{N})$ for some proof-net $\mathcal{N}$. The base case is trivial, because when $|\bar{\sigma}| = 0$ we have $\mathcal{G} = \mathcal{N}$ and since $\mathcal{G} \hookrightarrow \mathcal{U}$ also $\mathcal{U} = \mathcal{N}$. Therefore, for any $s \in V(\mathcal{G})$ we have $\mathfrak{s}(s) = 1$ and the claim vacuously holds. So assume otherwise and in particular: let $\mathcal{G} = \sigma(\mathcal{G}')$ for some SG-reduction step σ on a redex S such that $\mathcal{G}' = \bar{\sigma}'(\mathcal{N})$; let $\mathcal{U}' \leftarrow \mathcal{G}'$; and let $\bar{\mu} \leftarrow \sigma$ for some UG-reduction sequence $\bar{\mu}$ on a set of redexes $\bar{M}$ such that $\mathcal{U} = \bar{\mu}(\mathcal{U}') = \bar{\mu}(\bar{\mu}'(\mathcal{N}))$. Finally, take $s \in V(\mathcal{G})$ and $t \leftrightarrow s$ such that $\mathfrak{s}(t)(n) = a \cdot b \cdot c$ where $b = x_{l:m}$ or $\overline{x_{l:m}}$, for some $n, l, m \in \mathbb{N}$, and some $a, x_{l:m}, c \in \mathfrak{S}^{*}$. Finally let $0 \leq l' \leq m$ be the index for which we want to prove the claim.

1. If $s \notin \mathrm{int}(\sigma(S))$, then by definition of reduction $\sigma^{-1}(s) = s$. Moreover, for any redex $M \in \bar{M}$ we also have $t \notin \bar{\mu}(M)$, as per definition of unfolding. Therefore, in accordance with invariance Lemma 6.14, $\mathfrak{s}(t) = \mathfrak{s}(\bar{\mu}^{-1}(t))$. So let $\bar{\mu}^{-1}(t_l) \leftrightarrow \bar{\sigma}^{-1}(s)$ be the unique vertex of $\mathcal{U}'$ such that the claim holds by IH. Now, since $\mathfrak{s}(\bar{\mu}^{-1}(t_l)) = \mathfrak{s}(t_l)$, we trivially conclude that the claim still holds.

2. If $s \in \mathrm{int}(\sigma(S))$, then let $M \in \bar{M}$ such that $t \in M$, and let $\mu \in \bar{\mu}$ be the reduction step on M. We proceed with a case analysis on the kind of the redex S of σ.

   (a) Rules $(!), (\multimap), (a), (m)$. Absurd: by inspection of the definition of the redexes, we verify that $\mathrm{int}(\sigma(S)) = \varnothing$, contradicting the hypothesis of $s \in \mathrm{int}(\sigma(S))$.

   (b) Rule $(d!)$. Let S be as in Figure 5.5c, and, since $\mathrm{int}(\sigma(S)) = \{z_0, \ldots, z_k\}$, let $s = z_i$. Now, assume vertex names in M are as in S, but with the prime symbol, and let $t = z'_{i'}$. Observe that in $\mu(M)$, as per definition of context assignment, we have $\mathfrak{s}(v_{i'}) = \mathfrak{s}(z_{i'})$. But $\mathfrak{s}(v_{i'})$ belongs to $C(M)$, where by Lemma 6.14 it must have the same levelled context. Now, since $v_i \hookrightarrow v_{i'}$, let $v''_{i'}$ be the unique vertex in $\mathcal{U}'$ such that $v_i \hookrightarrow v''_{i'}$ and it satisfies the claim, i.e. $\mathfrak{s}(v''_{i'})(n) = a \cdot x_{l':m} \cdot c$, while $\mathfrak{s}(v''_{i'})(n') = \mathfrak{s}(v'_{i'})(n')$, for any $n' \neq n$. So, let M'' be the redex containing $v''_{i'}$, and μ'' be its reduction step. Since it belongs to the $C(M'')$, we find $v''_{i'}$ also in $\mu''(M'')$ with the same levelled-context. Consider $z''_{i'}$ in $\mu''(M'')$ and verify not only that $z_i \hookrightarrow z''_{i'}$, but also that, as previously remarked, $\mathfrak{s}(v''_{i'}) = \mathfrak{s}(z''_{i'})$. Hence the claim.

   (c) Rules $(d?), (d\multimap), (d\overline{\multimap}), (s), (r\multimap), (r?)$. The argument detailed in case 2b can be applied here with only minor changes.

   (d) Rule (tD). Let S be as in Figure 5.9, but let d be the level of !-link, and observe that $\mathrm{int}(\sigma(S)) = \bigcup_{1 \leq i \leq h} \mathcal{B}_i$, where $\mathcal{B}_i$ denotes the i-copy of the interior of the box $\mathcal{B}$. Let the names of vertices and boxes in M be as in S, but with prime symbols. So let $t \in \mathcal{B}'_{i'}$ and consider $\sigma^{-1}(t)$. Assume

the downward crossing of the i-th lift above $\mathcal{B}'_{i'}$ has context $!_n e$, Also, let $\mathfrak{s}\left(\mu^{-1}(t)\right)(d) = f \cdot g$, where $f = \mathfrak{s}(v_i)$. Then, by definition of context assignment, we first observe the following.

$$\mathfrak{s}(t)(n') = \begin{cases} \mathfrak{s}\left(\sigma^{-1}(t)\right) & \text{if } d' \neq d; \\ f \cdot e \cdot g & \text{if } d' = d. \end{cases} \tag{6.208}$$

Now, we observe that we can apply IH on $\mu^{-1}(t)$. Namely, for any $n \neq d$ and for any $x_{l:m} \neq e$ (or its negation), appearing in $\mathfrak{s}\left(\mu^{-1}(t)\right)$ at the n-th context, we know there exists a unique vertex t' in $\mathcal{U}$, such that $\sigma^{-1}(s) \hookrightarrow t'$ and for which: in $\mathfrak{s}(t')(n)$ is obtained from $\mathfrak{s}(t)$ substituting $x_{l:m}$ for $x_{l':m}$, while $\mathfrak{s}(t')(n') = \mathfrak{s}\left(\mu^{-1}(t)\right)(n')$, for any $n' \neq n$.

Now observe that t' belongs to a box $\mathcal{B}''$ in a redex M', so let μ' be its reduction step, which by definition of unfolding belongs to $\bar{\mu}$. Now, let $t'' \in \mu'(t')$ be the vertex belonging in the i-th copy of the box $\mathcal{B}''$, and verify that it satisfies the claim.

∎

For every vertex in a SG graph, there exists a unique master vertex in its unfolding.

Lemma 6.17 (Master copy). *Let* $N \xrightarrow{\bar{\sigma}}^*_{SG} \mathcal{G}$ *and* $N \xrightarrow{\bar{\mu}}^*_{UGRB} \mathcal{U}$ *be two reduction sequences such that* $\bar{\sigma} \hookrightarrow \bar{\mu}$. *For any* $v \in V(\mathcal{G})$ *and any* $V' \subset V(\mathcal{U})$, *if* $v \hookrightarrow V'$ *then there exists a unique* $v_m \in V'$ *such that* $\mathfrak{s}(v_m) \approx 1$, *and it is called the* master copy *of* v.

Proof. Immediate from Proposition 6.2 by fixing in its statement $i' = 0$. ∎

For every $k + 1$-ary mux in a SG graph, there are exactly k boundary lifts its unfolding.

Lemma 6.18 (Arities of muxes and cardinalities of their boundary lifts). *Let* $N \xrightarrow{\bar{\sigma}}^*_{SG} \mathcal{G}$ *and* $N \xrightarrow{\bar{\mu}}^*_{UGRB} \mathcal{U}$ *be two reduction sequences such that* $\bar{\sigma} \hookrightarrow \bar{\mu}$. *For any mux* $m \in L(\mathcal{G})$ *of arity* $k + 1$, *let* $L = \{l \in \text{bdLft}(\mathcal{U}) \mid m \hookrightarrow l\}$. *Then* $|L| = k$.

Proof. Let $m = \langle u_0, \ldots u_k \, (\mid_\star) \rangle z$ and consider the set $L' \subseteq L(\mathcal{U})$ of any lift whose premiss is the master copy of u_i, for some $0 \leq i \leq k$. Recall that, thanks to Lemma 6.17, we know that for any i the master copy is unique, so $|L'| = k + 1$. Now by definition of the unfolding relation, there is a sharing morphism between $\mathcal{U}$ and $\mathcal{G}$ which is connection-preserving and surjective (cf. Definition 5.8). Therefore, the master copy of z must be the conclusion of a lift $l_m \in L'$, which consequently is a master lift. Now let $L_b = L' \smallsetminus l_m$ and notice that it contains all and only the boundary lifts that are unfolding of m. Indeed, for every $l_i \in L_B$ different than l_m, we have $u_i \notin \text{Sh}(\mathcal{U})$ while $z_i \in \text{Sh}(\mathcal{U})$. Thus $L_b = L$ and trivially $|L_B| = k$. ∎

The unshared cost of a UG-reduction sequence is always equal to the cost of the SG-sequence it simulates.

Lemma 6.19 (Correctness of $\mathbb{C}^{SG}_{UG}$). *If* $N \xrightarrow{\bar{\sigma}}^*_{SG} \mathcal{G}$ *and* $N \xrightarrow{\bar{\mu}}^*_{UGRB} \mathcal{U}$ *are two reduction sequences such that* $\bar{\sigma} \hookrightarrow \bar{\mu}$, *then* $\mathbb{C}_{SG}(\bar{\sigma}) = \mathbb{C}^{SG}_{UG}(\bar{\mu})$.

Proof. We go by induction on $|\bar{\sigma}|$. The base case is trivial, because when $|\bar{\sigma}| = 0$ we have $\mathcal{G} = \mathcal{N}$ and since $\mathcal{G} \hookrightarrow \mathcal{U}$ also $\mathcal{U} = \mathcal{N}$. This means that $\mathbb{C}_{SG}(\bar{\sigma}) = \mathbb{C}_{UG}^{SG}(\bar{\mu}) = 0$, so the claim vacuously holds. So assume $|\bar{\sigma}| > 0$. Let $\mathcal{G} = \sigma(\mathcal{G}')$ for some SG-reduction step σ on a redex S such that $\mathcal{G}' = \bar{\sigma}'(\mathcal{N})$; let $\mathcal{U}' \leftarrow \mathcal{G}'$; and let $\bar{\mu} \leftarrow \sigma$ for some UG-reduction sequence $\bar{\mu}$ on a set of redexes $\bar{M}$ such that $\mathcal{U} = \bar{\mu}(\mathcal{U}') = \bar{\mu}(\bar{\mu}'(\mathcal{N}))$. By inductive hypothesis we also assume $\mathbb{C}_{UG}^{SG}(\bar{\mu}') = \mathbb{C}_{SG}(\bar{\sigma}')$, so we need to prove that $\mathbb{C}_{UG}^{SG}(\bar{\mu}) = \mathbb{C}_{SG}(\sigma)$.

1. Suppose that S does not contain muxes, i.e. the rule of S is one of the following: $(\multimap), (!), (t)$. Then by Lemma 6.17, let $M \in \bar{M}$ be the unique redex such that for any $v' \in M$ and any $v \in S$, if $v \hookrightarrow v'$ then v' is the master copy of v. Now, let μ be the reduction of M and $\bar{\mu}''$ be the reduction of $\bar{M} \smallsetminus M$. Now, observe that by Definition 6.3 of $\mathbb{C}_{SG}$ (see in particular Table 6.2) and Definition 6.18 of $\mathbb{C}_{UG}^{SG}$ (see in particular Table 6.4) we have $\mathbb{C}_{UG}^{SG}(\mu) = \mathbb{C}_{SG}(\sigma)$, while $\mathbb{C}_{UG}^{SG}(\bar{\mu}'') = 0$. Therefore $\mathbb{C}_{UG}^{SG}(\bar{\mu}) = \mathbb{C}_{SG}(\sigma)$.

2. If S contains muxes, and is not a (s) rule, then let $\bar{M}'' \subset \bar{M}$ be the set of any redex in $\bar{M}$ whose lifts are boundary. First, observe that, again by Definition 6.3 and 6.18 (cf. Table 6.2 and 6.4), for any reduction step μ'' on a redex of $\bar{M}''$, we have $\mathbb{C}_{SG}(\sigma) = k \times \mathbb{C}_{UG}^{SG}(\mu'')$, where $k + 1$ is the number of premisses of the mux of S. Conversely, for any reduction step μ''' on a redex of $\bar{M} \smallsetminus \bar{M}''$, as per definition, $\mathbb{C}_{UG}^{SG}(\bar{\mu}''') = 0$. But by Lemma 6.18 we know that $|\bar{M}''| = k$. Thus, $\mathbb{C}_{UG}^{SG}(\bar{\mu}) = \mathbb{C}_{SG}(\sigma)$.

3. Otherwise S is a (s) rule. Let $k + 1$ and $l + 1$ be the number of premisses of the positive and negative mux of S, respectively. Observe that by definition of reduction (cf. Figure 5.6b) in $\sigma(S)$ there are $k + 1$ negative lifts with arity $l + 1$, and $l + 1$ positive lifts with arity $k + 1$.

   By Lemma 6.18 this implies that $\bar{M}$ contains k positive boundary lifts and l negative boundary lifts. Since by definition of boundary lift, there cannot be two boundary lifts in a (s) redex, the boundary lifts of $\bar{M}$ belong to different redexes. Now, again per Lemma 6.18, it must be the case that $\bar{\mu}(\bar{M})$ contains $(k + 1) \times l$ negative lifts, and $(l + 1) \times k$ positive lifts. But this means that $\bar{\mu}$ introduces $k \times l$ negative lifts and $k \times l$ positive lifts, i.e. there are $k \times l$ redexes in $\bar{M}$ made of two pseudo-boundary lifts which become boundary when swapped. (This happens when the levelled context of premiss of the positive lift is 1 everywhere but at the level of the negative lift, where the context is identical to the variable occurrence of the and the negative lift.) Since $\mathbb{C}_{UG}^{SG}$ assign a unitary cost of such reduction, and 0 otherwise, we obtain that $\mathbb{C}_{UG}^{SG}(\bar{\mu}) = k \times l = \mathbb{C}_{SG}(\sigma)$.

■

6.6 Unshared cost comparison

Thanks to the careful analysis of the metrics on UG-reduction so far accomplished, we are now able to prove the complexity bound. We first observe that for any UG-reduction, the difference between its sharing cost and its mIELL cost is bounded by

its boundary-share-components cost (Lemma 6.20). Then, we proceed by bounding boundary-share-components cost to a quadratic function of the mIELL cost (Lemma 6.21 and 6.22). In such proof it is worth noticing that the only kind of reduction that we did not manage to bound by a linear function is the swap within a boundary share component generating a new component. In such a case, the bound depends linearly on the depth and quadratically on the mIELL cost. Finally we conclude by Theorem 6.1, which compares $\mathbb{C}_{SG}$ with $\mathbb{C}_{mIELL}$ for any pair of simulating reduction sequence.

Lemma 6.20 (Bound for unshared costs comparison). *For any proof-net $\mathcal{N}$ and any reduction sequence $\mathcal{N} \xrightarrow{\bar{\mu}}_{UGRB}^* \mathcal{U}$:*

$$\mathbb{C}_{UG}^{SG}(\bar{\mu}) - \mathbb{C}_{UG}^{mIELL}(\bar{\mu}) \le \mathbb{C}_{UG}^{bdShC}(\bar{\mu}). \tag{6.209}$$

Proof. Given a step μ of $\bar{\mu}$, it suffices a mere inspection of Lemma 6.16 and Definition 6.18 and, in particular, a comparison of the corresponding cases of Table 6.3 and Table 6.4, to immediately verify the inequation. ∎

Lemma 6.21 (Redundant boundary share components). *Given a UG-graph $\mathcal{U}$, and $v \in \text{bdLimSh}(\mathcal{U})$, let $L : u \sim v$ be the boundary lift chain of v and let $\bar{\mu}$ be a reduction sequence on L. If n is the number of any (s)-step $\mu \in \bar{\mu}$ such that $\Delta\text{bdShC}(\mu) = 1$, then*

$$n \le \ell(\mathcal{U}). \tag{6.210}$$

Proof. By hypothesis, $u \gg v$ or $v \gg v$. If both hold, then $u = v$ and the claim is trivially verified, because $L = \varnothing$. Otherwise, since the arguments of the two cases are identical, we shall assume $u \gg v$. Let $\bar{\mu}_{cs}$ be the set of any (s)-step μ in $\bar{\mu}$ such that $\mathbb{C}_{UG}^{bdShC}(\mu) = 1$.

We will now prove that for any level n there may be at most one step $\mu \in \bar{\mu}_{cs}$. This implies the claim, because in the worst case L contains lifts of any level $l \le \ell(\mathcal{U})$.

For the sake of contradiction, assume otherwise that L contains: a $|+\rangle^n$-link l, a $|+\rangle^{n'}$-link l', and $|-\rangle^{n''}$-link l'' such that there are two steps in $\bar{\mu}$ swapping both l, l' with l''. (The dual situation with one $|+\rangle$-link and two $|-\rangle$-link is identical, hence omitted.) Let $\text{VarO}(l) = a$, $\text{VarO}(l') = b$, and $\text{VarO}(l'') = c$. Then, there are some l-contexts $\gamma_1, \gamma_2, \gamma_3$ such that:

$$\mathfrak{s}(v) = \delta \cdot \underbrace{\gamma_1 \cdot !_n a \cdot \gamma_2 \cdot !_{n'} b \cdot \gamma_3}_{\gamma_0} \cdot !_{n''} \bar{c} \cdot \gamma_4, \tag{6.211}$$

where γ_0 is the l-context of the downward path π that goes from u to the conclusion of l''. Now, L contains only $|*\rangle$-links, therefore any reductions in $\bar{\mu}$ is necessarily an (s)- or an (a)-rule. By hypothesis, there is no (a)-rule involving l, l' or l'', hence π is long enough for any reduction step of $\bar{\mu}$. Therefore the context of π is invariant under any reduction preceding the swap between l' and l''. For such (s)-step we hypothesised that in the redex the lifts are interior, while in the reduct they are boundary. Thus $\mathfrak{s}(\gamma_0)(n') = b$ while $\mathfrak{s}(\gamma_0)(m) = 1$ for any $m \ne n'$. But given the presence of the weight of l, i.e. $!_n a$, this implies that $\mathfrak{s}(\gamma_2)(n) = \bar{a}$, while $\mathfrak{s}(\gamma_2)(m) = 1$, for any $m \ne n$. Therefore that π contains a $|-\rangle$-link which annihilates with l. Absurd. ∎

Lemma 6.22 (Bound for closed border reduction). *For any UGRB-reduction sequence $\bar{\mu}$ on a proof-net $\mathcal{N}$, $\mathbb{C}_{UG}^{bdShC}(\bar{\mu})$ is bounded by a quadratic function on $\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu})$.*

Proof. Let $\bar{\mu}$ be a UGRB-reduction sequence. We partition the reduction steps of $\bar{\mu}$ into three sets depending on their action on bdShC:

$\bar{\mu}_e$ the set of any step μ of $\bar{\mu}$ such that $\Delta \text{bdShC}(\mu) = -1$ (i.e. an erasure of a boundary share component), hence μ is of kind $(a), (m)$;

$\bar{\mu}_{cs}$ the set of any step μ of $\bar{\mu}$ such that $\Delta \text{bdShC}(\mu) = 1$ and μ is of kind (s) (i.e. a creation of boundary share component with a swap);

$\bar{\mu}_{co}$ the set of any step μ of $\bar{\mu}$ that does not belong to $\bar{\mu}_e$ or $\bar{\mu}_{cs}$.

Hence, by construction:

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}) = \mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_e) + \mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs}) + \mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{co}), \tag{6.212}$$

so we can separately discuss the three addends.

1. $\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_e)$. We observe that, since $\#(\text{bdShC}(\cdot))$ is always non negative, the number of erasures cannot outnumber the creations. Therefore,

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_e) \le \mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs}) + \mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{co}). \tag{6.213}$$

2. $\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs})$. Let $\bar{\mu}_{cs}'$ be a maximal subsequence of $\bar{\mu}$ such that, if $\bar{\mu} = \bar{\nu}\bar{\mu}_{cs}'\bar{\omega}$ for some reduction sequences $\bar{\nu}, \bar{\omega}$, then any step of $\bar{\mu}_{cs}'$ acts on the boundary lift chain L of every $v \in \text{bdLimSh}(\bar{\nu}(\mathcal{U}))$. Now, by Lemma 6.21 we have that the normalisation of L entail and increase of $\#(\text{bdShC})$ of at most $\ell(\mathcal{U})$. Moreover, by Fact 6.2 we know that $\#(\text{bdLimSh}(\bar{\nu}(\mathcal{U}))) \le \#(\text{intSh}(\bar{\nu}(\mathcal{U})))$. Therefore

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs}') \le \ell(\mathcal{U}) \times \#(\text{intSh}(\bar{\nu}(\mathcal{U}))), \tag{6.214}$$

and simplier:

$$\le \ell(\mathcal{U}) \times \widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\bar{\mu}). \tag{6.215}$$

Now, observe that the distinct sub-sequences of $\bar{\mu}$ defined as $\bar{\mu}_{cs}'$ are at most as many as the number of steps $\mu \in \bar{\mu}$ such that $\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) = 0$. Therefore:

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs}) \le \ell(\mathcal{U}) \times \mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) \times \widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\mu), \tag{6.216}$$

where, since $\mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu) \le \widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\mu)$, we can loosen and simplify as:

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{cs}) \le \ell(\mathcal{U}) \times (\widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\mu))^2. \tag{6.217}$$

3. $\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{co})$. Recall from the account provided by Lemma 6.16 in Table 6.3 that, for any step $\mu \in \bar{\mu}_{co}$, $\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\mu) \le \mathbb{C}_{\text{UG}}^{\text{mIELL}}(\mu)$. Therefore,

$$\mathbb{C}_{\text{UG}}^{\text{bdShC}}(\bar{\mu}_{co}) \le \mathbb{C}_{\text{UG}}^{\text{mIELL}}(\bar{\mu}_{co}) \tag{6.218}$$

$$\le \mathbb{C}_{\text{UG}}^{\text{mIELL}}(\bar{\mu}) \tag{6.219}$$

$$\le \widehat{\mathbb{C}}_{\text{UG}}^{\text{mIELL}}(\bar{\mu}). \tag{6.220}$$

Now we go back to (6.212) and substitute first (6.213):

$$\mathbb{C}_{UG}^{bdShC}(\bar{\mu}) \le 2 \times \left(\mathbb{C}_{UG}^{bdShC}(\bar{\mu}_{cs}) + \mathbb{C}_{UG}^{bdShC}(\bar{\mu}_{co}) \right) \tag{6.221}$$

and then (6.217) and (6.220)

$$\mathbb{C}_{UG}^{bdShC}(\bar{\mu}) \le 2 \times \left(\ell(\mathcal{U}) \times \left(\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right)^2 + \widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right) \tag{6.222}$$

$$\le 2 \times \widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) + 2 \times \ell(\mathcal{U}) \times \left(\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right)^2. \tag{6.223}$$

∎

Proof of Theorem 6.1. By definition of ↠, there exists an unshared graph $\mathcal{U}$ such that $\mathcal{G} \rightarrow \mathcal{U} \mapsto \mathcal{N}$ and there exists an intermediate unshared reduction $\bar{\mu} : \mathcal{N} \rightarrow_{UG}^{*} \mathcal{U}$. On one side, by Lemma 6.15, we know that $\mathbb{C}_{mIELL}(\bar{\rho}) = \widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu})$, and on the other, by Lemma 6.19, we similarly have: $\mathbb{C}_{SG}(\bar{\sigma}) = \mathbb{C}_{UG}^{SG}(\bar{\mu})$.

Now, as per Lemma 6.22 and Lemma 6.20, there exists a quadratic function f such that:

$$\mathbb{C}_{UG}^{SG}(\bar{\mu}) \le \mathbb{C}_{UG}^{mIELL}(\bar{\mu}) + f\left(\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right). \tag{6.224}$$

But since $\mathbb{C}_{UG}^{mIELL}(\bar{\mu}) \le \widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu})$, by definition:

$$\mathbb{C}_{UG}^{SG}(\bar{\mu}) \le f\left(\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right), \tag{6.225}$$

therefore

$$\mathbb{C}_{SG}(\bar{\sigma}) = \mathbb{C}_{UG}^{SG}(\bar{\mu}) \le f\left(\widehat{\mathbb{C}}_{UG}^{mIELL}(\bar{\mu}) \right) = f\left(\mathbb{C}_{mIELL}(\bar{\rho}) \right). \tag{6.226}$$

Quod erat demonstrandum. ∎

6.7 Discussion

6.7.1 Related works

Baillot, Coppola, and Dal Lago [2011] proved a results similar to ours. In particular, they show that for any ILAL (or IEAL) proof-net $\mathcal{N}$, the length of a SG-normalisation sequence on $\mathcal{N}$ and a read-back sequence is bounded by a polynomial [ibidem, Proposition 12] (or Kalmar-elementary [ibidem, Proposition 11]) function in the size of $\mathcal{N}$. This essentially means that SG-reduction cannot do dramatically worse than the reference systems, it cannot overstep their characteristic complexity classes. From a pragmatical point of view, this still tells us little, since one may think that the overhead is indeed polynomial, or even Kalmar-elementary. The first and more important advance we have presented is that we proved a stronger and finer limitation — a quadratic bound, whose dominating portion are a special case of swaps. Moreover, since we bound the complexity of any SGRB reduction sequence, and not necessarily a normalising one, our result is more generally stated, so it can be directly applied even with weaker notion of normal forms.

Their proof rests on a well-known correspondence between sharing reductions and GoI, adapting the ideas of the quantitative context semantics [Dal Lago, 2009]. Also, it seems

hardly adaptable to prove a similar result for the general case, because their argument directly exploits the known complexity bounds of the IEAL and ILAL. We instead accomplished the result directly by more syntactical means, in a dynamic comparison of the two cost measures. Such technique and the tool of sharing contexts represents a complexity-aware extension of the syntactical approach that has been already fruitfully used before [Guerrini, Martini, and Masini, 2003].

6.7.2 Open questions

Not much later its introduction, empirical benchmarks showed that sharing graphs can be a very efficient implementation of the λ-calculus. Terms that allow massive amount of sharing, indeed, can be normalised by a sharing-graph-based machine in polynomial time, whilst that requires exponential time with ordinary implementations in functional programming languages [Asperti, Giovannetti, and Naletto, 1996]. But the sharing machinery comes at a price, that is still unknown for the general case, but that in the restricted case of elementary proof-nets has been bounded, now quite satisfactorily, to a quadratic function. The result and the techniques open different questions.

Can we prove a similar complexity bound also in the general case of MELL proof-nets? Can we do so with the quantitative syntactic approach? The research hypothesis one may require is the legitimation of a cost for exponential reductions that always depends on the size of a box, even when they act on box nesting relocating one into another (e.g. promotion against dereliction).

Also on the same line, can we find not only a tight upper bound to the overhead introduced by optimal implementations, which gives a worst-case limitation, but also some convincing best-case limitation? How much share is needed in order for the trade-off to be convenient? When does the mux management overhead is amortised by the efficiency of sharing?

Can we relate SG with other more concrete computational models like abstract machines? Some of them, like the IAM [Laurent, 2001], are of particular interest because of their distinguished space efficiency.

Sharing contexts appear as a powerful tool to study SG in their unshared counterpart. Not only they can be used to provide a simple proof of their completeness with respect to normalisation, since the absence of deadlocks is obtainable as consequence of the positivity property, very similarly to what was done by Guerrini [1999], though more simply, thanks to the restricted setting of mIELL. But also offer a precise quantitative relation between a shared vertex in a sharing graph and its unfolding in the unshared graph (cf. Proposition 6.2). This makes of sharing context a promising tool to tackle all these complexity-related questions about sharing implementation and Lévy families, more in general.

We identified a kind of reduction steps that is critical for the complexity bound. Can it be directly lowered finding a tighter bound, perhaps linear? If not, are there any tunings for the rewriting system under which the problem can be circumvented? For instance, one may explore the idea to introduce additional rules to reduce the number of redundant swap rules, as it was already proposed and implemented for redundant

index-management links [Asperti and Chroboczek, 1997] [Asperti and Guerrini, 1998, Chapter 9].

Index

Bibliography

Beniamino Accattoli. Linear Logic and Strong Normalization. In Femke van Raams-donk, editor, *24th International Conference on Rewriting Techniques and Applications, RTA 2013, June 24-26, 2013, Eindhoven, The Netherlands*, volume 21 of *LIPIcs*, page 39–54. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2013. ISBN 978-3-939897-53-8. doi: 10.4230/LIPIcs.RTA.2013.39.

Sandra Alves and Mário Florido. Weak linearization of the lambda calculus. *Theoretical Computer Science*, 342(1):79–103, 2005. ISSN 0304-3975. doi: 10.1016/j.tcs.2005.06.005. URL http://www.sciencedirect.com/science/article/pii/S0304397505003403.

Andrea Asperti. Light affine logic. In *Logic in Computer Science, 13th International Symposium, Proceedings*, page 300–308. IEEE Computer Society, 1998.

Andrea Asperti and Juliusz Chroboczek. Safe Operators: Brackets Closed Forever Optimizing Optimal lambda-Calculus Implementations. *Applicable Algebra in Engineering, Communication and Computing*, 8(6):437–468, December 1997. ISSN 0938-1279, 1432-0622. doi: 10.1007/s002000050083. URL http://link.springer.com/article/10.1007/s002000050083.

Andrea Asperti and Stefano Guerrini. *The Optimal Implementation of Functional Programming Languages*, volume 45 of *Cambridge tracts in theoretical computer science*. Cambridge University Press, 1998.

Andrea Asperti and Cosimo Laneve. Paths, computations and labels in the lambda-calculus. *Theoretical Computer Science*, 142(2):277–297, 1995. ISSN 0304-3975. doi: 10.1016/0304-3975(94)00279-7.

Andrea Asperti and Harry G. Mairson. Parallel Beta Reduction Is Not Elementary Recursive. *Information and Computation*, 170(1):49–80, 2001. doi: 10.1006/inco.2001.2869.

Andrea Asperti and Luca Roversi. Intuitionistic light affine logic. *ACM Transactions on Computational Logic (TOCL)*, 3(1):137–175, 2002.

Andrea Asperti, Vincent Danos, Cosimo Laneve, and Laurent Regnier. Paths in the lambda-calculus. In *Logic in Computer Science, 1994*, page 426–436. IEEE, 1994. doi: 10.1109/LICS.1994.316048.

Andrea Asperti, Cecilia Giovannetti, and Andrea Naletto. The Bologna optimal higher-order machine. *Journal of Functional Programming*, 6(6):763–810, November

1996. ISSN 1469-7653, 0956-7968. doi: 10.1017/S0956796800001994. URL https://www.cambridge.org/core/journals/journal-of-functional-programming/article/the-bologna-optimal-higher-order-machine/1F2763B0F931680F9B15BDC750BEB343.

Andrea Asperti, Paolo Coppola, and Simone Martini. (Optimal) Duplication is not elementary recursive. *Information and Computation*, 193, 2004.

Clément Aubert and Thomas Seiller. Characterizing co-NL by a group action. *Mathematical Structures in Computer Science*, FirstView:1–33, December 2014. ISSN 1469-8072. doi: 10.1017/S0960129514000267. URL http://journals.cambridge.org/article_S0960129514000267.

Clément Aubert and Thomas Seiller. Logarithmic Space and Permutations. *Information and Computation*, November 2015. URL https://hal.archives-ouvertes.fr/hal-01005701/document. To appear.

Clément Aubert, Marc Bagnol, and Thomas Seiller. Unary Resolution: Characterizing Ptime. In Bart Jacobs and Christof Löding, editors, *Foundations of Software Science and Computation Structures*, number 9634 in Lecture Notes in Computer Science, page 373–389. Springer Berlin Heidelberg, April 2016. ISBN 978-3-662-49629-9 978-3-662-49630-5. doi: 10.1007/978-3-662-49630-5_22. URL http://link.springer.com/chapter/10.1007/978-3-662-49630-5_22.

Franz Baader and Tobias Nipkow. *Term Rewriting and All That*. Cambridge University Press, New York, NY, USA, 1998. ISBN 0-521-45520-0.

Patrick Baillot, Paolo Coppola, and Ugo Dal Lago. Light logics and optimal reduction: Completeness and complexity. *Information and Computation*, 209(2):118–142, 2011.

Gérard Boudol. The Lambda-calculus with multiplicities. Research report RR-2025, INRIA, 1993. URL https://hal.inria.fr/inria-00074646.

Alberto Carraro and Giulio Guerrieri. A Semantical and Operational Account of Call-by-Value Solvability. In Anca Muscholl, editor, *Foundations of Software Science and Computation Structures*, number 8412 in Lecture Notes in Computer Science, page 103–118. Springer Berlin Heidelberg, April 2014. ISBN 978-3-642-54829-1, 978-3-642-54830-7.

Ugo Dal Lago. Context semantics, linear logic, and computational complexity. *ACM Transactions on Computational Logic (TOCL)*, 10(4):25:1–25:32, August 2009. ISSN 1529-3785. doi: 10.1145/1555746.1555749.

Ugo Dal Lago and Patrick Baillot. On light logics, uniform encodings and polynomial time. *Mathematical Structures in Computer Science*, 16(4):713–733, August 2006. ISSN 1469-8072, 0960-1295. doi: 10.1017/S0960129506005421. URL https://www.cambridge.org/core/journals/mathematical-structures-in-computer-science/article/on-light-logics-uniform-encodings-and-polynomial-time/42C3872CADDAD642FF742AB5BAD31497.

Ugo Dal Lago, Claudia Faggian, Ichiro Hasuo, and Akira Yoshimizu. The Geometry of Synchronization. In *Proceedings of the Joint Meeting of the Twenty-Third*

EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS), CSL-LICS '14, page 35:1–35:10, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2886-9. doi: 10.1145/2603088.2603154. URL http://doi.acm.org/10.1145/2603088.2603154.

Ugo Dal Lago, Claudia Faggian, Benoit Valiron, and Akira Yoshimizu. Parallelism and Synchronization in an Infinitary Context. In *2015 30th Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, page 559–572, July 2015. doi: 10.1109/LICS.2015.58.

Vincent Danos and Thomas Ehrhard. Probabilistic coherence spaces as a model of higher-order probabilistic computation. *Information and Computation*, 209(6): 966–991, 2011. ISSN 0890-5401. doi: 10.1016/j.ic.2011.02.001.

Vincent Danos and Laurent Regnier. The structure of multiplicatives. *Archive for Mathematical Logic*, 28(3):181–203, 1989. ISSN 0933-5846. URL http://dx.doi.org/10.1007/BF01622878. 10.1007/BF01622878.

Vincent Danos and Laurent Regnier. Proof-nets and the Hilbert space. In Jean-Yves Girard, Yves Lafont, and Laurent Regnier, editors, *Advances in Linear Logic*, page 307–328. Cambridge University Press, 1995. ISBN 9780511629150. doi: 10.1017/CBO9780511629150.016.

Vincent Danos, Marco Pedicini, and Laurent Regnier. Directed virtual reductions. In Dirk van Dalen and Marc Bezem, editors, *Computer Science Logic*, number 1258 in Lecture Notes in Computer Science, page 76–88. Springer Berlin Heidelberg, 1997. ISBN 978-3-540-63172-9, 978-3-540-69201-0. URL http://link.springer.com/chapter/10.1007/3-540-63172-0_33.

Marc De Falco. The geometry of interaction of differential interaction nets. In *Logic in Computer Science, 2008. LICS'08. 23rd Annual IEEE Symposium on*, page 465–475. IEEE, June 2008. doi: 10.1109/LICS.2008.23.

Roberto Di Cosmo, Delia Kesner, and Emmanuel Polonovski. Proof Nets and Explicit Substitutions. In Jerzy Tiuryn, editor, *Foundations of Software Science and Computation Structures*, number 1784 in Lecture Notes in Computer Science, pages 63–81. Springer Berlin Heidelberg, March 2000. ISBN 978-3-540-67257-9 978-3-540-46432-7. URL http://link.springer.com/chapter/10.1007/3-540-46432-8_5. DOI: 10.1007/3-540-46432-8_5.

T. Ehrhard and L. Regnier. Differential interaction nets. *Theoretical Computer Science*, 364(2):166–195, 2006a. ISSN 0304-3975. doi: 10.1016/j.tcs.2006.08.003. URL http://www.sciencedirect.com/science/article/pii/S0304397506005299. Logic, Language, Information and Computation11th Workshop on Logic, Language, Information and Computation.

Thomas Ehrhard and Laurent Regnier. The differential lambda-calculus. *Theoretical Computer Science*, 309(1):1–41, 2003. ISSN 0304-3975. doi: 10.1016/S0304-3975(03)00392-X.

Thomas Ehrhard and Laurent Regnier. Differential interaction nets. *Electronic Notes in Theoretical Computer Science*, 123:35–74, 2005.

Thomas Ehrhard and Laurent Regnier. Böhm Trees, Krivine's Machine and the Taylor Expansion of Lambda-Terms. In Arnold Beckmann, Ulrich Berger, Benedikt Löwe, and JohnV. Tucker, editors, *Logical Approaches to Computational Barriers*, volume 3988 of *Lecture Notes in Computer Science*, pages 186–197. Springer Berlin Heidelberg, 2006b. ISBN 978-3-540-35466-6. doi: {10.1007/11780342_20}.

Thomas Ehrhard and Laurent Regnier. Uniformity and the Taylor expansion of ordinary lambda-terms. *Theoretical Computer Science*, 403(2):347–372, 2008. doi: 10.1016/j.tcs.2008.06.001.

M. Fernández, I. Mackie, and F.-R. Sinot. Closed reduction: explicit substitutions without alpha-conversion. *Mathematical Structures in Computer Science*, 15(2):343–381, April 2005. ISSN 1469-8072, 0960-1295. doi: 10.1017/S0960129504004633.

Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50(1):1–101, 1987. ISSN 0304-3975. doi: 10.1016/0304-3975(87)90045-4.

Jean-Yves Girard. Geometry of interaction I: Interpretation of System F. *Studies in Logic and the Foundations of Mathematics*, 127:221–260, 1989. doi: 10.1016/S0049-237X(08)70271-4.

Jean-Yves Girard. Light linear logic. In *Logic and computational complexity*, page 145–176. Springer, 1995.

Georges Gonthier, Martìn Abadi, and Jean-Jacques Lévy. The geometry of optimal lambda reduction. In *Proceedings of the 19th ACM SIGPLAN SIGACT symposium on Principles of programming languages*, POPL '92, page 15–26. ACM, 1992a. doi: 10.1145/143165.143172.

Georges Gonthier, Martìn Abadi, and Jean-Jacques Lévy. Linear Logic Without Boxes. In *LICS*, page 223–234. IEEE Computer Society, 1992b. ISBN 0-8186-2735-2. doi: 10.1109/LICS.1992.185535.

Giulio Guerrieri and Lorenzo Tortora de Falco. A new point of view on the Taylor expansion of proof-nets and uniformity. Presented at Linearity 2014, the third International Workshop on Linearity, Vienna, July 2014.

Stefano Guerrini. A General Theory of Sharing Graphs. *TCS: Theoretical Computer Science*, 227, 1999.

Stefano Guerrini, Simone Martini, and Andrea Masini. Coherence for sharing proof-nets. *Theoretical computer science*, 294(3):379–409, 2003.

Stefano Guerrini, Thomas Leventis, and Marco Solieri. Deep into optimality – complexity and correctness of sharing implementation of bounded logics. Presented at DICE 2012, Third International Workshop on Developments in Implicit Complexity, Tallin, 2012. URL http://ms.xt3.it/research/DeepIntoOptimality_2012-02-17.pdf.

Vinod Kathail. *Optimal Interpreters for lambda-calculus based functional languages*. PhD thesis, Massachusetts Institute of Technology, 1990.

Jean-Louis Krivine. A call-by-name lambda-calculus machine. *Higher-Order and Symbolic Computation*, 20(3):199–207, 2007. ISSN 1388-3690, 1573-0557. doi: 10.1007/s10990-007-9018-9.

Yves Lafont. Interaction Nets. In *Proceedings of the 17th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '90, page 95–108. ACM, 1990. ISBN 0-89791-343-4. doi: 10.1145/96709.96718. URL http://doi.acm.org/10.1145/96709.96718.

John Lamping. An algorithm for optimal lambda calculus reduction. In *17th ACM SIGPLAN SIGACT Symposium on Principles of Programming Languages*, page 16–30. ACM, 1989. doi: 10.1145/96709.96711.

Olivier Laurent. A Token Machine for Full Geometry of Interaction (Extended Abstract). In Samson Abramsky, editor, *Typed Lambda Calculi and Applications '01*, volume 2044 of *Lecture Notes in Computer Science*, page 283–297. Springer, May 2001.

Olivier Laurent. *Étude de la polarisation en logique*. phdthesis, Université de la Méditerranée - Aix-Marseille II, March 2002. URL https://tel.archives-ouvertes.fr/tel-00007884/document.

Jean-Jacques Lévy. *Réductions Correctes et Optimales dans le Lambda Calcul*. PhD thesis, Université Paris VII, 1978.

Jean-Jacques Lévy. Optimal Reductions in the Lambda-Calculus. In J. P. Seldin and J. R. Hindley, editors, *To H. B. Curry: Essays on Combinatory Logic, Lambda Calculus, and Formalism*. Academic Press, 1980.

Ian Mackie. The geometry of interaction machine. In *POPL 95 Proceedings of the 22nd ACM SIGPLAN SIGACT symposium on Principles of programming languages*, page 198–208. ACM, 1995. doi: 10.1145/199448.199483.

Ian Mackie. Efficient lambda-Evaluation with Interaction Nets. In Vincent van Oostrom, editor, *Rewriting Techniques and Applications*, number 3091 in Lecture Notes in Computer Science, pages 155–169. Springer Berlin Heidelberg, 2004. ISBN 978-3-540-22153-1 978-3-540-25979-4. URL http://link.springer.com/chapter/10.1007/978-3-540-25979-4_11.

Harry G. Mairson. From hilbert spaces to dilbert spaces: Context semantics made simple. In *In 22 Nd Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 2–17. Springer-Verlag, 2002.

Damiano Mazza. Infinitary Affine Proofs. *Mathematical Structures in Computer Science*, FirstView:1–22, 12 2015a. ISSN 1469-8072. doi: 10.1017/S0960129515000298. URL http://journals.cambridge.org/article_S0960129515000298.

Damiano Mazza. Simple Parsimonious Types and Logarithmic Space. In Stephan Kreutzer, editor, *24th EACSL Annual Conference on Computer Science Logic (CSL 2015)*, volume 41 of *Leibniz International Proceedings in Informatics (LIPIcs)*, page 24–40, Dagstuhl, Germany, 2015b. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. ISBN 978-3-939897-90-3. doi: 10.4230/LIPIcs.CSL.2015.24. URL http://drops.dagstuhl.de/opus/volltexte/2015/5405.

Damiano Mazza and Michele Pagani. The Separation Theorem for Differential Interaction Nets. In Nachum Dershowitz and Andrei Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning*, number 4790 in Lecture Notes in Computer Science, page 393–407. Springer Berlin Heidelberg, January 2007. ISBN

978-3-540-75558-6 978-3-540-75560-9. URL http://link.springer.com/chapter/10.1007/978-3-540-75560-9_29.

Damiano Mazza and Kazushige Terui. Parsimonious Types and Non-uniform Computation. In Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann, editors, *Automata, Languages, and Programming, Proceedings of ICALP*, volume 9135 of *Lecture Notes in Computer Science*, page 350–361. Springer Berlin Heidelberg, 2015. ISBN 978-3-662-47665-9. doi: 10.1007/978-3-662-47666-6_28. URL http://dx.doi.org/10.1007/978-3-662-47666-6_28.

Michele Pagani and Christine Tasson. The Inverse Taylor Expansion Problem in Linear Logic. In *24th Annual IEEE Symposium on Logic In Computer Science, 2009. LICS '09*, page 222–231, August 2009. doi: 10.1109/LICS.2009.35.

Michele Pagani and Lorenzo Tortora de Falco. Strong normalization property for second order linear logic. *Theoretical Computer Science*, 411(2):410–444, January 2010. ISSN 0304-3975. doi: 10.1016/j.tcs.2009.07.053. URL http://www.sciencedirect.com/science/article/pii/S0304397509005180.

Michele Pagani and Paolo Tranquilli. Parallel Reduction in Resource Lambda-Calculus. In Zhenjiang Hu, editor, *Programming Languages and Systems, 7th Asian Symposium (APLAS 2009)*, volume 5904 of *Lecture Notes in Computer Science.*, page 226–242, 2009.

Michele Pagani, Peter Selinger, and Benoit Valiron. Applying Quantitative Semantics to Higher-Order Quantum Computing. In P. Sewell, editor, *The 41th Annual ACM SIGPLAN SIGACT Symposium on Principles of Programming Languages, POPL14, San Diego, USA*. ACM, 2014. doi: 10.1145/2535838.2535879.

Marco Pedicini and Francesco Quaglia. PELCR: Parallel Environment for Optimal Lambda-calculus Reduction. *ACM Transactions on Computational Logic*, 8, 3(3), July 2007. ISSN 1529-3785. doi: 10.1145/1243996.1243997. URL http://doi.acm.org/10.1145/1243996.1243997.

Marco Pedicini, Giulio Pellitta, and Mario Piazza. Sequential and Parallel Abstract Machines for Optimal Reduction. In *Preproceedings of the 15th Symposium on Trends in Functional Programming (TFP2014)*, 2014.

Matthieu Perrinel. On Context Semantics and Interaction Nets. In *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, CSL-LICS '14, page 73:1–73:10, New York, NY, USA, 2014. ACM. ISBN 978-1-4503-2886-9. doi: 10.1145/2603088.2603155. URL http://doi.acm.org/10.1145/2603088.2603155.

Jorge Sousa Pinto. Parallel Implementation Models for the Lambda-Calculus Using the Geometry of Interaction. In Samson Abramsky, editor, *Typed Lambda Calculi and Applications*, volume 2044 of *Lecture Notes in Computer Science*, page 385–399. Springer Berlin Heidelberg, 2001. ISBN 978-3-540-41960-0. doi: 10.1007/3-540-45413-6_30.

Laurent Regnier. *Lambda-calcul et réseaux*. PhD thesis, Université Paris VII, 1992.

Marco Solieri. Ottimalità dell'ottimalità: complessità della riduzione su grafi di condivisione. Master's thesis, Alma Mater Studiorum, Università di Bologna, November 2011. URL http://amslaurea.cib.unibo.it/2744/1/Solieri-OttimalitaDellOttimalita.pdf.

Marco Solieri. Geometry of Resource Interaction – A Minimalist Approach. In Sandra Alves and Iliano Cervesato, editors, *Proceedings Third International Workshop on Linearity, Vienna, Austria, 13th July, 2014*, volume 176 of *Electronic Proceedings in Theoretical Computer Science*, page 79–94. Open Publishing Association, 2015. doi: 10.4204/EPTCS.176.7. URL http://arxiv.org/abs/1502.04775.

Marco Solieri. Geometry of Resource Interaction and Taylor-Ehrhard-Regnier Expansion of Lambda-terms - A Minimalist Approach. *Mathematical Structures in Computer Science*, 2016. URL http://ms.xt3.it/research/GoRI-Taylor_2016-09-07.pdf. To appear.

Richard Statman. The Typed lambda-Calculus is not Elementary Recursive. *Theoretical Computer Science*, 9:73–81, 1979. doi: 10.1016/0304-3975(79)90007-0.

Paolo Tranquilli. Intuitionistic differential nets and lambda-calculus. *Theoretical Computer Science*, 412(20):1979–1997, April 2011. ISSN 03043975. doi: 10.1016/j.tcs.2010.12.022. URL http://linkinghub.elsevier.com/retrieve/pii/S0304397510007139.

Lionel Vaux. On Linear Combinations of Lambda-Terms. In Franz Baader, editor, *Term Rewriting and Applications*, volume 4533 of *Lecture Notes in Computer Science*, page 374–388. Springer, 9 2007. ISBN 978-3-540-73447-5. doi: 10.1007/978-3-540-73449-9_28. URL http://dx.doi.org/10.1007/978-3-540-73449-9_28.