Université d'Aix-Marseille

Sur la syntaxe de la sémantique quantitative

Mémoire présenté par

Lionel Vaux Auclair

en vue d'obtenir

l'Habilitation à Diriger des Recherches
(spécialité : Mathématiques)

le 18 novembre 2021 devant le jury composé de :

Robin Cockett	University of Calgary
Ugo Dal Lago	Università di Bologna
Delia Kesner	Université de Paris
Guy McCusker	University of Bath
Paul-André Melliès	Université de Paris
Myriam Quatrini	Université d'Aix-Marseille
Laurent Regnier	Université d'Aix-Marseille

après avis des rapporteurs :

Robin Cockett	University of Calgary
Delia Kesner	Université de Paris
Guy McCusker	University of Bath

Version du 22 novembre 2021.

À Vir, encore.

[...]
speak white
parlez-nous production, profits et pourcentages
speak white
c'est une langue riche
pour acheter
mais pour se vendre
mais pour se vendre à perte d'âme
mais pour se vendre
[...]

Michèle Lalonde, *Speak White*, 1974

Ce mémoire, rédigé en français, est essentiellement constitué de l'inclusion de trois articles de recherche, en anglais donc.

Il présente mes travaux des dernières années, suivant une ligne directement issue de préoccupations datant de ma thèse de doctorat, mais qui a porté ses premiers fruits au milieu des années 2010. Il s'agit :

— de raffiner l'analyse de la normalisation offerte par le développement de Taylor des λ-termes pour la ramener au niveau de la β-réduction (l'étape élémentaire de la procédure de normalisation) ;

— d'étendre cette analyse dans un cadre non-uniforme, susceptible de prendre en compte par exemple une forme de non-déterminisme calculatoire ;

— le tout en conservant la nature quantitative du développement de Taylor.

Le premier point est un pré-requis pour les suivants car, dans un cadre non-uniforme, la normalisation peut engendrer des sommes infinies de coefficients, susceptibles de diverger. Ce phénomène disparait si on se restreint à la β-réduction, dont l'analyse par le développement de Taylor est toujours finitaire.

Les trois articles inclus dans le mémoire constituent une forme d'aboutissement de cette ligne de recherche :

— le premier, paru en 2019 [5], résout la question pour le λ-calcul algébrique, qui étend le λ-calcul ordinaire avec des combinaisons linéaires finies de λ-termes ;

— le second, écrit avec Jules Chouquet [6], permet d'adapter ces résultats à la syntaxe, bien plus permissive, des réseaux de démonstration de la logique linéaire ;

— le troisième, écrit avec Federico Olimpieri [7] revisite les résultats originels d'Ehrhard et Regnier sur la combinatoire du développement de Taylor en λ-calcul et montre comment, en se concentrant sur la β-réduction, on peut à la fois simplifier une partie de leur démarche et l'étendre à un λ-calcul non-déterministe à peu de frais.

Chaque article est inclus comme un chapitre du mémoire. Le tout est précédé d'un chapitre d'introduction, rédigé à la première personne et en français, qui présente le contexte scientifique dans lequel ces résultats s'insèrent.

La bibliographie qui clôt le mémoire est structurée en trois parties : d'abord les publications dont je suis le coauteur, suivies des thèses soutenues que j'ai coencadrées, et enfin le reste des sources citées dans le mémoire. Dans le texte du mémoire, les références numériques (par exemple [5]) renvoient aux deux premières catégories, tandis que les références alphabétiques (par exemple [ER03]) renvoient à la dernière.

Mercis

Outre les *usual suspects* des remerciements — membres du jury, collègues, amis, famille : ils et elles se reconnaitront — je voudrais ici remercier très chaleureusement : Michele Alberti, Thomas Leventis, Jules Chouquet, Federico Olimpieri et Zeinab Galal, qui ont eu la sagesse de ne pas attendre que je soutienne mon habilitation pour soutenir leurs thèses respectives ; Rémy Cerda, Lison Blondeau-Patissier et Giulia Manara, à qui cette attente est épargnée ; mais aussi Étienne Miquey, Gabriele Muciaccia, Marine Giorgis et Yaelle Maman, Antoine Mottet, Nicolas Jeannerod, Matteo de Leo, Julien Gabet, Auriane Bertrand et Suliman Er, Axel Kerinec, Edoardo Rivetti, et Jean-Baptiste Vienney, que j'ai eu la chance d'accompagner pour quelques semaines ou mois. J'ai sans doute plus appris avec chacune et chacun que ce que j'ai prétendu apprendre à chacune et chacun.

Et puis Ulysse et Marius, qui m'ont appris autre chose.

Table des matières

Chapitre 1

Introduction

1.1 Un peu de contexte

Mon domaine de prédilection est la théorie de la démonstration, éclairée par la correspondance de Curry–Howard entre preuves et programmes. Cette approche consiste en l'établissement d'une sorte de dictionnaire bilingue logique–programmation :

$$\begin{array}{rcl} \text{formule} & \sim & \text{type} \\ \text{preuve} & \sim & \text{programme} \\ \text{règle d'inférence} & \sim & \text{règle de formation syntaxique} \\ \text{normalisation} & \sim & \text{évaluation} \\ & \cdots & \end{array}$$

Le procédé de normalisation des preuves le plus célèbre est l'élimination des coupures du calcul des séquents, due à Gentzen pour démontrer la cohérence de l'arithmétique par des moyens élémentaires (sans bien sûr violer les théorèmes d'incomplétude de Gödel) ; mais c'est seulement à la fin des années 1960 qu'une correspondance précise apparait, entre la déduction naturelle (un système introduit par Gentzen, comme le calcul des séquents, mais plus proche du raisonnement usuel) et le λ-calcul (un formalisme introduit par Church pour capturer la notion de fonction calculable, et le paradigme de référence pour la programmation fonctionnelle).

La principale qualité de cette correspondance est de servir de véhicule pour des pollinisations croisées entre logique et programmation. Par exemple, elle a permis l'adaptation à la logique de techniques d'analyse des programmes. C'est notamment le cas de la sémantique dénotationnelle, qui consiste à regarder les programmes comme représentant des fonctions, et à étudier les propriétés des fonctions ainsi représentables. La logique linéaire de Girard est le fruit d'une telle analyse en logique : elle reflète dans un système logique la structure fine de la sémantique quantitative, initialement introduite par Girard comme modèle du système F, c'est-à-dire du λ-calcul typé avec polymorphisme paramétré.

Sémantique quantitative. La notion de sémantique quantitative consiste en l'interprétation des termes du λ-calcul par des séries entières généralisées. L'idée est de voir un programme comme une superposition de monômes, chacun capturant une approximation finie de son comportement calculatoire : le degré du monôme est relié au nombre de fois où le programme utilise son argument dans cette approximation.

Par la correspondance de Curry-Howard entre preuves et programmes, les démonstrations de la logique intuitionniste peuvent ainsi être vues comme des fonctions analytiques, auxquelles un opérateur de dérivation permet d'associer des approximations linéaires : par raffinements successifs, cette interprétation a mené Girard à l'introduction de la logique linéaire, un paradigme aujourd'hui devenu incontournable en théorie de la démonstration.

La présentation originale du concept de sémantique quantitative par Girard au début des années 1980 [Gir88] reposait sur des outils catégoriques assez délicats, à la suite de ses travaux sur les dilatateurs. Au début des années 2000, Ehrhard a introduit de nouveaux modèles dénotationnels [Ehr05] permettant de reformuler cette sémantique quantitative pour la logique linéaire et le λ-calcul typé, dans un cadre algébrique plus standard : les types sont des espaces vectoriels topologiques particuliers, les preuves de la logique linéaire sont des morphismes linéaires et continus, et les termes du λ-calcul sont définis par des séries entières.

Développement de Taylor des programmes et des preuves. Avec ces nouveaux modèles, il devient aisé de transposer aux objets de la logique et du calcul les outils et techniques du calcul différentiel. En particulier, les morphismes sont infiniment différentiables et admettent un développement de Taylor en tout point. En relisant ces propriétés au niveau syntaxique, Ehrhard et Regnier ont introduit le λ-calcul différentiel, une extension du λ-calcul avec un opérateur formel de dérivation [ER03]. Dans cette théorie, le développement de Taylor devient un opérateur traduisant les λ-termes purs en sommes pondérées de λ-termes à ressources (des λ-termes différentiels ne comportant plus que des applications au terme nul) obtenues en remplaçant chaque application par son développement de Taylor en 0 [ER08]. Du fait de leur linéarité, les termes à ressources conservent une dynamique, mais celle-ci est très simple et finitaire : la taille des termes décroit avec la réduction, et les termes à ressources sont donc tous fortement normalisables.

Cette approche a été très fructueuse. Ehrhard et Regnier ont en particulier montré que le développement de Taylor d'un λ-terme pur est toujours normalisable, et que sa forme normale correspond exactement à l'arbre de Böhm du λ-terme — une notion généralisée de forme normale pour les λ-termes [ER06a].

Une des retombées les plus notables de cette approche est qu'elle permet d'établir des caractérisations de propriétés opérationnelles (typiquement de normalisabilité) en raisonnant inductivement sur les éléments du développement de Taylor [Ehr12a ; BM20], alors que l'état de l'art précédent nécessitait un raisonnement par réductibilité par exemple. Les termes à ressources (du moins ceux dont la forme normale est non nulle) peuvent d'ailleurs être vus comme des dérivations de typage dans un système de types avec intersection non idempotente [Car07 ; Car18a]. [1]

Le développement de Taylor est ainsi une structure intermédiaire entre la syntaxe des termes

1. Au passage, cette remarque de de Carvalho a renouvelé l'intérêt pour les systèmes de types avec intersection non idempotente. En s'affranchissant du lien avec le développement de Taylor mais en affinant la relation entre typage et temps d'exécution, cette ligne de travail a produit des systèmes qui encadrent très précisément le nombre d'étapes de réduction menant à une forme normale suivant une certaine stratégie [BG13 ; BKV17 ; AGK20].

et leur sémantique (en particulier celle des arbres de Böhm), intimement liée à la dynamique du λ-calcul en tant que langage de programmation.

1.2 Travaux

Thèse : Logique linéaire différentielle et polarisation. Au cours de ma thèse j'ai étudié la logique linéaire différentielle, une extension de la logique linéaire due à Ehrhard et Regnier[ER06b], qui introduit une symétrie sur les règles structurelles.

Les règles structurelles sont celles qui permettent d'utiliser une hypothèse autant de fois que nécessaire : en logique linéaire ces règles sont restreintes à certaines formules, dites exponentielles. Les versions duales de ces règles, introduites par Ehrhard et Regnier, sont issues d'une structure particulière sur formules exponentielles dans le modèle des espaces de finitude dû à Ehrhard [Ehr05] : les règles costructurelles sont celles qui permettent de dériver formellement les λ-termes [ER03].

En appliquant un procédé de polarisation à la manière d'Olivier Laurent [Lau02], j'ai pu proposer une interprétation calculatoire de règles costructurelles généralisées, sous la forme d'un produit de convolution sur les continuations (une abstraction des piles d'exécution pour les programmes), reflétant la dynamique de l'élimination des coupures en logique linéaire différentielle [9 ; 11].

Je n'ai plus travaillé sur cet axe de recherche depuis 2009. Si on peut reprocher à ces travaux une forme d'idiosyncrasie, certaines contributions techniques ont eu une descendance. Par exemple, dans une première étape consistant à étendre au $\lambda\mu$-calcul de Parigot le principe syntaxique de dérivation [1], j'ai simplifié la syntaxe du λ-calcul différentiel d'Ehrhard et Regnier, et c'est cette présentation qui fait maintenant référence. Mais la retombée la plus notable est la suivante.

Non déterminisme quantitatif. La notion de dérivée n'a de sens qu'en présence de sommes, et plus généralement de combinaisons linéaires. Le traitement de ces combinaisons linéaires dans la syntaxe du λ-calcul différentiel posait un certain nombre de difficultés techniques, qui n'avaient en fait pas grand chose à voir avec la dérivation. Ces difficultés m'ont poussé à développer une théorie de la réécriture d'ordre supérieur en présence de combinaisons linéaires de termes, en introduisant le λ-calcul algébrique [10 ; 2].[2] J'ai ainsi pu démontrer que les propriétés essentielles de confluence et de terminaison de l'évaluation en λ-calcul ne pouvaient être préservées qu'au prix de certaines restrictions : en l'absence de contrainte, toute extension de la β-réduction (la relation de réécriture définissant l'évaluation en λ-calcul) rendue compatible avec les équations d'espace vectoriel est triviale.

Au moment de leur développement, j'avais considéré ces travaux comme assez secondaires, de l'ordre du simple nettoyage, car l'essentiel de la nouveauté de la logique linéaire différentielle était ailleurs. Ils ont cependant joué un rôle important dans le développement d'une de mes principales direction de recherche par la suite : l'étude du non-déterminisme calculatoire dans un cadre quantitatif. On peut en effet considérer la somme de deux termes comme un choix

2. Celui-ci est bien mal nommé : il aurait été plus éclairant et honnête de l'appeler λ-calcul vectoriel ou λ-calcul avec combinaisons linéaires par exemple. Il est malheureusement un peu tard pour revenir là-dessus.

non-déterministe : au lieu d'étudier les comportements possibles d'un objet comme de simples alternatives, on cherche à quantifier la part de chacun des comportements atomiques qui constituent le comportement global. [3] Le λ-calcul algébrique en est en quelque sorte le langage universel, sans contraindre *a priori* la structure ni la signification de l'opérateur de choix.

À peu près à la même époque, Arrighi et Dowek avaient d'ailleurs introduit un calcul très similaire [AD08], avec une motivation toute différente : la représentation d'algorithmes quantiques. Ce qui distingue les deux approches, c'est le traitement de la linéarité, qui reflète le paradigme d'évaluation sous-jacent : appel par nom ou par valeur [Ass+14].

Propriétés opérationnelles en λ-calcul algébrique. C'est dans cette ligne de recherche que s'inscrivent plusieurs des travaux d'étudiants que j'ai accompagnés. Ainsi, un des apports de la thèse de Michele Alberti [23] était une étude fine du phénomène de dégénérescence de la β-réduction en présence de coefficients négatifs, qui lui a permis d'en proposer un premier contournement. S'appuyant sur une idée esquissée par Ehrhard et Regnier [ER03, Section 5], il a introduit une notion de normalisabilité plus robuste, neutralisant les identités problématiques entre combinaisons linéaires. Il a pu en déduire que la restriction de la β-réduction aux termes canoniques (c'est-à-dire ceux pour lesquels les combinaisons linéaires sont systématiquement mises sous forme canonique), bien que non confluente en général, permet bien d'atteindre ces formes normales, qui sont uniques lorsqu'elles existent.

Au passage, il a identifié un défaut de standardisation de la β-réduction du λ-calcul algébrique [23, Section 3.2]. En λ-calcul, toute suite de réductions peut être standardisée : on peut exiger que les réductions les plus externes (les réductions dites *de tête*) soient effectuées en premier. Ceci échoue en λ-calcul algébrique car une réduction interne peut, modulo les équations algébriques, faire apparaitre une somme en tête, dont on peut ensuite ne réduire qu'un des membres.

La thèse de Thomas Leventis [24] était consacrée au λ-calcul probabiliste, qu'on peut voir comme un sous-système du λ-calcul algébrique à coefficients réels positifs. Il a développé une notion d'arbre de Böhm probabiliste, dont il a démontré qu'elle capturait l'équivalence observationnelle probabiliste. Ce travail nécessitait entre autres d'obtenir un résultat de standardisation pour le λ-calcul probabiliste. L'obstacle mentionné plus haut persiste, mais il se trouve qu'une forme affaiblie de standardisation suffit : toute suite de réductions peut être prolongée en une suite de réductions standardisable. Leventis a en fait pu établir ce dernier résultat dans le cadre général du λ-calcul algébrique [Lev19] : sa preuve repose sur une analyse pointue de la réduction en λ-calcul algébrique, mettant en jeu un choix précis de règles de réécriture des combinaisons linéaires.

Une conséquence directe de la standardisation faible est que, pour tout terme normalisable, la forme normale de ce terme est atteinte par la réduction gauche (la réduction de tête suivie, inductivement, de la réduction gauche dans les arguments de la variable de tête). En particulier l'équivalence induite par la normalisation sur les λ-termes usuels (sans sommes) est la même en λ-calcul algébrique que dans le λ-calcul pur. C'est un résultat de conservativité faible : la conservativité forte serait que la β-équivalence (l'équivalence induite par la réduction) soit la même pour les λ-termes purs dans les deux calculs.

3. C'est l'idée qu'on trouve dans le premier papier de Girard sur la sémantique quantitative [Gir88].

Il se trouve que ce résultat est également valide sous certaines conditions mais son histoire est mouvementée. Ehrhard et Regnier l'avaient annoncée dans le λ-calcul différentiel comme une conséquence directe de la confluence [ER03, Proposition 19], mais c'est contredit dans le cas général par la dégénérescence de la β-équivalence en présence de coefficients négatifs, sans nier la confluence. Dans le cas où on se restreint à des coefficients positifs, j'avais cru régler la question pour le $\lambda\mu$-calcul différentiel et repris le même argument *verbatim* pour le λ-calcul algébrique ; malheureusement, cette « preuve » est irrémédiablement fausse. [4]

Ce n'est que récemment, durant le stage de recherche d'Axel Kerinec en 2019, que nous avons finalement établi la conservativité forte avec la seule hypothèse de positivité. J'ai présenté ce résultat aux journées 2019 du groupe de travail Scalp du GDR-IM, mais il nous reste à le mettre en forme pour publication.

Modèle relationnel et finitaire pour les types de données. En parallèle de l'approche syntaxique du non-déterminisme quantitatif, j'ai débuté en 2008 une collaboration avec Christine Tasson visant à étendre l'interprétation de la logique linéaire et du λ-calcul dans les espaces de finitude à des langages permettant de calculer sur les types de données usuels (entiers, listes, *etc.*). La difficulté provient du fait que cette interprétation est limitée à un cadre simplement typé : or les astuces usuelles pour coder les types de données dans le λ-calcul ou la logique linéaire sortent de ce cadre. Il faut donc les introduire explicitement.

J'ai établi un premier résultat en ce sens, en démontrant que l'interprétation du système T de Gödel (programmation fonctionnelle avec un type des entiers naturels) dans un modèle relationnel avec entiers paresseux était finitaire [15 ; 3].

En cherchant à étendre cette approche à d'autres types de données, Tasson et moi-même avons mis au jour une construction très générale d'espace de finitude [4] : notre résultat permet non seulement de construire explicitement les objets représentant les types de données usuels, mais il assure également la fonctorialité de cette construction et, sous des conditions raisonnables, l'existence de plus petits points fixes pour les foncteurs obtenus. Or une solution standard pour interpréter les types de données est de considérer les plus petits points fixes de foncteurs polynomiaux, qui se trouvent être des instances de notre construction.

Normalisabilité du développement de Taylor en λ-calcul algébrique. La propriété centrale qui assure la pertinence du développement de Taylor des λ-termes est sa compatibilité avec la normalisation :

- le développement de Taylor d'un λ-terme est toujours normalisable ;
- si le terme lui-même est normalisable, alors la forme normale du développement de Taylor est le développement de Taylor de la forme normale du terme ;
- et dans le cas général, la forme normale du développement de Taylor est le développement de Taylor de l'arbre de Böhm du terme.

La preuve de ce résultat par Ehrhard et Regnier [ER08 ; ER06a] dépend fortement d'une propriété d'uniformité qui est validée par le λ-calcul pur, mais violée par ses extensions non-déterministes.

4. Pour la lectrice curieuse : le problème est le cas du redex, dans le lemme de conservativité de la β-réduction [1, Lemma 3.30] [2, Lemma 3.20].

D'ailleurs, en général, le développement de Taylor d'un terme algébrique n'est pas normalisable : le long d'un chemin de réduction infini, on peut accumuler des sommes non convergentes de coefficients. Un premier contournement de cette limitation a été établi par Ehrhard, qui a démontré que, dans un cadre typé (y compris au second ordre, dans une extension du système F), le développement de Taylor des λ-termes algébriques reste normalisable [Ehr10]. Il a introduit une structure de finitude sur les ensembles de termes du λ-calcul à ressources telle que : pour toute combinaison linéaire infinie de termes à ressources avec un support finitaire, on peut calculer une forme normale. Il démontre ensuite par une méthode de réductibilité que les termes typables ont un développement de Taylor à support finitaire.

Ceci vient soutenir l'intuition que la finitude caractérise l'absence de réductions infinies. Avec Michele Pagani et Christine Tasson, nous avons d'ailleurs adapté cette approche pour obtenir une correspondance exacte : on peut raffiner la notion de finitude de sorte qu'un λ-terme est fortement normalisable si et seulement si son développement est finitaire [12].

Reste qu'à la fois travail d'Ehrhard et le nôtre se limitent à établir la normalisabilité du développement de Taylor sous certaines hypothèses : ils ne disent rien de sa possible commutation avec la normalisation. La preuve d'Ehrhard et Regnier reposait sur un calcul explicite des coefficients dans la forme normale des termes à ressources issus du développement de Taylor : en l'absence d'uniformité, cette technique devient inaccessible.

Simulation de la β-réduction dans le développement de Taylor. Établir la commutation dans le cas général demande de changer de point de vue : il s'agit d'établir que la forme normale du développement de Taylor, dans les cas où elle existe, définit une sémantique dénotationnelle, c'est-à-dire qu'elle est invariante par β-réduction.

Là où la normalisation est *a priori* un processus infinitaire, qui doit être maîtrisé par le typage, l'uniformité, ou d'autres contraintes pour converger, j'ai pu montrer que la β-réduction est essentiellement finitaire et qu'on peut la simuler à travers le développement de Taylor [13 ; 5]. Plus précisément, j'ai considéré une notion de réduction parallèle sur les termes à ressources et montré que :

- étant donné un terme à ressources t et un λ-terme algébrique M, il y a un nombre fini d'éléments s du support du développement de Taylor de M tels que t apparait dans un réduit parallèle de s ;
- on peut donc étendre cette réduction parallèle aux combinaisons linéaires infinies que sont les développements de Taylor, sans aucune restriction sur les λ-termes algébriques considérés ;
- pour tout pas de β-réduction de M à N (et on peut même considérer la β-réduction parallèle ici), le développement de Taylor de M se réduit en celui de N.

Il s'ensuit directement que, sur les termes normalisables, le développement de Taylor est compatible avec la normalisation.

Mieux : on peut affaiblir la notion de finitude introduite par Ehrhard sur les termes à ressources, pour capturer exactement le fait que la normalisation d'une combinaison linéaire infinie ne produit que des sommes finies de coefficients. Les termes dont le développement de Taylor est finitaire en ce sens incluent à la fois tous les λ-termes purs et les termes algébriques normalisables. Sur ces termes, la normalisation du développement de Taylor induit une sémantique dénotationnelle qui est une généralisation commune des arbres de Böhm pour le λ-calcul pur et du modèle des espaces de finitude pour le λ-calcul algébrique typé. Ces travaux constituent le Chapitre 2 du présent mémoire.

Développement de Taylor dans les réseaux de la logique linéaire. La technique que j'ai mise au point pour établir la compatibilité du développement de Taylor avec la réduction est suffisamment robuste pour être exploitée dans un cadre plus large. Avec Jules Chouquet, nous l'avons étendue aux réseaux de démonstration de la logique linéaire : ces derniers admettent un développement de Taylor dans les réseaux à ressources, qui forment le fragment multilinéaire de la logique linéaire différentielle [ER06b ; Ehr18]. Il est à noter que, même en l'absence de non déterminisme, les réseaux sont fondamentalement non uniformes : Tasson a montré qu'on ne peut pas définir de relation de cohérence sur les réseaux à ressources telle que le développement de Taylor d'un réseau forme toujours une clique [Tas09, Section V.4.1]. Il aurait donc été impossible d'adapter la technique d'Ehrhard et Regnier dans ce cadre.

Pour simuler l'élimination des coupures en logique linéaire à travers le développement de Taylor, il faut là encore considérer une version parallèle de la réduction dans le langage cible. Et à nouveau, la première étape consiste à démontrer que, étant donné un réseau à ressources q et un réseau de la logique linéaire R, il y a un nombre fini d'éléments r du développement de Taylor de R tels que q apparait dans un réduit parallèle de r. Nous avons d'abord établi ce résultat pour les réseaux sans affaiblissements [14], ce qui permettait une simplification technique : notre approche repose sur une analyse de l'évolution des chemins de correction [5] à travers l'élimination parallèle des coupures, qui sont plus simples à décrire dans ce cas. Dans le cas général, que nous avons traité ensuite [6] et qui constitue le Chapitre 3 du présent mémoire, il faut introduire une structure supplémentaire de sauts depuis les affaiblissements.

Nos résultats permettent donc d'étendre l'élimination parallèle des coupures aux combinaisons linéaires infinies de réseaux à ressources obtenus par développement de Taylor, ce qui est la notion adéquate pour simuler celle de la logique linéaire. Chouquet a effectivement complété la commutation [25] en établissant les identités nécessaires sur les coefficients : là encore on obtient un modèle dénotationnel à travers lequel se factorisent les sémantiques quantitatives de la logique linéaire qui valident le développement de Taylor.

Un groupoïde de permutations sur les termes à ressources. La dernière décennie a vu se développer une convergence entre d'une part l'approche de la sémantique quantitative du λ-calcul et de la logique linéaire, et d'autre part la notion d'espèce de structures initialement introduite par Joyal comme une approche catégorique de la combinatoire [Joy86]. La sémantique quantitative de Girard était basée sur une notion de foncteurs normaux tout-à-fait similaire à celle des foncteurs analytiques de Joyal, un lien déjà exploité par Hasegawa [Has02]. Plus récemment, Fiore, Gambino, Hyland et Winskel ont introduit un modèle bicatégorique du λ-calcul, basé sur une généralisation des espèces de structures [Fio+08].

On peut voir ces espèces de structures généralisées comme formant une version bicatégorique du modèle relationnel du λ-calcul. Les liens étroits qu'entretiennent le modèle relationnel, le développement de Taylor et les systèmes de types avec intersections sont connus depuis les

5. C'est-à-dire les chemins dans les graphes de correction du critère de Danos–Regnier.

travaux de de Carvalho sur la caractérisation du temps d'exécution en λ-calcul [Car07 ; Car18a]. Il semblait donc naturel d'explorer ces liens dans le cadre bicatégorique offert par les espèces généralisées : c'était le sujet de thèse de Federico Olimpieri.

Il se trouve que Tsukada, Asada et Ong avaient déjà établi un lien entre espèces généralisées et développement de Taylor [TAO17], en introduisant une notion de développement de Taylor rigide sur lequel agissent des isomorphismes de types : ils démontrent que le modèle bicatégorique du λ-calcul ainsi obtenu est naturellement isomorphe à celui des espèces de structures. Leur approche est limitée à un cadre typé ; de plus, elle s'éloigne du λ-calcul à ressources standard en imposant de nommer explicitement chaque occurrence de variable (en particulier l'abstraction est polyadique) ; enfin, pour prendre en compte une forme de non-déterminisme, ils équipent les approximations linéaires que sont les termes à ressources de marqueurs explicites pour chaque branche de l'opérateur de choix.

Avec Olimpieri, nous avons exploité certaines de ces idées en les adaptant au développement de Taylor standard [18 ; 7]. Nous avons introduit une version rigide du λ-calcul à ressources qui se contente de fixer l'ordre des copies d'arguments dans les applications. Les termes de ce calcul sont les objets d'un groupoïde d'arbres de permutations : ces arbres agissent sur les termes en permutant les copies d'arguments, de sorte qu'un terme à ressources usuel n'est rien d'autre qu'une composante connexe de ce groupoïde. Ceci nous permet de revisiter les résultats combinatoires établis par Ehrhard et Regnier pour caractériser les coefficients du développement de Taylor : en particulier, le coefficient d'un terme est l'inverse du cardinal du groupe d'isotropie de chacun de ses représentants rigides. Par ailleurs, nous avons montré que la présence de marqueurs du choix restaurait la possibilité d'exploiter la propriété d'uniformité du λ-calcul pour établir la compatibilité du développement de Taylor avec la normalisation.

Ces travaux forment le Chapitre 4 du présent mémoire. Ils apportent un éclairage intéressant sur la version originelle du développement de Taylor des λ-termes, sans toutefois donner directement un modèle bicatégorique du λ-calcul pur dans les espèces généralisées. Dans sa thèse [26], Olimpieri a poursuivi une autre piste, avec succès : il a construit un objet réflexif dans la bicatégorie des espèces généralisées, ainsi qu'un système de types avec intersections dont les dérivations décrivent le modèle du λ-calcul pur construit sur cet objet réflexif. Il a ensuite défini un développement de Taylor rigide polyadique enrichi avec des morphismes, qui donne une syntaxe de termes pour les dérivations de typage précédentes : la réduction de ces termes décrit exactement les 2-morphismes associés à la β-réduction.

Cette contribution d'Olimpieri résonne particulièrement avec certains résultats obtenus par Zeinab Galal dans sa thèse [27]. Celle-ci développe une approche bicatégorique des techniques d'orthogonalité pour les modèles de la logique linéaire. Plus précisément, elle introduit et étudie des extensions par orthogonalité du modèle des profoncteurs, qui est le modèle bicatégorique de la logique linéaire sous-jacent au modèle des espèces généralisées pour le λ-calcul pur. Ces extensions comprennent une version bicatégorique des espaces de finitude, ainsi qu'une version fonctorielle des fonctions stables. Les résultats les plus prospectifs de la thèse de Galal concernent une relation d'orthogonalité entre espèces symétriques et espèces cartésiennes : il se trouve que la construction d'Olimpieri est elle-même paramétrée par une pseudo-monade qui induit un opérateur d'intersection sur les types, non-idempotent dans le cas symétrique et idempotent dans le cas cartésien. Ceci laisse entrevoir une possible connexion entre les deux systèmes de types, analogue dans ce cadre bicatégorique de l'écrasement extensionnel du modèle relationnel sur le modèle de Scott [Ehr12b ; Ehr12a].

1.3 Interlude subjectif : ce qui me meut

On a vu que, pour justifier l'étude de systèmes qui mettent en jeu des sommes, voire des combinaisons linéaires, de termes ou de preuves, on peut brandir l'argument du non-déterminisme :

- la somme représente un choix non-déterministe — et si la somme n'est pas idempotente, on garde une trace du nombre de choix pouvant mener à un certain résultat ;
- une distribution de probabilités discrètes à support fini n'est qu'un cas particulier de combinaison linéaire à coefficients réels positifs ;
- la superposition d'états quantiques est généralement vue comme un vecteur à coefficients complexes.

Ces motivations sont excellentes, et on les retrouve souvent dans les introductions d'articles du domaine, mais je ne serais pas honnête en prétendant que ces applications potentielles sont essentielles dans mon travail.

Mon propre intérêt pour ces questions provient initialement de la nécessité de les traiter proprement dans le cadre du λ-calcul différentiel et de la logique linéaire différentielle. Je trouve enthousiasmante la possibilité offerte par certains modèles, jouissant de structures mathématiques riches, d'appliquer à l'étude de la logique et du calcul les méthodes de la combinatoire, de l'algèbre et de l'analyse, plutôt que de se cantonner à des variations sur les notions d'ordre partiel ou de graphe. Explorer le langage sous-jacent à ces structures me semble une entreprise essentielle.

Au fil de mes travaux, j'ai également pu constater que la prise en compte de ces aspects dans une syntaxe, telle que le λ-calcul algébrique, qui conserve l'exigence d'une approche contextuelle, produit toutes sortes d'effets intéressants. Pris dans toute sa généralité, le λ-calcul algébrique est une sorte de continent sauvage, c'est-à-dire incohérent *a priori* : toute paire de termes admet un antécédent commun par β-réduction. On peut en domestiquer certains territoires par diverses contraintes appliquées *a posteriori* : stratégies de réduction, typage, réalisabilité, contraintes sur les coefficients, sur la forme des termes, *etc.*

Le cas de la normalisation est emblématique :

- dans le cas général tout terme est réductible et donc, dans le sens usuel, aucun terme n'est normalisable ;
- si on écarte les coefficients négatifs, les termes typés sont normalisables, et les formes normales obtenues étendent conservativement celles du λ-calcul ;
- on peut neutraliser temporairement les coefficients, normaliser si c'est possible, puis restaurer les coefficients dans le résultat : les travaux d'Alberti montrent que la forme normale obtenue ne dépend pas du représentant « neutralisé » qu'on a choisi ;
- si on se restreint aux combinaisons probabilistes, les arbres de Leventis donnent une solution sans typage ;

— si on ne restreint pas la forme des termes ni les coefficients, mais qu'on se place dans un cadre typé, l'existence de modèles comme les espaces de finitude fournit au moins une notion sémantique de forme normale ;

— et pour les termes Taylor-normalisables, c'est-à-dire ceux dont le développement de Taylor est finitaire au sens mentionné plus haut, la forme normale du développement de Taylor généralise la notion d'arbre de Böhm.

Ainsi, là où la pureté du λ-calcul usuel tend à unifier les notions, le caractère sauvage du λ-calcul algébrique tend à les hiérarchiser, de sorte que conserver de bonnes propriétés demande d'identifier les définitions et techniques les plus robustes. C'est cette exigence qui m'a permis de mettre au point une méthode pour simuler la réduction à travers le développement de Taylor, en toute généralité, puis d'étendre cette méthode aux réseaux de démonstration avec Chouquet.

1.4 La suite

Les travaux mentionnés plus hauts peuvent susciter nombre de prolongements. Je détaille ici trois pistes parmi celles qui m'occupent le plus aujourd'hui, chacune liée à un projet de thèse débutant ou en cours.

Développement de Taylor pour le λ-calcul infinitaire. Les premiers travaux de la thèse de Rémy Cerda, débutée en 2020, semblent indiquer que la simulation de la réduction à travers le développement de Taylor reste pertinente dans un cadre infinitaire : on peut approcher les termes du λ-calcul infinitaire [Ken+97] par les mêmes termes à ressources que pour le λ-calcul usuel, [6] et donc conserver des approximations finies ; alors on peut simuler non seulement la β-réduction en un pas mais aussi les suites convergentes de réductions à travers le développement de Taylor.

Dans ce cadre, nous travaillons à obtenir au moins les retombées suivantes :

— une preuve de la standardisation infinitaire, basée sur le développement de Taylor, et peut-être plus simple que la généralisation coinductive de l'approche classique [EP13] ;

— la caractérisation, à travers leur développement de Taylor, de termes dont la réduction est sûre, c'est-à-dire évitant certains phénomènes pathologiques du λ-calcul infinitaire (non confluence dans le cas général, nécessité d'introduire une réduction spécifique pour les termes non solvables, *etc.*) — par exemple par une propriété de finitude.

Élimination des coupures parallèles dans les réseaux de démonstration. Le point de vue apporté par l'élimination parallèle des coupures permet d'envisager une approche nouvelle des propriétés opérationnelles des réseaux de démonstration : confluence, factorisation, standardisation, normalisation par niveaux, *etc.* En effet, en étendant la notion d'élimination parallèle des coupures aux réseaux de la logique linéaire, on pourra par exemple adapter la technique de Tait et Martin-Löf pour la confluence.

6. Ce point ne vaut que pour le fragment Λ^{001} du λ-calcul infinitaire, qui restreint la coinduction aux positions d'arguments dans les applications. Il faut adapter le calcul à ressources dans les autres cas.

La difficulté principale est de formaliser cette élimination parallèle en présence des boîtes exponentielles. De premiers résultats ont été obtenus dans le cadre des réseaux $?$-canoniques, [7] durant le stage de M2 de Giulia Manara, que j'ai dirigé en collaboration avec Giulio Guerrieri. La poursuite de ce programme est l'un des axes du projet de thèse de Manara, qui débute sous la direction conjointe de Thomas Ehrhard et moi-même.

On sait en tout cas que les résultats de finitude déjà obtenus avec Chouquet constituent une brique essentielle pour produire une notion de normalisation par évaluation pour la logique linéaire [17].

Développement de Taylor et sémantique des jeux. Le rapprochement entre développement de Taylor et sémantique des jeux est dans l'air du temps. Il est par exemple explicitement mentionné par Tsukada, Asada et Ong, comme une intuition sous-jacente à leur travail sur les espèces généralisées [TAO17].

Avec Lison Blondeau-Patissier et Pierre Clairambault, nous cherchons à formaliser la relation entre l'interprétation d'un λ-terme comme une stratégie et la normalisation de son développement de Taylor. Notre approche consiste à raffiner cette relation au niveau des approximations finies :

- on peut développer une stratégie en un ensemble d'*augmentations* [BC21], qui en sont des approximations finies ;
- les augmentations satisfaisant certaines contraintes (automatiquement valides dans l'interprétation d'un λ-terme) correspondent exactement à des termes à ressources en forme normale ;
- modulo cette correspondance, la stratégie interprétant un terme est exactement la forme normale de son développement de Taylor.

C'est sur cette prémisse que s'appuie le projet de thèse de Blondeau-Patissier, qui débute sous la direction conjointe de Clairambault et moi-même. Outre l'établissement de cette correspondance, il s'agit d'en tirer les fils : par exemple en en déduisant une notion de développement de Taylor pour des langages typés, avec ordre supérieur et références, pour lesquels la sémantique des jeux est actuellement le principal moyen d'étude ; ou bien en révélant la correspondance entre les arbres de Böhm probabilistes de Leventis [24 ; Lev18], le développement de Taylor probabiliste de Dal Lago et Leventis [LL19] et les jeux innocents probabilistes de Clairambault et Paquet [CP18].

Une étape préliminaire est de proposer une notion de développement compatible avec les arbres de Nakajima, qui sont la version infiniment η-développée des arbres de Böhm. En effet, la sémantique des jeux est intrinsèquement extensionnelle, tandis que le développement de Taylor usuel ne valide pas la règle η. L'inspiration fournie par les augmentations nous a déjà permis de proposer un langage de termes à ressources qui sont des approximations finies de termes infiniment η-développés. Toute la théorie du développement de Taylor semble pouvoir s'adapter dans ce cadre : calcul des coefficients dans le cas uniforme, compatibilité avec la réduction, correspondance entre arbre de Nakajima et forme normale du développement, *etc.*

7. Il s'agit des réseaux de ce qu'on appelait la *nouvelle syntaxe*, quand elle était nouvelle.

Chapter 2

Normalizing the Taylor expansion of non-deterministic λ-terms, *via* parallel reduction of resource vectors

This chapter is essentially the inclusion of the article of the same name [5], published in Logical Methods in Computer Science *in 2019.*

Abstract: It has been known since Ehrhard and Regnier's seminal work on the Taylor expansion of λ-terms that this operation commutes with normalization: the expansion of a λ-term is always normalizable and its normal form is the expansion of the Böhm tree of the term.

We generalize this result to the non-uniform setting of the algebraic λ-calculus, *i.e.*, λ-calculus extended with linear combinations of terms. This requires us to tackle two difficulties: foremost is the fact that Ehrhard and Regnier's techniques rely heavily on the uniform, deterministic nature of the ordinary λ-calculus, and thus cannot be adapted; second is the absence of any satisfactory generic extension of the notion of Böhm tree in presence of quantitative non-determinism, which is reflected by the fact that the Taylor expansion of an algebraic λ-term is not always normalizable.

Our solution is to provide a fine grained study of the dynamics of β-reduction under Taylor expansion, by introducing a notion of reduction on resource vectors, *i.e.* infinite linear combinations of resource λ-terms. The latter form the multilinear fragment of the differential λ-calculus, and resource vectors are the target of the Taylor expansion of λ-terms. We show the reduction of resource vectors contains the image of any β-reduction step, from which we deduce that Taylor expansion and normalization commute on the nose.

We moreover identify a class of algebraic λ-terms, encompassing both normalizable algebraic λ-terms and arbitrary ordinary λ-terms: the expansion of these is always normalizable, which guides the definition of a generalization of Böhm trees to this setting.

2.1 Introduction

Quantitative semantics was first proposed by Girard [Gir88] as an alternative to domains and continuous functionals, for defining denotational models of λ-calculi with a natural interpretation of non-determinism: a type is given by a collection of "atomic states"; a term of type A is then represented by a vector (*i.e.* a possibly infinite formal linear combination) of states. The main matter is the treatment of the function space: the construction requires the interpretation of function terms to be analytic, *i.e.* defined by power series.

This interpretation of λ-terms was at the origin of linear logic: the application of an analytic map to its argument boils down to the linear application of its power series (seen as a matrix) to the vector of powers of the argument; similarly, linear logic decomposes the application of λ-calculus into the linear cut rule and the promotion operator. Indeed, the seminal model of linear logic, namely coherence spaces and stable/linear functions, was introduced as a qualitative version of quantitative semantics [Gir86, especially Appendix C].

Dealing with power series, quantitative semantics must account for infinite sums. The interpretations of terms in Girard's original model can be seen as a special case of Joyal's analytic functors [Joy86]: in particular, coefficients are sets and infinite sums are given by coproducts. This allows to give a semantics to fixed point operators and to the pure, untyped λ-calculus. On the other hand, it does not provide a natural way to deal with weighted (*e.g.*, probabilistic) non-determinism, where coefficients are taken in an external semiring of scalars.

In the early 2000's, Ehrhard introduced an alternative presentation of quantitative semantics [Ehr05], limited to a typed setting, but where types can be interpreted as particular vector spaces, or more generally semimodules over an arbitrary fixed semiring; called *finiteness spaces*, these are moreover equipped with a linear topology, allowing to interpret linear logic proofs as linear and continuous maps, in a standard sense. In this setting, the formal operation of differentiation of power series recovers its usual meaning of linear approximation of a function, and morphisms in the induced model of λ-calculus are subject to Taylor expansion: the application $\varphi(\alpha)$ of the analytic function φ to the vector α boils down to the sum $\sum_{n\in\mathbf{N}} \frac{1}{n!}\left(\frac{\partial^n\varphi}{\partial x^n}\right)_{x=0} \cdot \alpha^n$ where $\left(\frac{\partial^n\varphi}{\partial x^n}\right)_{x=0}$ is the n-th derivative of φ computed at 0, which is an n-linear map, and α^n is the n-th tensor power of α.

Ehrhard and Regnier gave a computational meaning to such derivatives by introducing linearized variants of application and substitution in the λ-calculus, which led to the differential λ-calculus [ER03], and then the resource λ-calculus [ER08] — the latter retains iterated derivatives at zero as the only form of application. They were then able to recast the above Taylor expansion formula in a syntactic, untyped setting: to every λ-term M, they associate a vector $\tau(M)$ of resource λ-terms, *i.e.* terms of the resource λ-calculus.

The Taylor expansion of a λ-term can be seen as an intermediate, infinite object, between the term and its denotation in quantitative semantics. Indeed, resource terms still retain a dynamics, if a very simple, finitary one: the size of terms is strictly decreasing under reduction. Furthermore, normal resource terms are in close relationship with the atomic states of quantitative semantics of the pure λ-calculus (or equivalently with the elements of a reflexive object in the relational model [BEM07]; or with normal type derivations in a non-idempotent intersection type system

[Car18a]), so that the normal form of $\tau(M)$ can be considered as the denotation of M, which allows for a very generic description of quantitative semantics.

Other approaches to quantitative semantics generally impose a constraint on the computational model *a priori*. For instance, the model of finiteness spaces [Ehr05] is, by design, limited to strongly normalizing computation. Another example is that of probabilistic coherence spaces [DE11], a model of untyped λ-calculi extended with probabilistic choice, rather than arbitrary weighted superpositions. Alternatively, one can interpret non-deterministic extensions of PCF [Lai+13; Lai16], provided the semiring of scalars has all infinite sums. By contrast, the "normalization of Taylor expansion" approach is more canonical, as it does not rely on a restriction on the scalars, nor on the terms to be interpreted.

Of course, there is a price attached to such canonicity: in general, the normal form of a vector of resource λ-terms is not well defined, because we may have to consider infinite sums of scalars. Ehrhard and Regnier were nonetheless able to prove that the Taylor expansion $\tau(M)$ of a pure λ-term is always normalizable [ER08]. This can be seen as a new proof of the fact that Girard's quantitative semantics of pure λ-terms uses finite cardinals only [Has96]. They moreover established that this normal form is exactly the Taylor expansion of the Böhm tree $\mathsf{BT}(M)$ of M [ER06a] ($\mathsf{BT}(M)$ is the possibly infinite tree obtained by hereditarily applying the head reduction strategy in M). Both results rely heavily on the uniformity property of the pure λ-calculus: all the resource terms in $\tau(M)$ follow a single syntactic tree pattern. This is a bit disappointing since quantitative semantics was introduced as a model of non-determinism, which is ruled out by uniformity.

Actually, the Taylor expansion operator extends naturally to the algebraic λ-calculus [2]: a generic, non-uniform extension of λ-calculus, augmenting the syntax with formal finite linear combinations of terms. Then it is not difficult to find terms whose Taylor expansion is not normalizable. Nonetheless, interpreting types as finiteness spaces of resource terms, Ehrhard [Ehr10] proved by a reducibility technique that the Taylor expansion of algebraic λ-terms typed in a variant of system F is always normalizable.

2.1.1 Main results

In the present paper, we generalize Ehrhard's result and show that all weakly normalizable algebraic λ-terms have a normalizable Taylor expansion (Theorem 2.8.21, p.67). [1]

We moreover relate the normal form of the expansion of a term with the normal form of the term itself, both in a computational sense (*i.e.* the irreducible form obtained after a sequence of reductions) and in a more denotational sense, via an analogue of the notion of Böhm tree: Taylor expansion does commute with normalization, in both those senses (Theorem 2.8.22, p.67; Theorem 2.9.14, p.74).

When restricted to pure λ-terms, Theorem 2.9.14 provides a new proof, not relying on uniformity, that the normal form of $\tau(M)$ is isomorphic to $\mathsf{BT}(M)$. In their full extent, our

1. We had already obtained such a result for strongly normalizable λ-terms in a previous work with Pagani and Tasson [12]: there, we further proved that the finiteness structure on resource λ-terms could be refined to characterize exactly the strong normalizability property in a λ-calculus with finite formal sums of terms. Here we rely on a much coarser notion of finiteness: see subsection 2.8.1.

results provide a generalization of the notion of non-deterministic Böhm tree [LP95] in a weighted, quantitative setting.

Let us stress that neither Ehrhard's work [Ehr10] nor our own previous work with Pagani and Tasson [12] addressed the commutation of normalization and Taylor expansion. Indeed, in the absence of uniformity, the techniques used by Ehrhard and Regnier [ER08; ER06a] are no longer available, and we had to design another approach. [2] Our solution is to introduce a notion of reduction on resource vectors, so that: (i) this reduction contains the translation of any β-reduction step (Lemma 2.7.6, p.60); (ii) normalizability (and the value of the normal form) of resource vectors is preserved under reduction (Lemma 2.8.3, p.63). This approach turns out to be quite delicate, and its development led us to two technical contributions that we deem important enough to be noted here:

- the notion of *reduction structure* (subsection 2.5.3) that allows to control the families of resource terms simultaneously involved in the reduction of a resource vector: in particular this provides a novel, modular mean to circumvent the inconsistency of β-reduction in presence of negative coefficients (a typical deficiency of the algebraic λ-calculus [2]);
- our analysis of the effect of parallel reduction on the size of resource λ-terms (Section 2.6): this constitutes the technical core of our approach, and it plays a crucial rôle in establishing key additional properties such as confluence (Lemma 2.6.17, p.52, and Corollary 2.6.29, p.58) and conservativity (Lemma 2.7.14, p.61, and Lemma 2.8.23, p.68).

2.1.2 Structure of the paper

The paper begins with a few mathematical preliminaries, in section 2.2: we recall some definitions about semirings and semimodules (Subsection 2.2.1), if only to fix notations and vocabulary; we also provide a very brief review of finiteness spaces (Subsection 2.2.2), then detail the particular case of linear-continuous maps defined by summable families of vectors (subsection 2.2.3), the latter notion pervading the paper.

In Section 2.3 we review the syntax and the reduction relation of the resource λ-calculus, as introduced by Ehrhard and Regnier [ER08]. The subject is quite standard now, and the only new material we provide is about minor and unsurprising combinatorial properties of multilinear substitution.

Section 2.4 contains our first notable contribution: after recalling the Taylor expansion construction, we prove that it is compatible with substitution. This result is related with the functoriality of promotion in quantitative denotational models and the proof technique is quite similar. In the passing, we recall the syntax of the algebraic λ-calculus and briefly discuss the issues raised by the contextual extension of β-reduction in presence of linear combinations of terms, as evidenced by previous work [10; AD08; 2, *etc.*].

2. It is in fact possible to refine Ehrhard and Regnier's approach, *via* the introduction of a *rigid* variant of Taylor expansion [TAO17], which can then be adapted to the non-deterministic setting. This allows to describe the coefficients in the normal form of Taylor expansion, like in the uniform case, and then prove that Taylor expansion commutes with the computation of Böhm trees. It does not solve the problem of possible divergence, though, and one has to assume the semiring of coefficients is complete, *i.e.* that all sums converge. See Subsection 2.1.3 on related work for more details.

In Section 2.5, we discuss the possible extensions of the reduction of the resource λ-calculus to resource vectors, *i.e.* infinite linear combinations of resource terms, and identify two main issues. First, in order to simulate β-reduction, we are led to consider the parallel reduction of resource terms in resource vectors, which is not always well defined. Indeed, a single resource term might have unboundedly many antecedents by parallel reduction, hence this process might generate infinite sums of coefficients: we refer to this phenomenon as the *size collapse* of parallel resource reduction (Subsection 2.5.2). Second, similarly to the case of the algebraic λ-calculus, the induced equational theory might become trivial, due the interplay between coefficients in vectors and the reduction relation. To address the latter problem we introduce the notion of reduction structure (Subsection 2.5.3) which allows us to modularly restrict the set of resource terms involved in a reduction: later in the paper, we will identify reduction structures ensuring the consistency of the reduction of resource vectors (Subsection 2.8.4).

In Section 2.6, we introduce successive restrictions of the parallel reduction of resource vectors, in order to avoid the abovementioned size collapse. We first observe that, to bound the size of a term as a function of the size of any of its reducts, it is sufficient to bound the length of chains of immediately nested fired redexes in a single parallel reduction step (Subsection 2.6.1). This condition does not allow us to close a pair of reductions to a common reduct, because it is not stable under unions of fired redexes. We thus tighten it to bounding the length of all chains of (not necessarily immediately) nested fired redexes (Subsection 2.6.2): this enables us to obtain a strong confluence result, under a mild hypothesis on the semiring. An even more demanding condition is to require the fired redexes as well as the substituted variables to occur at a bounded depth (subsection 2.6.3): then we can define a maximal parallel reduction step for each bound, which entails strong confluence without any additional hypothesis. Finally, we consider reduction structures involving resource terms of bounded height (Subsection 2.6.4): when restricting to such a bounded reduction structure, the strongest of the above three conditions is automatically verified.

We then show, in Section 2.7, that the translation of β-reduction through Taylor expansion fits into this setting: the height of the resource terms involved in a Taylor expansion is bounded by that of the original algebraic λ-term, and every β-reduction step is an instance of the previously introduced parallel reduction of resource vectors. As a consequence of our strongest confluence result, we moreover obtain that any reduction step from the Taylor expansion of a λ-term can be extended into the translation of a parallel β-reduction step.

We turn our attention to normalization in Section 2.8. We first show that normalizable resource vectors are stable under reduction. We moreover establish that their normal form is obtained as the limit of the parallel left reduction strategy (Subsection 2.8.1). Then we introduce *Taylor normalizable* algebraic λ-terms as those having a normalizable Taylor expansion, and deduce from the previous results that they are stable under β-reduction (Subsection 2.8.2): in particular, the normal form of Taylor expansion does define a denotational semantics for that class of terms. Then we establish that normalizable terms are Taylor normalizable (subsection 2.8.3): it follows that normalization and Taylor expansion commute on the nose.

We conclude with Section 2.9, showing how our techniques can be applied to the class of hereditarily determinable terms, that we introduce *ad-hoc*: those include pure λ-terms as well as normalizable algebraic λ-terms as a particular case, and we show that all hereditarily determinable terms are Taylor normalizable and the coefficients of the normal form are given by a sequence of approximants, close to the Böhm tree construction.

2.1.3 Related and future work

Besides the seminal work by Ehrhard and Regnier [ER08; ER06a] in the pure case, we have already cited previous approaches to the normalizability of Taylor expansion based on finiteness conditions [Ehr10; 12].

A natural question to ask is how our generic notion of normal form of Taylor expansion compares with previously introduced notions of denotation in non-deterministic settings: non-deterministic Böhm trees [LP95], probabilistic Böhm trees [24], weighted relational models [DE11; Lai+13; Lai16], *etc.* The very statement of such a question raises several difficulties, prompting further lines of research.

One first obstacle is the fact that, by contrast with the uniform case of the ordinary λ-calculus, the Taylor expansion operator is not injective on algebraic λ-terms (see Subsection 2.4.5), not even on the partial normal forms that we use to introduce the approximants in section 2.9. This is to be related with the quotient that the non-deterministic Böhm trees of de'Liguoro and Piperno [LP95] must undergo in order to capture observational equivalence. On the other hand, to our knowledge, finding sufficient conditions on the semiring of scalars ensuring that the Taylor expansion becomes injective is still an open question.

Also, we define normalizable vectors based on the notion of summability: a sum of vectors converges when it is componentwise finite *i.e.*, for each component, only finitely many vectors have a non-zero coefficient (see subsection 2.2.3). If more information is available on scalars, namely if the semiring of scalars is complete in some topological or order-theoretic sense, it becomes possible to normalize the Taylor expansion of all terms.

Indeed, Tsukada, Asada and Ong have recently established [TAO17] the commutation between computing Böhm trees and Taylor expansion with coefficients taken in the complete semiring of positive reals $[0, +\infty]$ where all sums converge. Let us precise that they do not consider weighted non-determinism, only formal binary sums of terms, and that the notion of Böhm tree they consider is a very syntactic one, similar to the partial normal forms we introduce in section 2.9. Their approach is based on a precise description of the relationship between the coefficients of resource terms in the expansion of a term and those in the expansion of its Böhm-tree, using a *rigid* Taylor expansion as an intermediate step: this avoids the ambiguity between the sums of coefficients generated by redundancies in the expansion and those representing non-deterministic superpositions.

Tsukada, Asada and Ong's work can thus be considered as a refinement of Ehrhard and Regnier's method, that they are moreover able to generalize to the non-deterministic case provided the semiring of scalars is complete. By contrast, our approach is focused on β-reduction and identifies a class of algebraic λ-terms for which the normalization of Taylor expansion converges independently from the topology on scalars. It seems only natural to investigate the connections between both approaches, in particular to tackle the case of weighted non-determinism in a complete semiring, as a first step towards the treatment of probabilistic or quantum superposition, as also suggested by the conclusion of their paper.

In the probabilistic setting, though, the Böhm tree construction [24] relies on both the topological properties of real numbers and the restriction to discrete probability subdistributions. Relying on this, Dal Lago and Leventis have recently shown [LL19] that the sum defining the normal form of Taylor expansion of an arbitrary probabilistic λ-term always converges with finite coefficients, and that this normal form is the Taylor expansion of its probabilistic Böhm tree, in the non-extensional sense [24, section 4.2.1]. To get a better understanding of the shape of Taylor expansions of probabilistic λ-terms and their stability under reduction, a possible first step is to investigate probabilistic coherence spaces [DE11] on resource λ-terms: these would be the analogue, in the probabilistic setting, of the finiteness structures ensuring the summability of normal forms in the non-deterministic setting (see Subsection 2.8.3).

Apart from relating our version of quantitative semantics with pre-existing notions of denotation for non-deterministic λ-calculi, we plan to investigate possible applications to other proof theoretic or computational frameworks: namely, linear logic proof nets [Gir87] and infinitary λ-calculus [Ken+97].

The Taylor expansion of λ-terms can be generalized to linear logic proof nets: the case of linear logic can even be considered as being more primitive, as it is directly related with the structure of those denotational models that validate the Taylor expansion formula [Ehr18]. Proof nets, however, do not enjoy the uniformity property of λ-terms: no general coherence relation is satisfied by the elements of the Taylor expansion of a proof net [Tas09, section V.4.1]. This can be related with the non-injectivity of coherence semantics [Tor03]. In particular, it is really unclear how Ehrhard and Regnier's methods, or even Tsukada, Asada and Ong's could be transposed to this setting. By contrast, our recent work with Chouquet [14] shows that our study of reduction under Taylor expansion can be adapted to proof nets.

It is also quite easy to extend the Taylor expansion operator to infinite λ-terms, at least for those of Λ^{001}, where only the argument position of applications is treated coinductively. For infinite λ-terms, it is no longer the case that the support of Taylor expansion involves resource λ-terms of bounded height only. Fortunately, we can still rely on the results of subsection 2.6.2, where we only require a bound on the nesting of fired redexes: this should allow us to give a counterpart, through Taylor expansion, of the strongly converging reduction sequences of infinite λ-terms. More speculatively, another possible outcome is a characterization of hereditarily head normalizable terms via their Taylor expansion, adapting our previous work on normalizability with Pagani and Tasson [12].

2.2 Technical preliminaries

We write:

- $\mathbf{N}$ for the semiring of natural numbers;
- $\mathfrak{P}(X)$ for the powerset of a set X: $\mathcal{X} \in \mathfrak{P}(X)$ iff $\mathcal{X} \subseteq X$;
- $\#X$ for the cardinal of a finite set X;
- $!X$ for the set of finite multisets of elements of X;
- $[x_1, \ldots, x_n] \in !X$ for the multiset with elements $x_1, \ldots, x_n \in X$ (taking repetitions into account), and then $\#[x_1, \ldots, x_n] = n$ for its cardinality;

- $\prod_{i\in I} X_i$ and $\sum_{i\in I} X_i$ respectively for the product and sum of a family $(X_i)_{i\in I}$ of sets: in particular $\sum_{i\in I} X_i = \bigcup_{i\in I} \{i\} \times X_i$;
- $X^I = \prod_{i\in I} X$ for the set of applications from I to X or, equivalently, for the set of I indexed families of elements of X.

Throughout the paper we will be led to consider various categories of sets and elements associated with a single base set X: elements of X, subsets of X, finite multisets of elements of X, *etc.* In order to help keeping track of those categories, we generally adopt the following typographic conventions:

- we use small latin letters for the elements of X, say $a, b, c \in X$;
- for subsets of X, we use cursive capitals, say $\mathcal{A}, \mathcal{B}, \mathcal{C} \in \mathfrak{P}(X)$;
- for sets of subsets of X, we use Fraktur capitals, say $\mathfrak{A}, \mathfrak{B}, \mathfrak{C} \subseteq \mathfrak{P}(X)$;
- for (possibly infinite) linear combinations of elements of X, we use small greek letters, say $\alpha, \beta, \gamma \in \mathbf{S}^X$, where $\mathbf{S}$ denotes some set of scalar coefficients;
- we transpose all of the above conventions to the set $!X$ of finite multisets by overlining: *e.g.*, we write $\overline{a} = [a_1, \dots, a_n] \in !X$, $\overline{\mathcal{A}} \subseteq !X$ or $\overline{\alpha} \in \mathbf{S}^{!X}$.

In the remaining of this section, we introduce basic mathematical content that will be used throughout the paper.

2.2.1 Semirings and semimodules

A *semiring*[3] $\mathbf{S}$ is the data of a carrier set $\underline{\mathbf{S}}$, together with commutative monoids $(\underline{\mathbf{S}}, +_{\mathbf{S}}, 0_{\mathbf{S}})$ and $(\underline{\mathbf{S}}, \cdot_{\mathbf{S}}, 1_{\mathbf{S}})$ such that the multiplicative structure distributes over the additive one, *i.e.* for all $a, b, c \in \underline{\mathbf{S}}$, $a \cdot_{\mathbf{S}} 0_{\mathbf{S}} = 0_{\mathbf{S}}$ and $a \cdot_{\mathbf{S}} (b +_{\mathbf{S}} c) = a \cdot_{\mathbf{S}} b +_{\mathbf{S}} a \cdot_{\mathbf{S}} c$.

We will in general abuse notation and identify $\mathbf{S}$ with its carrier set $\underline{\mathbf{S}}$. We will moreover omit the subscripts on symbols $+$, $\cdot$, 0 and 1, and denote multiplication by concatenation: $ab = a \cdot b$. We also use standard notations for finite sums and products in $\mathbf{S}$, *e.g.* $\sum_{i=1}^n a_i = a_1 + \cdots + a_n$. For any semiring $\mathbf{S}$, there is a unique semiring morphism (in the obvious sense) from $\mathbf{N}$ to $\mathbf{S}$: to $n \in \mathbf{N}$ we associate the sum $\sum_{i=1}^n 1 \in \mathbf{S}$ that we also write $n \in \mathbf{S}$, although this morphism is not necessarily injective. Consider for instance the semiring $\mathbf{B}$ of booleans, with $\underline{\mathbf{B}} = \{0, 1\}$, $+_{\mathbf{B}} = \max$ and $\cdot_{\mathbf{B}} = \times$.

We finish this subsection by recalling the definitions of semimodules and their morphisms. A *(left)* $\mathbf{S}$*-semimodule* $\mathcal{M}$ is the data of a commutative monoid $(\underline{\mathcal{M}}, 0_{\mathcal{M}}, +_{\mathcal{M}})$ together with an external product $._{\mathcal{M}} : \mathbf{S} \times \underline{\mathcal{M}} \to \underline{\mathcal{M}}$ subject to the following identities:

$$
\begin{aligned}
0._{\mathcal{M}} m &= 0_{\mathcal{M}} & 1._{\mathcal{M}} m &= m \\
(a+b)._{\mathcal{M}} m &= a._{\mathcal{M}} m +_{\mathcal{M}} b._{\mathcal{M}} m & a._{\mathcal{M}}(b._{\mathcal{M}} m) &= ab._{\mathcal{M}} m \\
a._{\mathcal{M}} 0_{\mathcal{M}} &= 0_{\mathcal{M}} & a._{\mathcal{M}}(m +_{\mathcal{M}} n) &= a._{\mathcal{M}} m +_{\mathcal{M}} a._{\mathcal{M}} n
\end{aligned}
$$

3. The terminology of semirings is much less well established than that of rings, and one can find various non equivalent definitions depending on the presence of units or on commutativity requirements. Following Golan's terminology [Gol13], our semirings are *commutative semirings*, which is required here because we consider multilinear applications between modules.

for all $a, b \in \mathbf{S}$ and $m, n \in \underline{\mathcal{M}}$.

Again, we will in general abuse notation and identify $\mathcal{M}$ with its carrier set $\underline{\mathcal{M}}$, and omit the subscripts on symbols $+$, . and 0.

Let $\mathcal{M}$ and $\mathcal{N}$ be $\mathbf{S}$-semimodules. We say $\phi : \mathcal{M} \to \mathcal{N}$ is *linear* if

$$\phi\left(\sum_{i=1}^{n} a_i.m_i\right) = \sum_{i=1}^{n} a_i.\phi(m_i)$$

for all $m_1, \ldots, m_n \in \mathcal{M}$ and all $a_1, \ldots, a_n \in \mathbf{S}$. If moreover $\mathcal{M}_1, \ldots, \mathcal{M}_n$ are $\mathbf{S}$-semimodules, we say $\psi : \mathcal{M}_1 \times \cdots \times \mathcal{M}_n \to \mathcal{N}$ is n*-linear* if it is linear in each component.

Given a set X, $\mathbf{S}^X$ is the semimodule of formal linear combinations of elements of X: a *vector* $\xi \in \mathbf{S}^X$ is nothing but an X-indexed family of scalars $(\xi_x)_{x\in X}$, that we may also denote by $\sum_{x\in X} \xi_x.x$. The *support* $|\xi|$ of a vector $\xi \in \mathbf{S}^X$ is the set of elements of X having a non-zero coefficient in ξ:

$$|\xi| := \{x \in X \; ; \; \xi_x \neq 0\}.$$

We write $\mathbf{S}[X]$ for the set of vectors with finite support:

$$\mathbf{S}[X] := \left\{\xi \in \mathbf{S}^X \; ; \; |\xi| \text{ is finite}\right\}.$$

In particular $\mathbf{S}[X]$ is the semimodule freely generated by X, and is a subsemimodule of $\mathbf{S}^X$.

2.2.2 Finiteness spaces

A finiteness space [Ehr05] is a subsemimodule of $\mathbf{S}^X$ obtained by imposing a restriction on the support of vectors, as follows.

If X is a set, we call *structure on* X any set $\mathfrak{S} \subseteq \mathfrak{P}(X)$, and then the dual structure is

$$\mathfrak{S}^\perp := \{\mathcal{X}' \subseteq X \; ; \; \text{for all } \mathcal{X} \in \mathfrak{S}, \mathcal{X} \cap \mathcal{X}' \text{ is finite}\}.$$

A *relational finiteness space* is a pair $(X, \mathfrak{F})$, where X is a set (the *web* of the finiteness space) and $\mathfrak{F} \subseteq \mathfrak{P}(X)$ is a structure on X such that $\mathfrak{F} = \mathfrak{F}^{\perp\perp}$: $\mathfrak{F}$ is then called a *finiteness structure*, and we say $\mathcal{X} \subseteq X$ is *finitary* in $(X, \mathfrak{F})$ iff $\mathcal{X} \in \mathfrak{F}$. The *finiteness space* generated by $(X, \mathfrak{F})$, denoted by $\mathbf{S}\langle X, \mathfrak{F}\rangle$, or simply $\mathbf{S}\langle\mathfrak{F}\rangle$, is then the set of vectors on X with finitary support: $\xi \in \mathbf{S}\langle\mathfrak{F}\rangle$ iff $|\xi| \in \mathfrak{F}$.

By this definition, if $\xi \in \mathbf{S}\langle\mathfrak{F}\rangle$ and $\xi' \in \mathbf{S}\langle\mathfrak{F}^\perp\rangle$ then the sum $\sum_{x\in X} \xi_x \xi'_x$ involves finitely many nonzero summands.

Finitary subsets are downwards closed for inclusion, and finite unions of finitary subsets are finitary, hence $\mathbf{S}\langle X, \mathfrak{F}\rangle$ is a subsemimodule of $\mathbf{S}^X$. Moreover, the least (resp. greatest) finiteness structure on X is the set $\mathfrak{P}_f(X)$ of finite subsets of X (resp. the powerset $\mathfrak{P}(X)$), generating the finiteness space $\mathbf{S}[X]$ (resp. $\mathbf{S}^X$).

We do not describe the whole category of finiteness spaces and linear-continuous maps here. In particular we do not recall the details of the linear topology induced on $\mathbf{S}\langle X, \mathfrak{F}\rangle$ by $\mathfrak{F}$: the reader may refer to Ehrhard's original paper [Ehr05] or his survey presentation of differential linear logic [Ehr18].

In the following, we focus on a very particular case, where the finiteness structure on base types is trivial (*i.e.* there is no restriction on the support of vectors): linear-continuous maps are then univocally generated by *summable functions.*

We started with the general notion of finiteness space nonetheless, because it provides a good background for the general spirit of our contributions: we are interested in infinite objects restricted so that, componentwise, all our constructions involve finite sums only. Also, the semimodule of normalizable resource vectors introduced in section 2.8 is easier to work with once its finiteness space structure is exposed.

2.2.3 Summable functions

Let $\overrightarrow{\xi} = (\xi_i)_{i\in I} \in \left(\mathbf{S}^X\right)^I$ be a family of vectors: write $\xi_i = \sum_{x\in X} \xi_{i,x}.x$. We say $\overrightarrow{\xi}$ is *summable* if, for all $x \in X$, $\{i \in I\ ;\ x \in |\xi_i|\}$ is finite. In this case, we define the sum $\sum \overrightarrow{\xi} = \sum_{i\in I} \xi_i \in \mathbf{S}^X$ in the obvious, pointwise way: [4]

$$\left(\sum \overrightarrow{\xi}\right)_x := \sum_{i\in I} \xi_{i,x}.$$

Of course, any finite family of vectors is summable and, fixing an index set I and a base set X, summable families in $\left(\mathbf{S}^X\right)^I$ form an $\mathbf{S}$-semimodule, with operations defined pointwise.

Moreover, if $(\xi_i)_{i\in I} \in \left(\mathbf{S}^X\right)^I$ is summable, then it follows from the inclusion $|a_i.\xi_i| \subseteq |\xi_i|$ that $(a_i.\xi_i)_{i\in I}$ is also summable for any family of scalars $(a_i)_{i\in I} \in \mathbf{S}^I$. Whenever the n-ary function $f : X_1 \times \cdots \times X_n \to \mathbf{S}^Y$ (*i.e.* the family $(f(x_1,\ldots,x_n))_{(x_1,\ldots,x_n)\in X_1\times\cdots\times X_n}$) is summable, we can thus define its *extension* $\langle f\rangle : \mathbf{S}^{X_1} \times \cdots \times \mathbf{S}^{X_n} \to \mathbf{S}^Y$ by

$$\langle f\rangle(\xi_1,\ldots,\xi_n) := \sum_{(x_1,\ldots,x_n)\in X_1\times\cdots\times X_n} \xi_{1,x_1}\cdots\xi_{n,x_n}.f(x_1,\ldots,x_n).$$

Note that we can consider $f : X \to \mathbf{S}^Y$ as a $Y \times X$ matrix: $f_{y,x} = f(x)_y$. Then if f is summable and $\xi \in \mathbf{S}^X$, $\langle f\rangle(\xi)$ is nothing but the application of the matrix f to the column ξ: the summability hypothesis ensures that this is well defined.

It turns out that the linear extensions of summable functions are exactly the linear-continuous maps, defined as follows:

Definition 2.2.1. Let $\varphi : \mathbf{S}^{X_1} \times \cdots \times \mathbf{S}^{X_n} \to \mathbf{S}^Y$. We say φ is n-*linear-continuous* if, for all summable families $\overrightarrow{\xi_1} = (\xi_{1,i})_{i\in I_1} \in \left(\mathbf{S}^{X_1}\right)^{I_1},\ldots,\overrightarrow{\xi_n} = (\xi_{n,i})_{i\in I_n} \in \left(\mathbf{S}^{X_n}\right)^{I_n}$, the

4. The reader can check that the family $\overrightarrow{\xi}$ is summable iff the support set

$$\{(i,x) \in I \times X\ ;\ \xi_{i,x} \neq 0\}$$

is finitary in the relational arrow finiteness space $(I \times X, \mathfrak{P}(I) \multimap \mathfrak{P}(X))$ as defined by Ehrhard [Ehr05, see in particular Lemma 3]. Then $\sum \overrightarrow{\xi}$ is the result of applying the matrix $(\xi_{i,x})_{i\in I,x\in X}$ to the vector $(1)_{i\in I} \in \mathbf{S}\langle\mathfrak{P}(I)\rangle = \mathbf{S}^I$.

family $(\varphi(\xi_{1,i_1},\ldots,\xi_{n,i_n}))_{(i_1,\ldots,i_n)\in I_1\times\cdots\times I_n}$ is summable and, for all families of scalars, $\overrightarrow{a_1} = (a_{1,i})_{i\in I_1} \in \mathbf{S}^{I_1},\ldots,\overrightarrow{a_n} = (a_{n,i})_{i\in I_n} \in \mathbf{S}^{I_n}$, we have

$$\varphi\left(\sum_{i_1\in I_1} a_{1,i_1}.\xi_{1,i_1},\ldots,\sum_{i_n\in I_n} a_{n,i_n}.\xi_{n,i_n}\right) = \sum_{(i_1,\ldots,i_n)\in I_1\times\cdots\times I_n} a_{1,i_1}\cdots a_{n,i_n}.\varphi(\xi_{1,i_1},\ldots,\xi_{n,i_n}).$$

Lemma 2.2.2. *If $\varphi : \mathbf{S}^{X_1}\times\cdots\times\mathbf{S}^{X_n} \to \mathbf{S}^Y$ is n-linear-continuous then its restriction $\varphi\restriction_{X_1\times\cdots\times X_n}$ is a summable n-ary function and $\varphi = \langle\varphi\restriction_{X_1\times\cdots\times X_n}\rangle$. Conversely, if $f : X_1\times\cdots\times X_n \to \mathbf{S}^Y$ is a summable n-ary function then $\langle f\rangle$ is n-linear-continuous.*

Proof. It is possible to derive both implications from general results on finiteness spaces. [5] We also sketch a direct proof.

The first implication follows directly from the definitions, observing that each diagonal family of vectors $(x)_{x\in X_i}$ is obviously summable.

For the converse: let $\overrightarrow{\xi_1} = (\xi_{1,i})_{i\in I_1} \in \left(\mathbf{S}^{X_1}\right)^{I_1},\ldots,\overrightarrow{\xi_n} = (\xi_{n,i})_{i\in I_n} \in \left(\mathbf{S}^{X_n}\right)^{I_n}$ be summable families. We first prove that the family

$$(\xi_{1,i_1,x_1}\cdots\xi_{n,i_n,x_n}.f(x_1,\ldots,x_n))_{(i_1,\ldots,i_n)\in I_1\times\cdots\times I_n,(x_1,\ldots,x_n)\in X_1\times\cdots\times X_n}$$

is summable. Fix $y\in Y$. If $y \in |\xi_{1,i_1,x_1}\cdots\xi_{n,i_n,x_n}.f(x_1,\ldots,x_n)|$ then in particular $y \in |f(x_1,\ldots,x_n)|$: since f is summable, there are finitely many such tuples $(x_1,\ldots,x_n)\in X_1\times\cdots\times X_n$. For each such tuple $(x_1,\ldots,x_n)$ and each $k\in\{1,\ldots,n\}$, since $\overrightarrow{\xi}_k$ is summable, there are finitely many i_k's such that $\xi_{k,i_k,x_k}\neq 0$. The necessary equation then follows from the associativity of sums. $\square$

From now on, we will identify summable functions with their multilinear-continuous extensions. Moreover, it should be clear that multilinear-continuous maps compose.

2.3 The resource λ-calculus

In this section, we recall the syntax and reduction of the resource λ-calculus, that was introduced by Ehrhard and Regnier [ER08] as the multilinear fragment of the differential λ-calculus [ER03]. The syntax is very similar to that of Boudol's resource λ-calculus [Bou93] but the intended meaning (multilinear approximations of λ-terms) as well as the dynamics is fundamentally different.

5. One might check that a map $\varphi : \mathbf{S}^{X_1}\times\cdots\times\mathbf{S}^{X_n}\to\mathbf{S}^Y$ is n-linear-continuous in the sense of Definition 2.2.1 iff it is n-linear and continuous in the sense of the linear topology of finiteness spaces, observing that the topology on $\mathbf{S}^X = \mathbf{S}\langle\mathfrak{P}(X)\rangle$ is the product topology ($\mathbf{S}$ being endowed with the discrete topology) [Ehr05, Section 3]. Moreover, n-ary summable functions $f : X_1\times\cdots\times X_n\to\mathbf{S}^Y$ are the elements of the finiteness space $\mathbf{S}\langle\mathfrak{P}(X_1)\otimes\cdots\otimes\mathfrak{P}(X_n)\multimap\mathfrak{P}(Y)\rangle$. As a general fact, the linear-continuous maps $\mathbf{S}\langle\mathfrak{F}\rangle\to\mathbf{S}\langle\mathfrak{G}\rangle$ are exactly the linear extensions of vectors in $\mathbf{S}\langle\mathfrak{F}\multimap\mathfrak{G}\rangle$. But linear-continuous maps from a tensor product of finiteness spaces correspond with multilinear-*hypocontinuous* maps [Ehr05, Section 3] rather than the more restrictive multilinear-continuous maps. In the very simple setting of summable functions, though, both notions coincide, since $\mathbf{S}^X$ is always locally linearly compact [Ehr05, Proposition 15].

We also recall the definitions of the multilinear counterparts of term substitution: partial differentiation and multilinear substitution.

In the passing, we introduce various quantities on resource λ-terms (size, height, and number and maximum depth of occurrences of a variable) and we state basic results that will be used throughout the paper.

Finally, we present the dynamics of the calculus: resource reduction and normalization.

2.3.1 Resource expressions

In the remaining of the paper, we suppose an infinite, countable set $\mathcal{V}$ of variables is fixed: we use small letters x, y, z to denote variables.

We define the sets Δ of *resource terms* and $!\Delta$ of *resource monomials* by mutual induction as follows: [6]

$$\begin{array}{rcccl} \Delta & \ni & s, t, u, v, w & ::= & x \mid \lambda x\, s \mid \langle s \rangle\, \bar{t} \\ !\Delta & \ni & \bar{s}, \bar{t}, \bar{u}, \bar{v}, \bar{w} & ::= & [] \mid [s] \cdot \bar{t}. \end{array}$$

Terms are considered up to α-equivalence and monomials up to permutativity: we write $[t_1, \ldots, t_n]$ for $[t_1] \cdot (\cdots ([t_n] \cdot []))$ and equate $[t_1, \ldots, t_n]$ with $\left[t_{f(1)}, \ldots, t_{f(n)}\right]$ for all permutation f of $\{1, \ldots, n\}$, so that resource monomials coincide with finite multisets of resource terms. [7] We will then write $\bar{s} \cdot \bar{t}$ for the multiset union of $\bar{s}$ and $\bar{t}$, and $\#[s_1, \ldots, s_n] := n$.

We call *resource expression* any resource term or resource monomial and write $(!)\Delta$ for either Δ or $!\Delta$: whenever we use this notation several times in the same context, all occurrences consistently denote the same set. When we make a definition or a proof by induction on resource expressions, we actually use a mutual induction on resource terms and monomials.

Definition 2.3.1. We define by induction over a resource expression $e \in (!)\Delta$, its *size* $\mathbf{s}(e) \in \mathbf{N}$ and its *height* $\mathbf{h}(e) \in \mathbf{N}$:

$$\begin{array}{rlrl} \mathbf{s}(x) & := 1 & \mathbf{h}(x) & := 1 \\ \mathbf{s}(\lambda x\, s) & := 1 + \mathbf{s}(s) & \mathbf{h}(\lambda x\, s) & := 1 + \mathbf{h}(s) \\ \mathbf{s}\big(\langle s \rangle\, \bar{t}\big) & := 1 + \mathbf{s}(s) + \mathbf{s}\big(\bar{t}\big) & \mathbf{h}\big(\langle s \rangle\, \bar{t}\big) & := \max\big\{\mathbf{h}(s), 1 + \mathbf{h}\big(\bar{t}\big)\big\} \\ \mathbf{s}([s_1, \ldots, s_n]) & := \displaystyle\sum_{i=1}^{n} \mathbf{s}(s_i) & \mathbf{h}([s_1, \ldots, s_n]) & := \max\{\mathbf{h}(s_i)\ ;\ 1 \leq i \leq n\}. \end{array}$$

It should be clear that, for all $e \in (!)\Delta$, $\mathbf{h}(e) \leq \mathbf{s}(e)$. Also observe that $\mathbf{s}(s) > 0$ and $\mathbf{h}(s) > 0$ for all $s \in \Delta$, and $\mathbf{s}(\bar{s}) \geq \#\bar{s}$ for all $\bar{s} \in !\Delta$. In the application case, we chose not to increment the height of the function: this is not crucial but it will allow to simplify some of

6. We use a self explanatory if not standard variant of BNF notation for introducing syntactic objects:

$$!\Delta \ni \bar{s}, \bar{t}, \bar{u}, \bar{v}, \bar{w} ::= [] \mid [s] \cdot \bar{t}$$

means that we define the set $!\Delta$ of resource monomials as that inductively generated by the empty monomial, and addition of a term to a monomial, and that we will denote resource monomials using overlined letters among $\bar{s}, \bar{t}, \bar{u}, \bar{v}, \bar{w}$, possibly with sub- and superscripts.

7. Resource monomials are often called *bags*, *bunches* or *poly-terms* in the literature, but we prefer to strengthen the analogy with power series here.

our computations in Section 2.6. In particular, in the case of a redex we have $\mathbf{h}\big(\langle \lambda x\, s\rangle\, \bar{t}\big) = 1 + \max\big\{\mathbf{h}(s), \mathbf{h}(\bar{t})\big\}$.

For all resource expression e, we write $\mathbf{fv}(e)$ for the set of its free variables. In the remaining of the paper, we will often have to prove that some set $\mathcal{E} \subseteq (!)\Delta$ is finite: we will generally use the fact that $\mathcal{E}$ is finite iff both $\{\mathbf{s}(e)\ ;\ e \in \mathcal{E}\}$ and $\mathbf{fv}(\mathcal{E}) := \bigcup_{e\in\mathcal{E}} \mathbf{fv}(e)$ are finite.

Besides the size and height of an expression, we will also need finer grained information on occurrences of variables, providing a quantitative counterpart to the set of free variables:

Definition 2.3.2. We define by induction over resource expressions the *number* $\mathbf{n}_x(e) \in \mathbf{N}$ *of occurrences* and the *set* $\mathbf{d}_x(e) \in \mathbf{N}$ *of occurrence depths* of a variable x in $e \in (!)\Delta$:

$$\begin{aligned}
\mathbf{n}_x(y) &:= \begin{cases} 1 & \text{if } x = y \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{n}_x(\lambda y\, s) &:= \mathbf{n}_x(s) && \text{(choosing } y \neq x\text{)} \\
\mathbf{n}_x\big(\langle s\rangle\, \bar{t}\big) &:= \mathbf{n}_x(s) + \mathbf{n}_x\big(\bar{t}\big) \\
\mathbf{n}_x([s_1,\dots,s_n]) &:= \sum_{i=1}^{n} \mathbf{n}_x(s_i)
\end{aligned}$$

and

$$\begin{aligned}
\mathbf{d}_x(y) &:= \begin{cases} \{1\} & \text{if } x = y \\ \emptyset & \text{otherwise} \end{cases} \\
\mathbf{d}_x(\lambda y\, s) &:= \{d+1\ ;\ d \in \mathbf{d}_x(s)\} && \text{(choosing } y \neq x\text{)} \\
\mathbf{d}_x([s_1,\dots,s_n]) &:= \textstyle\bigcup_{i=1}^{n} \mathbf{d}_x(s_i) \\
\mathbf{d}_x\big(\langle s\rangle\, \bar{t}\big) &:= \mathbf{d}_x(s) \cup \big\{d+1\ ;\ d \in \mathbf{d}_x\big(\bar{t}\big)\big\}.
\end{aligned}$$

We then write $\mathbf{md}_x(e) := \max \mathbf{d}_x(e)$ for the *maximal depth of occurrences* of x in e.

Again, it should be clear that $\mathbf{n}_x(e) \leq \mathbf{s}(e)$ and $\mathbf{md}_x(e) \leq \mathbf{h}(e)$. Moreover, $x \in \mathbf{fv}(e)$ iff $\mathbf{n}_x(e) \neq 0$ iff $\mathbf{d}_x(e) \neq \emptyset$ iff $\mathbf{md}_x(e) \neq 0$.

2.3.2 Partial derivatives

In the resource λ-calculus, the substitution $e[s/x]$ of a term s for a variable x in e admits a linear counterpart: this operator was initially introduced in the differential λ-calculus [ER03] in the form of a partial differentiation operation, reflecting the interpretation of λ-terms as analytic maps in quantitative semantics.

Partial differentiation enforces the introduction of formal finite sums of resource expressions: these are the actual objects of the resource λ-calculus, and in particular the dynamics will act on finite sums of terms rather than on simple resource terms (see Subsection 2.3.4). We extend all syntactic constructs to finite sums of resource expressions by linearity: if $\sigma = \sum_{i=1}^{n} s_i \in \mathbf{N}[\Delta]$ and $\bar{\tau} = \sum_{j=1}^{p} \bar{t}_j \in \mathbf{N}[!\Delta]$, we set $\lambda x\, \sigma := \sum_{i=1}^{n} \lambda x\, s_i$, $\langle \sigma\rangle\, \bar{\tau} := \sum_{i=1}^{n}\sum_{j=1}^{p} \langle s_i\rangle\, \bar{t}_j$ and $[\sigma]\cdot\bar{\tau} := \sum_{i=1}^{n}\sum_{j=1}^{p} [s_i]\cdot \bar{t}_j$.

This linearity of syntactic constructs will be generalized to vectors of resource expressions in the next section. For now, up to linearity, it is already possible to consider the substitution

$e[\sigma/x]$ of a finite sum of terms σ for a variable term x in an expression e: in particular $e[0/x] = 0$ whenever $x \in \mathbf{fv}(e)$. This is in turn extended to sums by linearity: $\epsilon[\sigma/x] = \sum_{i=1}^{n} e_i[\sigma/x]$ when $\epsilon = \sum_{i=1}^{n} e_i$. Observe that this is *not* linear in σ, because x may occur several times in e: for instance, with a monomial of degree 2, $[x,x][t+u/x] = [t,t] + [t,u] + [u,t] + [u,u]$.

Partial differentiation is then defined as follows:

Definition 2.3.3. For all $u \in \Delta$ and $x \in \mathcal{V}$, we define the *partial derivative* $\frac{\partial e}{\partial x} \cdot u \in \mathbf{N}[(!)\Delta]$ of $e \in (!)\Delta$, by induction on e:

$$\frac{\partial y}{\partial x} \cdot u := \begin{cases} u & \text{if } x = y \\ 0 & \text{otherwise} \end{cases}$$

$$\frac{\partial \lambda y\, s}{\partial x} \cdot u := \lambda y \left(\frac{\partial s}{\partial x} \cdot u \right) \qquad \text{(choosing } y \notin \{x\} \cup \mathbf{fv}(u)\text{)}$$

$$\frac{\partial \langle s \rangle\, \bar{t}}{\partial x} \cdot u := \left\langle \frac{\partial s}{\partial x} \cdot u \right\rangle \bar{t} + \langle s \rangle \left(\frac{\partial \bar{t}}{\partial x} \cdot u \right)$$

$$\frac{\partial [s_1, \ldots, s_n]}{\partial x} \cdot u := \sum_{i=1}^{n} \left[s_1, \ldots, \frac{\partial s_i}{\partial x} \cdot u, \ldots, s_n \right].$$

Partial differentiation is extended to finite sums of expressions by bilinearity: if $\epsilon = \sum_{i=1}^{n} e_i \in \mathbf{N}[(!)\Delta]$ and $\sigma = \sum_{j=1}^{p} s_j \in \mathbf{N}[\Delta]$, we set

$$\frac{\partial \epsilon}{\partial x} \cdot \sigma = \sum_{i=1}^{n} \sum_{j=1}^{p} \frac{\partial e_i}{\partial x} \cdot s_j.$$

Lemma 2.3.4 ([ER08, Lemma 2])**.** *If $x \notin \mathbf{fv}(u)$ then*

$$\frac{\partial}{\partial y} \left(\frac{\partial e}{\partial x} \cdot t \right) \cdot u = \frac{\partial}{\partial x} \left(\frac{\partial e}{\partial y} \cdot u \right) \cdot t + \frac{\partial e}{\partial x} \cdot \left(\frac{\partial t}{\partial y} \cdot u \right).$$

If moreover $y \notin \mathbf{fv}(t)$, we obtain a version of Schwarz's theorem on the symmetry of second derivatives:

$$\frac{\partial}{\partial y} \left(\frac{\partial e}{\partial x} \cdot t \right) \cdot u = \frac{\partial}{\partial x} \left(\frac{\partial e}{\partial y} \cdot u \right) \cdot t.$$

If $x \notin \mathbf{fv}(s_i)$ for all $i \in \{1, \ldots, n\}$, we write

$$\frac{\partial^n e}{\partial x^n} \cdot (s_1, \ldots, s_n) := \frac{\partial}{\partial x} \left(\cdots \frac{\partial e}{\partial x} \cdot s_1 \cdots \right) \cdot s_n.$$

More generally, we write

$$\frac{\partial^n e}{\partial x^n} \cdot (s_1, \ldots, s_n) := \left(\frac{\partial^n e[y/x]}{\partial y^n} \cdot (s_1, \ldots, s_n) \right) [x/y]$$

for any $y \notin \bigcup_{i=1}^{n} \mathbf{fv}(s_i) \cup (\mathbf{fv}(e) \setminus \{x\})$: it should be clear that this definition does not depend on the choice of such a variable y. By the previous lemma,

$$\frac{\partial^n e}{\partial x^n} \cdot (s_1, \ldots, s_n) = \frac{\partial^n e}{\partial x^n} \cdot \left(s_{f(1)}, \ldots, s_{f(n)} \right)$$

for any permutation f of $\{1,\dots,n\}$ and we will thus write

$$\frac{\partial^n e}{\partial x^n}\cdot\overline{s} := \frac{\partial^n e}{\partial x^n}\cdot(s_1,\dots,s_n)$$

whenever $\overline{s}=[s_1,\dots,s_n]$.

An alternative, more direct presentation of iterated partial derivatives is as follows. Suppose $\mathbf{n}_x(e)=m$, and write $x_1,\dots,x_m$ for the occurrences of x in e. Then:

$$\frac{\partial^n e}{\partial x^n}\cdot[s_1,\dots,s_n] = \sum_{\substack{f:\{1,\dots,n\}\to\{1,\dots,m\}\\ f\text{ injective}}} e\big[s_1,\dots,s_n/x_{f(1)},\dots,x_{f(n)}\big]$$

More formally, we obtain:

Lemma 2.3.5. *For all monomial $\overline{u}=[u_1,\dots,u_n]\in{!\Delta}$ and all variable $x\in\mathcal{V}$:*[8]

$$\frac{\partial^n y}{\partial x^n}\cdot\overline{u} = \begin{cases} y & \text{if } n=0\\ u_1 & \text{if } x=y \text{ and } n=1\\ 0 & \text{otherwise}\end{cases}$$

$$\frac{\partial^n \lambda y\, s}{\partial x^n}\cdot\overline{u} = \lambda y\left(\frac{\partial^n s}{\partial x^n}\cdot\overline{u}\right) \qquad \text{(choosing } y\notin\{x\}\cup\mathbf{fv}(\overline{u})\text{)}$$

$$\frac{\partial^n \langle s\rangle\,\overline{t}}{\partial x^n}\cdot\overline{u} = \sum_{(I,J)\text{ partition of }\{1,\dots,n\}} \left\langle \frac{\partial^{\#I} s}{\partial x^{\#I}}\cdot\overline{u}_I\right\rangle \frac{\partial^{\#J}\overline{t}}{\partial x^{\#J}}\cdot\overline{u}_J$$

$$\frac{\partial^n [s_1,\dots,s_k]}{\partial x^n}\cdot\overline{u} = \sum_{(I_1,\dots,I_k)\text{ partition of }\{1,\dots,n\}} \left[\frac{\partial^{\#I_1} s_1}{\partial x^{\#I_1}}\cdot\overline{u}_{I_1}\dots,\frac{\partial^{\#I_k} s_k}{\partial x^{\#I_k}}\cdot\overline{u}_{I_k}\right]$$

where $\overline{u}_I$ denotes $\left[u_{i_1},\dots,u_{i_p}\right]$ whenever $I=\{i_1,\dots,i_p\}$ with $p=\#I$.

Proof. Easy, by induction on n. □

Lemma 2.3.6. *For all $e\in(!)\Delta$, $\overline{s}\in{!\Delta}$, $x\neq y\in\mathcal{V}$ and $e'\in\left|\frac{\partial^n e}{\partial x^n}\cdot\overline{s}\right|$ with $n=\#\overline{s}$, moreover assuming that $x\notin\mathbf{fv}(\overline{s})$:*

- $\mathbf{n}_x(e)\geq n$ *and* $\mathbf{n}_x(e')=\mathbf{n}_x(e)-n$;
- $\mathbf{n}_y(e')=\mathbf{n}_y(e)+\mathbf{n}_y(\overline{s})$;
- $\mathbf{d}_x(e')\subseteq\mathbf{d}_x(e)$;
- $\mathbf{d}_y(e)\subseteq\mathbf{d}_y(e')\subseteq\mathbf{d}_y(e)\cup\{d+d'-1\,;\,d\in\mathbf{d}_x(e),d'\in\mathbf{d}_y(\overline{s})\}$;
- $\mathbf{s}(e')=\mathbf{s}(e)+\mathbf{s}(\overline{s})-n$;
- $\mathbf{h}(e)\leq\mathbf{h}(e')\leq\max\{\mathbf{h}(e),\mathbf{md}_x(e)+\mathbf{h}(\overline{s})-1\}$.

Proof. Each result is easily established by induction on e, using the previous lemma to enable the induction. □

8. In this definition and in the remaining of the paper, we say a tuple $(I_1,\dots,I_n)\in\mathfrak{P}(I)^n$ is a partition of I if $I=\bigcup_{i=1}^n I_k$, and the I_k's are pairwise disjoint. We do not require the I_k's to be nonempty. Hence a partition of I into a n-tuple is uniquely defined by a function from I to $\{1,\dots,n\}$.

2.3.3 Multilinear substitution

Recall that Taylor expansion involves iterated derivatives at 0. If $n = \#\overline{s}$ and $x \notin \mathbf{fv}(\overline{s})$ we write

$$\partial_x e \cdot \overline{s} := \left(\frac{\partial^n e}{\partial x^n} \cdot \overline{s}\right)[0/x].$$

Observe that by Lemma 2.3.6: if $n > \mathbf{n}_x(e)$ then $\frac{\partial^n e}{\partial x^n} \cdot \overline{s} = 0$; and if $n < \mathbf{n}_x(e)$ then $x \in \mathbf{fv}(e')$ for all $e' \in \left|\frac{\partial^n e}{\partial x^n} \cdot \overline{s}\right|$, and then $e'[0/x] = 0$. In other words,

$$\partial_x e \cdot \overline{s} = \begin{cases} \frac{\partial^n e}{\partial x^n} \cdot \overline{s} & \text{if } n = \mathbf{n}_x(e) \\ 0 & \text{otherwise} \end{cases} \quad .$$

We say $\partial_x e \cdot \overline{s}$ is the *n-linear substitution* of $\overline{s}$ for x in e. More generally, we write

$$\partial_x e \cdot \overline{s} := (\partial_y e[y/x] \cdot \overline{s})[x/y]$$

for any $y \notin \mathbf{fv}(\overline{s}) \cup (\mathbf{fv}(e) \setminus x)$ and it should again be clear that this definition does not depend of the choice of such a y. By a straightforward application of Lemma 2.3.6, we obtain:

Lemma 2.3.7. *For all $e \in (!)\Delta$, $\overline{s} \in !\Delta$, $x \neq y \in \mathcal{V}$ and $e' \in |\partial_x e \cdot \overline{s}|$, assuming $x \notin \mathbf{fv}(\overline{s})$:*

- $\mathbf{n}_x(e) = \#\overline{s}$ *and* $\mathbf{n}_x(e') = 0$;
- $\mathbf{n}_y(e') = \mathbf{n}_y(e) + \mathbf{n}_y(\overline{s})$;
- $\mathbf{d}_x(e') = \emptyset$;
- $\mathbf{d}_y(e) \subseteq \mathbf{d}_y(e') \subseteq \mathbf{d}_y(e) \cup \{d + d' - 1\ ;\ d \in \mathbf{d}_x(e), d' \in \mathbf{d}_y(\overline{s})\}$;
- $\mathbf{s}(e') = \mathbf{s}(e) + \mathbf{s}(\overline{s}) - \#\overline{s}$;
- $\mathbf{h}(e) \leq \mathbf{h}(e') \leq \max\{\mathbf{h}(e), \mathbf{md}_x(e) + \mathbf{h}(\overline{s}) - 1\}$.

In particular, $\mathbf{fv}(e') = (\mathbf{fv}(e) \setminus \{x\}) \cup \mathbf{fv}(\overline{s})$, and $\max\{\mathbf{s}(e), \mathbf{s}(\overline{s})\} \leq \mathbf{s}(e') \leq \mathbf{s}(e) + \mathbf{s}(\overline{s})$.

Again, we can give a direct presentation of multilinear substitution. Suppose $\mathbf{n}_x(e) = m$, and write $x_1, \ldots, x_m$ for the occurrences of x in e. Then:

$$\partial_x e \cdot [s_1, \ldots, s_n] = \sum_{\substack{f:\{1,\ldots,n\} \to \{1,\ldots,m\} \\ f \text{ bijective}}} e\left[s_1, \ldots, s_n / x_{f(1)}, \ldots, x_{f(n)}\right].$$

More formally, as a consequence of Lemma 2.3.5:

Lemma 2.3.8. *For all monomial $\overline{u} = [u_1, \ldots, u_n] \in !\Delta$ and all variable $x \in \mathcal{V}$:*

$$\partial_x y \cdot \overline{u} = \begin{cases} y & \text{if } y \neq x \text{ and } n = 0 \\ u_1 & \text{if } y = x \text{ and } n = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\partial_x \lambda y\, s \cdot \overline{u} = \lambda y\, (\partial_x s \cdot \overline{u}) \qquad \text{(choosing } y \notin \{x\} \cup \mathbf{fv}(\overline{u})\text{)}$$

$$\partial_x \langle s\rangle\, \overline{t} \cdot \overline{u} = \sum_{\substack{(I,J) \text{ partition of } \{1,\ldots,n\} \\ \text{s.t. } \#I = \mathbf{n}_x(s) \text{ and } \#J = \mathbf{n}_x(\overline{t})}} \langle \partial_x s \cdot \overline{u}_I\rangle\, \partial_x \overline{t} \cdot \overline{u}_J$$

$$\partial_x [s_1, \ldots, s_k] \cdot \overline{u} = \sum_{\substack{(I_1,\ldots,I_k) \text{ partition of } \{1,\ldots,n\} \\ \text{s.t. } \forall j,\ \#I_j = \mathbf{n}_x(s_j)}} [\partial_x s_1 \cdot \overline{u}_{I_1}, \ldots, \partial_x s_k \cdot \overline{u}_{I_k}]$$

where the conditions on cardinalities of subsets of $\{1, \ldots, n\}$ in the application and monomial cases may be omitted.

A similar result is the commutation of multilinear substitutions:

Lemma 2.3.9. *If $x \notin \mathbf{fv}(\overline{u})$ then:*

$$\partial_y \big(\partial_x e \cdot \overline{t}\big) \cdot \overline{u} = \sum_{\substack{(I,J) \text{ partition of } \{1,\ldots,\#\overline{u}\} \\ \text{s.t. } \#I = \mathbf{n}_x(e) \text{ and } \#J = \mathbf{n}_x(\overline{t})}} \partial_x(\partial_y e \cdot \overline{u}_I) \cdot \big(\partial_y \overline{t} \cdot \overline{u}_J\big).$$

Proof. Write $n = \#\overline{t}$ and $p = \#\overline{u}$. It is sufficient to prove

$$\frac{\partial^p}{\partial y^p}\left(\frac{\partial^n e}{\partial x^n} \cdot \overline{t}\right) \cdot \overline{u} = \sum_{(I,J) \text{ partition of } \{1,\ldots,p\}} \frac{\partial^n}{\partial x^n}\left(\frac{\partial^{\#I} s}{\partial y^{\#I}} \cdot \overline{u}_I\right) \cdot \left(\frac{\partial^{\#J} \overline{t}}{\partial y^{\#J}} \cdot \overline{u}_J\right)$$

by induction on n and p, using Lemma 2.3.4. □

2.3.4 Resource reduction

If $\to$ is a reduction relation, we will write $\to^?$ (resp. $\to^+$; $\to^*$) for its reflexive (resp. transitive; reflexive and transitive) closure.

In the resource λ-calculus, a redex is a term of the form $\langle \lambda x\, t\rangle\, \overline{u} \in \Delta$ and its reduct is $\partial_x t \cdot \overline{u} \in \mathbf{N}[\Delta]$. The resource reduction $\to_\partial$ is then the contextual closure of this reduction step on finite sums of resource expressions. More precisely:

Definition 2.3.10. We define the *resource reduction* relation $\to_\partial \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$ inductively as follows:

- $\langle \lambda x\, s\rangle\, \overline{t} \to_\partial \partial_x s \cdot \overline{t}$ for all $s \in \Delta$ and $\overline{t} \in !\Delta$;
- $\lambda x\, s \to_\partial \lambda x\, \sigma'$ as soon as $s \to_\partial \sigma'$;
- $\langle s\rangle\, \overline{t} \to_\partial \langle \sigma'\rangle\, \overline{t}$ as soon as $s \to_\partial \sigma'$;

– $\langle s\rangle\,\bar{t} \to_\partial \langle s\rangle\,\overline{\tau}'$ as soon as $\bar{t} \to_\partial \overline{\tau}'$;

– $[s]\cdot\bar{t} \to_\partial [\sigma']\cdot\bar{t}$ as soon as $s \to_\partial \sigma'$.

We extend this reduction to finite sums of resource expressions: write $\epsilon \to_\partial \epsilon'$ if $\epsilon = \sum_{i=0}^n e_i$ and $\epsilon' = \sum_{i=0}^n \epsilon'_i$ with $e_0 \to_\partial \epsilon'_0$ and, for all $i \in \{1,\dots,n\}$, $e_i \to_\partial^? \epsilon'_i$.

Observe that we allow for parallel reduction of any nonzero number of summands in a finite sum. This reduction is particularly well behaved. In particular, it is confluent in a strong sense:

Lemma 2.3.11. *For all $\epsilon,\epsilon_0,\epsilon_1 \in \mathbf{N}[(!)\Delta]$, if $\epsilon \to_\partial \epsilon_0$ and $\epsilon \to_\partial \epsilon_1$ then there is $\epsilon' \in \mathbf{N}[(!)\Delta]$ such that $\epsilon_0 \to_\partial^? \epsilon'$ and $\epsilon_1 \to_\partial^? \epsilon'$.*

Proof. The proof follows a well-trodden path for proving confluence.

One first proves by induction on s that if $s \to_\partial \sigma'$ then $\partial_x s\cdot\bar{t} \to_\partial^? \partial_x\sigma'\cdot\bar{t}$, and if $\bar{t} \to_\partial \overline{\tau}'$ then $\partial_x s\cdot\bar{t} \to_\partial^? \partial_x s\cdot\overline{\tau}'$. Note that the reflexive closure is made necessary by the possibility that $\partial_x s\cdot\bar{t} = 0$, and the transitive closure is not needed because there is no duplication of the redexes of $\bar{t}$ in the summands of the multilinear substitution $\partial_x s\cdot\bar{t}$.

One then proves that if $e \to_\partial \epsilon_0$ and $e \to_\partial \epsilon_1$ then there is $\epsilon' \in \mathbf{N}[(!)\Delta]$ such that $\epsilon_0 \to_\partial^? \epsilon'$ and $\epsilon_1 \to_\partial^? \epsilon'$. The proof is straightforward, by induction on the pair of reductions $e \to_\partial \epsilon_0$ and $e \to_\partial \epsilon_1$, using the previous result in case e is a redex which is reduced in ϵ_0 but not in ϵ_1 (or *vice versa*). □

In other words, $\to_\partial^?$ enjoys the diamond property. [9] Moreover, the effect of reduction on the size of terms is very regular. First introduce some useful notation: write $e \succ_\partial e'$ if $e \to_\partial \epsilon'$ with $e' \in |\epsilon'|$.

Lemma 2.3.12. *Let $e \succ_\partial e'$. Then $\mathbf{fv}(e') = \mathbf{fv}(e)$, and $\mathbf{s}(e') + 2 \leq \mathbf{s}(e) \leq 2\mathbf{s}(e') + 2$.*

Proof. By induction on the reduction $e \to_\partial \epsilon'$ with $e' \in |\epsilon'|$. The inductive contextuality cases are easy, and we only detail the base case, *i.e.* $e = \langle \lambda x\, t\rangle\,\overline{u}$ and $\epsilon' = \partial_x t\cdot\overline{u}$.

Write $n = \mathbf{n}_x(t)$. The result then follows from Lemma 2.3.7, observing that $\mathbf{s}(e') = \mathbf{s}(t) + \mathbf{s}(\overline{u}) - n = \mathbf{s}(e) - 2 - n$ and $n \leq \mathbf{s}(\overline{u}) \leq \mathbf{s}(e')$. □

We will write $\geq_\partial$ (resp. $>_\partial$) for $\succ_\partial^*$ (resp. $\succ_\partial^+$). Observe that $e \geq_\partial e'$ (resp. $e >_\partial e'$) iff there is $\epsilon' \in \mathbf{N}[(!)\Delta]$ such that $e' \in |\epsilon'|$ and $e \to_\partial^* \epsilon'$ (resp. $e \to_\partial^+ \epsilon'$). Moreover, $\{e' \;;\; e \geq_\partial e'\}$ is always finite and $>_\partial$ defines a well-founded strict partial order. A direct consequence is that $\to_\partial$ always converges to a unique normal form:

Lemma 2.3.13. *The reduction $\to_\partial$ is confluent and strongly normalizing. Moreover, for all $\epsilon \in \mathbf{N}[(!)\Delta]$, the set $\{\epsilon' \;;\; \epsilon \to_\partial^* \epsilon'\}$ is finite.*

Proof. Confluence is a consequence of Lemma 2.3.11. By Lemma 2.3.12, the transitive closure $\succ_\partial^+$ is a well-founded strict partial order. Observe that the elements of $\mathbf{N}[(!)\Delta]$ can be considered as finite multisets of resource expressions: then $\to_\partial^+$ is included in the multiset ordering induced

9. This strong confluence result was not mentioned in Ehrhard and Regnier's papers about resource λ-calculus [ER08; ER06a] but they established a very similar result for differential nets [ER06b, Section 4]: Lemma 2.3.11 can be understood as a reformulation of the latter in the setting of resource calculus.

by $\succ_\partial^+$, and it follows that $\to_\partial^+$ defines a well-founded strict partial order on $\mathbf{N}[(!)\Delta]$, *i.e.* $\to_\partial$ is strongly normalizing.

The final property follows from strong normalizability applying König's lemma to the tree of possible reductions, observing that each ϵ has finitely many $\to_\partial$-reducts. □

If $\epsilon \in \mathbf{N}[(!)\Delta]$, we then write $\mathsf{NF}(\epsilon)$ for the unique sum of normal resource expressions such that $\epsilon \to_\partial^* \mathsf{NF}(\epsilon)$. A consequence of the previous lemma is that any reduction discipline reaches this normal form:

Corollary 2.3.14. *Let* $\to \subseteq \mathbf{N}[(!)\Delta] \times \mathbf{N}[(!)\Delta]$ *be such that* $\to \subseteq \to_\partial^*$. *Moreover assume that, for all non normal* $\epsilon \in \mathbf{N}[(!)\Delta]$ *there is* $\epsilon' \neq \epsilon$ *such that* $\epsilon \to \epsilon'$. *Then* $\epsilon \to^* \mathsf{NF}(\epsilon)$ *for all* $\epsilon \in \mathbf{N}[(!)\Delta]$.

2.4 Vectors of resource expressions and Taylor expansion of algebraic λ-terms

2.4.1 Resource vectors

A vector $\sigma = \sum_{s\in\Delta} \sigma_s.s$ of resource terms will be called a *term vector* whenever its set of free variables $\mathbf{fv}(\sigma) := \bigcup_{s\in|\sigma|} \mathbf{fv}(s)$ is finite. Similarly, we will call *monomial vector* any vector of resource monomials whose set of free variables is finite. We will abuse notation and write $\mathbf{S}^\Delta$ for the set of term vectors and $\mathbf{S}^{!\Delta}$ for the set of monomial vectors. [10]

A *resource vector* will be any of a term vector or a monomial vector, and we will write $\mathbf{S}^{(!)\Delta}$ for either $\mathbf{S}^\Delta$ or $\mathbf{S}^{!\Delta}$: as for resource expressions, whenever we use this notation several times in the same context, all occurrences consistently denote the same set.

The syntactic constructs are extended to resource vectors by linearity: for all $\sigma \in \mathbf{S}^\Delta$ and $\overline{\sigma}, \overline{\tau} \in \mathbf{S}^{!\Delta}$, we set

$$\begin{aligned} \lambda x\, \sigma \quad &:= \quad \sum_{s\in\Delta} \sigma_s.\lambda x\, s, \\ \langle\sigma\rangle\, \overline{\tau} \quad &:= \quad \sum_{s\in\Delta, \overline{t}\in!\Delta} \sigma_s \overline{\tau}_{\overline{t}}.\langle s\rangle\, \overline{t}, \\ \text{and } [\sigma_1,\ldots,\sigma_n] \quad &:= \quad \sum_{s_1,\ldots,s_n\in\Delta} (\sigma_1)_{s_1}\cdots(\sigma_n)_{s_n}.[s_1,\ldots,s_n]. \end{aligned}$$

This poses no problem for finite vectors: *e.g.*, if $|\sigma|$ is finite then finitely many of the vectors $\sigma_s.\lambda x\, s$ are non-zero, hence the sum is finite. In the general case, however, we actually need to

10. The restriction to vectors with finitely many free variables is purely technical. For instance, it allows us to assume that a sum of abstractions $\sigma = \sum_{i\in I} \lambda x_i\, s_i$ can always use a common abstracted variable: $\sigma = \sum_{i\in I} \lambda x\, (s_i[x/x_i])$, with $x \notin \cup_{i\in I}\mathbf{fv}(\lambda x_i\, s_i)$. Working without this restriction would only lead to more contorted statements and tedious bookkeeping: consider, *e.g.*, what would happen to the definition of the substitution of a term vector for a variable (Definition 2.4.4), especially the abstraction case.

prove that the above sums are well defined: the constructors of the calculus define summable functions, which thus extend to multilinear-continuous maps. [11]

Lemma 2.4.1. *The following families of vectors are summable:*

$$(\lambda x\, s)_{s\in\Delta}, \quad \left(\langle s\rangle\, \overline{t}\right)_{s\in\Delta, \overline{t}\in !\Delta}, \quad ([s])_{s\in\Delta} \quad \textit{and} \quad \left(\overline{s}\cdot\overline{t}\right)_{\overline{s},\overline{t}\in !\Delta}.$$

Proof. The proof is direct, but we detail it if only to make the requirements explicit.

For all $u \in \Delta$ there is at most one s such that $u \in |\lambda x\, s|$ (in which case $u = \lambda x\, s$) and at most one pair $(s, \overline{t})$ such that $u \in \left|\langle s\rangle\, \overline{t}\right|$ (in which case $u = \langle s\rangle\, \overline{t}$).

For all $\overline{u} \in !\Delta$ there is at most one s such that $\overline{u} \in |[s]|$ (in which case $\overline{u} = [s]$), and there are finitely many $\overline{s}$ and $\overline{t}$ such that $\overline{u} \in \left|\overline{s}\cdot\overline{t}\right|$ (those such that $\overline{u} = \overline{s}\cdot\overline{t}$). □

For each term vector σ, we then write σ^n for the monomial vector

$$[\overbrace{\sigma,\ldots,\sigma}^{n \text{ times}}].$$

2.4.2 Partial differentiation of resource vectors.

We can extend partial derivatives to vectors by linear-continuity (recall that, via the unique semiring morphism from $\mathbf{N}$ to $\mathbf{S}$, we can consider that $\mathbf{N}[(!)\Delta] \subseteq \mathbf{S}^{(!)\Delta}$).

Lemma 2.4.2. *The function*

$$\begin{array}{rcl} (!)\Delta \times !\Delta & \to & \mathbf{S}^{(!)\Delta} \\ (e, [s_1,\ldots,s_n]) & \mapsto & \dfrac{\partial^n e}{\partial x^n}\cdot [s_1,\ldots,s_n] \end{array}$$

is summable.

Proof. Let $e' \in (!)\Delta$ and assume that $e' \in \left|\frac{\partial^n e}{\partial x^n}\cdot\overline{s}\right|$ with $\#\overline{s} = n$. By Lemma 2.3.6, $\mathbf{fv}(e) \subseteq \mathbf{fv}(e') \cup \{x\}$, $\mathbf{fv}(\overline{s}) \subseteq \mathbf{fv}(e')$, $\mathbf{s}(e) \le \mathbf{s}(e')$ and $\mathbf{s}(\overline{s}) \le \mathbf{s}(e')$: e' being fixed, there are finitely many $(e, \overline{s})$ satisfying these constraints. □

The characterization of iterated partial derivatives given in Lemma 2.3.5 extends directly to resource vectors, by the linear-continuity of syntactic constructs and partial derivatives. For instance, given term vectors $\sigma, \rho_1, \ldots, \rho_n \in \mathbf{S}^{\Delta}$ and a monomial vector $\overline{\tau} \in \mathbf{S}^{!\Delta}$, we obtain:

$$\frac{\partial^n \langle\sigma\rangle\, \overline{\tau}}{\partial x^n}\cdot[\rho_1,\ldots,\rho_n] = \sum_{(I,J)\text{ partition of }\{1,\ldots,n\}} \left\langle \frac{\partial^{\#I}\sigma}{\partial x^{\#I}}\cdot\overline{\rho}_I \right\rangle \frac{\partial^{\#J}\overline{\tau}}{\partial x^{\#J}}\cdot\overline{\rho}_J.$$

Now we can consider iterated differentiation along a fixed term vector ρ: $\frac{\partial^n \epsilon}{\partial x^n}\cdot\rho^n$. We obtain:

11. The one-to-one correspondence between summable n-ary functions and multilinear-continuous maps was established for semimodules of the form $\mathbf{S}^X$, *i.e.* the semimodules of all vectors on a fixed set. Due to the restriction we put on free variables, $\mathbf{S}^{(!)\Delta}$ is not of this form: it should rather be written $\bigcup_{V\in\mathfrak{P}_f(\mathcal{V})}\mathbf{S}^{(!)\Delta_V}$ where $(!)\Delta_V := \{e \in (!)\Delta\ ;\ \mathbf{fv}(e) \subseteq V\}$. So when we say a function is multilinear-continuous on $\mathbf{S}^{(!)\Delta}$, we actually mean that its restriction to each $\mathbf{S}^{(!)\Delta_V}$ with $V \in \mathfrak{P}_f(\mathcal{V})$ is multilinear-continuous. In the present case, keeping this precision implicit is quite innocuous, but we will be more careful when considering the restriction to bounded vectors in Subsection 2.6.4, and to normalizable vectors in Section 2.8.

Lemma 2.4.3. *For all $\sigma, \tau_1, \dots, \tau_n, \rho \in \mathbf{S}^{\Delta}$ and all $\overline{\tau} \in \mathbf{S}^{!\Delta}$,*

$$\begin{aligned} \frac{\partial^k \langle\sigma\rangle\, \overline{\tau}}{\partial x^k} \cdot \rho^k &= \sum_{l=0}^{k} \begin{bmatrix} k \\ l, k-l \end{bmatrix} \left\langle \frac{\partial^l \sigma}{\partial x^l} \cdot \rho^l \right\rangle \frac{\partial^{k-l}\overline{\tau}}{\partial x^{k-l}} \cdot \rho^{k-l} \qquad \textit{and} \\ \frac{\partial^k [\tau_1, \dots, \tau_n]}{\partial x^k} \cdot \rho^k &= \sum_{\substack{k_1,\dots,k_n \in \mathbf{N} \\ k_1+\dots+k_n=k}} \begin{bmatrix} k \\ k_1, \dots, k_n \end{bmatrix} \left[\frac{\partial^{k_1}\tau_1}{\partial x^{k_1}} \cdot \rho^{k_1}, \dots, \frac{\partial^{k_n}\tau_n}{\partial x^{k_n}} \cdot \rho^{k_n} \right]. \end{aligned}$$

Proof. First recall that, if $k = \sum_{i=1}^n k_i$, the *multinomial coefficient* $\begin{bmatrix} k \\ k_1,\dots,k_n \end{bmatrix} := \frac{k!}{\prod_{i=1}^n k_i!}$ is nothing but the number of partitions of $\{1, \dots, k\}$ into n sets $I_1, \dots, I_n$ such that $\# I_j = k_j$ for $1 \leq j \leq n$ [DLMF, §26.4]. Then both results derive directly from Lemma 2.3.5. $\square$

2.4.3 Substitutions

Since $|\partial_x e \cdot \overline{s}| \subseteq \left|\frac{\partial^n e}{\partial x^n} \cdot \overline{s}\right|$, multilinear substitution also defines a summable binary function and we will write

$$\partial_x \epsilon \cdot \overline{\sigma} := \sum_{e \in (!)\Delta, \overline{s} \in !\Delta} \epsilon_e \overline{\sigma}_{\overline{s}} . \partial_x e \cdot \overline{s}.$$

By contrast with partial derivatives, the usual substitution is not linear, so the substitution of resource vectors must be defined directly.

Definition 2.4.4. We define by induction over resource expressions the *substitution* $e[\sigma/x] \in \mathbf{S}^{(!)\Delta}$ of $\sigma \in \mathbf{S}^{\Delta}$ for a variable x in $e \in (!)\Delta$:

$$\begin{aligned} x[\sigma/x] &:= \begin{cases} \sigma & \text{if } x = y \\ y & \text{otherwise} \end{cases} \\ (\lambda y\, s)[\sigma/x] &:= \lambda y\, s[\sigma/x] \qquad\qquad (\text{choosing } y \notin \mathbf{fv}(\sigma) \cup \{x\}) \\ [s_1, \dots, s_n][\sigma/x] &:= [s_1[\sigma/x], \dots, s_n[\sigma/x]] \\ \big(\langle s\rangle\, \overline{t}\big)[\sigma/x] &:= \langle s[\sigma/x]\rangle\, \overline{t}[\sigma/x] \end{aligned}$$

Lemma 2.4.5. *For all $e \in (!)\Delta$, $x \in \mathcal{V}$ and $\sigma \in \mathbf{S}^{\Delta}$:*

- *if $\sigma \in \Delta$ then $e[\sigma/x] \in \Delta$;*
- *if $\sigma \in \mathbf{S}[\Delta]$ then $e[\sigma/x] \in \mathbf{S}[(!)\Delta]$;*
- *if $x \notin \mathbf{fv}(e)$ then $e[\sigma/x] = e$;*
- *if $x \in \mathbf{fv}(e)$ then $e[0/x] = 0$;*
- *for all $e' \in |e[\sigma/x]|$, $\mathbf{fv}(e) \setminus \{x\} \subseteq \mathbf{fv}(e') \subseteq (\mathbf{fv}(e) \setminus \{x\}) \cup \mathbf{fv}(\sigma)$ and $\mathbf{s}(e') \geq \mathbf{s}(e)$.*

Proof. Each statement follows easily by induction on e. $\square$

A consequence of the last item is that the function

$$\begin{aligned} (!)\Delta &\to \mathbf{S}^{(!)\Delta} \\ e &\mapsto e[\sigma/x] \end{aligned}$$

is summable: we thus write

$$\epsilon[\sigma/x] := \sum_{e \in \mathbf{S}^{(!)\Delta}} \epsilon_e . e[\sigma/x].$$

2.4.4 Promotion

Observe that the family $(\sigma^n)_{n\in\mathbf{N}}$ is summable because the supports $|\sigma^n|$ for $n \in \mathbf{N}$ are pairwise disjoint. We then define the *promotion* of σ as $\sigma^! := \sum_{n\in\mathbf{N}} \frac{1}{n!}.\sigma^n$.

For this definition to make sense, we need inverses of natural numbers to be available: we say $\mathbf{S}$ *has fractions* if every $n \in \mathbf{N} \setminus \{0\}$ admits a multiplicative inverse in $\mathbf{S}$. This inverse is necessarily unique and we write it $\frac{1}{n}$. Observe that $\mathbf{S}$ has fractions iff there is a semiring morphism from the semiring $\mathbf{Q}^+$ of non-negative rational numbers to $\mathbf{S}$, and then this morphism is unique, but not necessarily injective: consider the semiring $\mathbf{B}$ of booleans. *Semifields*, *i.e.* commutative semirings in which every non-zero element admits an inverse, obviously have fractions: $\mathbf{Q}^+$ and $\mathbf{B}$ are actually semifields. In the following, we will keep this requirement implicit: whenever we use quotients by natural numbers, it means we assume $\mathbf{S}$ has fractions.

Lemma 2.4.6. *For all σ and $\tau \in \mathbf{S}^\Delta$, $\sigma^![\tau/x] = (\sigma[\tau/x])^!$.*

Proof. By the linear-continuity of $\epsilon \mapsto \epsilon[\sigma/x]$, it is sufficient to prove that

$$\sigma^n[\tau/x] = (\sigma[\tau/x])^n$$

which follows from the n-linear-continuity of $(\sigma_1, \ldots, \sigma_n) \mapsto [\sigma_1, \ldots, \sigma_n]$ and the definition of substitution. □

Lemma 2.4.7. *The following identities hold:*

$$\begin{aligned}
\partial_x x \cdot \rho^! &= \rho \\
\partial_x y \cdot \rho^! &= y \\
\partial_x \lambda y\, \sigma \cdot \rho^! &= \lambda y \left(\partial_x \sigma \cdot \rho^!\right) && \textit{(choosing } y \notin \{x\} \cup \mathbf{fv}(\rho)\textit{)} \\
\partial_x \langle\sigma\rangle\, \overline{\tau} \cdot \rho^! &= \left\langle \partial_x \sigma \cdot \rho^! \right\rangle \partial_x \overline{\tau} \cdot \rho^! \\
\partial_x [\sigma_1, \ldots, \sigma_n] \cdot \rho^! &= \left[\partial_x \sigma_1 \cdot \rho^!, \ldots, \partial_x \sigma_n \cdot \rho^!\right]
\end{aligned}$$

Proof. Since each syntactic constructor is multilinear-continuous, it is sufficient to consider the case of $\partial_x e \cdot \rho^!$ for a resource expression $e \in (!)\Delta$. First observe that, if $k = \mathbf{n}_x(e)$ then $\partial_x e \cdot \rho^! = \frac{1}{k!} . \frac{\partial^k e}{\partial x^k} \cdot \rho^k$. In particular the case of variables is straightforward.

The case of abstractions follows directly, since $\frac{\partial^k \lambda x\, s}{\partial x^k} \cdot \rho^k = \lambda x \left(\frac{\partial^k s}{\partial x^k} \cdot \rho^k\right)$.

If $e = \langle s\rangle\, \overline{t}$, write $l = \mathbf{n}_x(s)$ and $m = \mathbf{n}_x(\overline{t})$. It follows from Lemma 2.4.3 that $\partial_x e \cdot \rho^k = \left[\begin{smallmatrix} k \\ l,m \end{smallmatrix}\right] . \left\langle \partial_x s \cdot \rho^l \right\rangle \partial_x \overline{t} \cdot \rho^m$ and then $\frac{1}{k!} . \partial_x e \cdot \rho^k = \left\langle \frac{1}{l!} . \partial_x s \cdot \rho^l \right\rangle \frac{1}{m!} . \partial_x \overline{t} \cdot \rho^m$.

Similarly, if $e = [t_1, \ldots, t_n]$, write $k_i = \mathbf{n}_x(t_i)$ for all $i \in \{1, \ldots, n\}$. It follows from Lemma 2.4.3 that $\partial_x e \cdot \rho^k = \left[\begin{smallmatrix} k \\ k_1, \ldots, k_n \end{smallmatrix}\right] . \left[\partial_x t_1 \cdot \rho^{k_1}, \ldots, \partial_x t_n \cdot \rho^{k_n}\right]$ and then $\frac{1}{k!} . \partial_x e \cdot \rho^k = \left[\frac{1}{k_1!} \partial_x t_1 \cdot \rho^{k_1}, \ldots, \frac{1}{k_n!} \partial_x t_n \cdot \rho^{k_n}\right]$. □

Lemma 2.4.8. *For all* $\epsilon \in \mathbf{S}^{(!)\Delta}$ *an* $\sigma \in \mathbf{S}^{\Delta}$,

$$\epsilon[\sigma/x] = \partial_x \epsilon \cdot \sigma^!.$$

Proof. By the linear-continuity of $\epsilon \mapsto \partial_x \epsilon \cdot \sigma^!$ and $\epsilon \mapsto \epsilon[\sigma/x]$, it is sufficient to show that

$$e[\sigma/x] = \partial_x e \cdot \sigma^!$$

for all resource expression e. The proof is then by induction on e, using the previous Lemma in each case. □

By Lemma 2.4.6, we thus obtain

$$\partial_x \sigma^! \cdot \tau^! = \left(\partial_x \sigma \cdot \tau^!\right)^!$$

which can be seen as a counterpart of the functoriality of promotion in linear logic. To our knowledge it is the first published proof of such a result for resource vectors. This will enable us to prove the commutation of Taylor expansion and substitution (Lemma 2.4.10), another unsurprising yet non-trivial result.

2.4.5 Taylor expansion of algebraic λ-terms

Since resource vectors form a module, there is no reason to restrict the source language of Taylor expansion to the pure λ-calculus: we can consider formal finite linear combinations of λ-terms.

We will thus consider the terms given by the following grammar:

$$\Sigma_{\mathbf{S}} \quad \ni \quad M, N, P \quad ::= \quad x \mid \lambda x\, M \mid (M)\, N \mid 0 \mid a.M \mid M + N$$

where a ranges in $\mathbf{S}$. [12] For now, terms are considered up to the usual α-equivalence only: the null term 0, scalar multiplication $a.M$ and sum of terms $M + N$ are purely syntactic constructs.

Definition 2.4.9. We define the *Taylor expansion* $\tau(M) \in \mathbf{S}^{(!)\Delta}$ of a term $M \in \Sigma_{\mathbf{S}}$ inductively as follows:

$$\begin{aligned} \tau(x) &:= x & \tau(0) &:= 0 \\ \tau(\lambda x\, M) &:= \lambda x\, \tau(M) & \tau(a.M) &:= a.\tau(M) \\ \tau((M)\, N) &:= \langle \tau(M) \rangle\, \tau(N)^! & \tau(M+N) &:= \tau(M) + \tau(N). \end{aligned}$$

Lemma 2.4.10. *For all* $M, N \in \Sigma_{\mathbf{S}}$*, and all variable* x*,*

$$\tau(M[N/x]) = \partial_x \tau(M) \cdot \tau(N)^! = \tau(M)[\tau(N)/x].$$

12. We follow Krivine's convention [Kri90], by writing $(M)\, N$ for the application of term M to term N. We more generally write $(M)\, N_1 \cdots N_k$ for $(\cdots (M)\, N_1 \cdots)\, N_k$. Moreover, among term constructors, we give sums the lowest priority so that $(M)\, N + P$ should be read as $((M)\, N) + P$ rather than $(M)\, (N + P)$.

Proof. By induction on M, using Lemmas 2.4.7 and 2.4.8. $\square$

Let us insist on the fact that, despite its very simple and unsurprising statement, the previous lemma relies on the entire technical development of the previous subsections. Again, to our knowledge, it is the first proof that Taylor expansion commutes with substitution, in an untyped and non-uniform setting, without any additional assumption.

By contrast, one can forget everything about the semiring of coefficients and consider only the support of Taylor expansion. Recall that $\mathbf{B}$ denotes the semiring of booleans. Then we can consider that $\mathbf{B}^{(!)\Delta} = \mathfrak{P}((!)\Delta)$ and write, *e.g.*, $\lambda x\, \mathcal{S} = \{\lambda x\, s\ ;\ s \in \mathcal{S}\}$ for all set $\mathcal{S}$ of resource terms.

Definition 2.4.11. The *Taylor support* $\mathcal{T}(M) \subseteq \Delta$ of $M \in \Sigma_{\mathbf{S}}$ is defined inductively as follows: [13]

$$\begin{aligned} \mathcal{T}(x) &:= \{x\} & \mathcal{T}(0) &:= \emptyset \\ \mathcal{T}(\lambda x\, M) &:= \lambda x\, \mathcal{T}(M) & \mathcal{T}(a.M) &:= \mathcal{T}(M) \\ \mathcal{T}((M)\, N) &:= \langle \mathcal{T}(M)\rangle\, \mathcal{T}(N)^! & \mathcal{T}(M+N) &:= \mathcal{T}(M) \cup \mathcal{T}(N). \end{aligned}$$

It should be clear that $|\tau(M)| \subseteq \mathcal{T}(M)$, but the inclusion might be strict, if only because $\mathcal{T}(0.M) = \mathcal{T}(M)$. By contrast with the technicality of the previous subsection, the following *qualitative* analogue of Lemma 2.4.10 is easily established:

Lemma 2.4.12. *For all $M, N \in \Sigma_{\mathbf{S}}$, and all variable x,*

$$\mathcal{T}(M[N/x]) = \partial_x \mathcal{T}(M) \cdot \mathcal{T}(N)^! = \mathcal{T}(M)[\mathcal{T}(N)/x].$$

Proof. The qualitative version of Lemma 2.4.7 is straightforward. The result follows by induction on M. $\square$

The restriction of $\mathcal{T}$ to the set Λ of pure λ-terms was used by Ehrhard and Regnier [ER08] in their study of Taylor expansion. They showed that if $M \in \Lambda$ then $\mathcal{T}(M)$ is uniform: all the resource terms in $\mathcal{T}(M)$ have the same outermost syntactic construct and this property is preserved inductively on subterms. They moreover proved that $\tau(M)$, and in fact M itself, is entirely characterized by $\mathcal{T}(M)$: in this case, $\tau(M) = \sum_{s \in \mathcal{T}(M)} \frac{1}{m(s)} s$ where $m(s)$ is an integer coefficient depending only on s. Of course this property fails in the non uniform setting of $\Sigma_{\mathbf{S}}$.

Now, let us consider the equivalence induced on terms by Taylor expansion: write $M \simeq_\tau N$ if $\tau(M) = \tau(N)$.

13. One might be tempted to make an exception in case $a = 0$ and set $\mathcal{T}(0.M) = \emptyset$ but this would only complicate the definition and further developments for little benefit: what about $\mathcal{T}(a.M + b.M)$ (resp. $\mathcal{T}(a.b.M)$) in a semiring where $a \neq 0$, $b \neq 0$ and $a + b = 0$ (resp. $ab = 0$)? If we try and cope with those too, we are led to make $\mathcal{T}$ invariant under the equations of $\mathbf{S}$-module, which is precisely what we want to avoid here: see the case of τ in the remaining of the present section.

Lemma 2.4.13. *The following equations hold:*

$$\begin{array}{rcc} 0 + M \simeq_\tau M & M + N \simeq_\tau N + M & (M + N) + P \simeq_\tau M + (N + P) \\ 0.M \simeq_\tau 0 & 1.M \simeq_\tau M & a.M + b.M \simeq_\tau (a + b).M \\ a.0 \simeq_\tau 0 & a.(b.M) \simeq_\tau (ab).M & a.(M + N) \simeq_\tau a.M + a.N \\ \lambda x\, 0 \simeq_\tau 0 & \lambda x\, (a.M) \simeq_\tau a.\lambda x\, M & \lambda x\, (M + N) \simeq_\tau \lambda x\, M + \lambda x\, N \\ (0)\, P \simeq_\tau 0 & (a.M)\, P \simeq_\tau a.(M)\, P & (M + N)\, P \simeq_\tau (M)\, P + (N)\, P \end{array}$$

Moreover, $\simeq_\tau$ is compatible with syntactic constructs: if $M \simeq_\tau M'$ then $\lambda x\, M \simeq_\tau \lambda x\, M'$, $(M)\, N \simeq_\tau (M')\, N$, $(N)\, M \simeq_\tau (N)\, M'$, $a.M \simeq_\tau a.M'$, $M + N \simeq_\tau M' + N$ and $N + M \simeq_\tau N + M'$.

Proof. Up to Taylor expansion, these equations reflect the fact that $\mathbf{S}^{(!)\Delta}$ forms a semimodule (first three lines), and that all the constructions used in the definition of τ are multilinear-continuous, except for promotion (last two lines). Compatibility follows from the inductive definition of τ. □

Let us write $\simeq_{\mathrm{v}}$ for the least compatible equivalence relation containing the equations of the previous lemma, and call *vector λ-terms* the elements of the quotient $\Sigma_{\mathbf{S}}/\simeq_{\mathrm{v}}$: these are the terms of the previously studied algebraic λ-calculus [2; 23]. [14]

It is clear that $\Sigma_{\mathbf{S}}/\simeq_{\mathrm{v}}$ forms a $\mathbf{S}$-semimodule. In fact, one can show [2] that $\Sigma_{\mathbf{S}}/\simeq_{\mathrm{v}}$ is freely generated by the $\simeq_{\mathrm{v}}$-equivalence classes of base terms, *i.e.* those described by the following grammar:

$$\Sigma^{\mathrm{b}}_{\mathbf{S}} \quad \ni \quad B \quad ::= \quad x \mid \lambda x\, B \mid (B)\, M.$$

Hence we could write $\Sigma_{\mathbf{S}}/\simeq_{\mathrm{v}} = \mathbf{S}\left[\Sigma^{\mathrm{b}}_{\mathbf{S}}/\simeq_{\mathrm{v}}\right]$.

Notice however that Taylor expansion is not injective on vector λ-terms in general.

Example 2.4.14. *We can consider that $\Sigma_{\mathbf{B}}/\simeq_{\mathrm{v}} = \mathfrak{P}_f\left(\Sigma^{\mathrm{b}}_{\mathbf{B}}/\simeq_{\mathrm{v}}\right)$ and $\tau(M) \subseteq \Delta$ for all $M \in \Sigma_{\mathbf{B}}$. It is then easy to check that,* e.g., *$\tau((x)\, \emptyset) \subseteq \tau((x)\, x)$, hence $(x)\, \emptyset +_{\mathbf{B}} (x)\, x \simeq_\tau (x)\, x$.* [15]

This contrasts with the case of pure λ-terms, for which τ is always injective: in this case, it is in fact sufficient to look at the linear resource terms in supports of Taylor expansions.

Fact 2.4.15. For all $M, N \in \Lambda$, $\ell(M) \in |\tau(N)|$ iff $M = N$, where ℓ is defined inductively as follows:

$$\ell(x) := x \qquad \ell(\lambda x\, M) := \lambda x\, \ell(M) \qquad \ell((M)\, N) := \langle \ell(M) \rangle\, [\ell(N)].$$

14. In those previous works, the elements of $\Sigma_{\mathbf{S}}/\simeq_{\mathrm{v}}$ were called algebraic λ-terms, but here we reserve this name for another, simpler, notion.

15. This discrepancy is also present in the non-deterministic Böhm trees of de'Liguoro and Piperno [LP95]: in that qualitative setting, they can solve it by introducing a preorder on trees based on set inclusion. They moreover show that this preorder coincides with that induced by a well chosen domain theoretic model, as well as with the observational preorder associated with must-solvability. This preorder should be related with that induced by the inclusion of normal forms of Taylor expansions (which are always defined since we then work with support sets rather than general vectors).

To our knowledge, finding sufficient conditions on $\mathbf{S}$ ensuring that τ becomes injective on $\Sigma_{\mathbf{S}}/{\simeq_{\mathrm{v}}}$ is still an open question.

Observe moreover that the $\mathbf{S}$-semimodule structure of $\Sigma_{\mathbf{S}}/{\simeq_{\mathrm{v}}}$ gets in the way when we want to study β-reduction and normalization: it is well known [10; AD08; 2] that β-reduction in a semimodule of terms is inconsistent in presence of negative coefficients.

Example 2.4.16. *Consider $\delta_M := \lambda x\,(M + (x)\,x)$ and $\infty_M := (\delta_M)\,\delta_M$. Observe that ∞_M β-reduces to $M + \infty_M$. Suppose $\mathbf{S}$ is a ring. Then any congruence $\simeq$ on $\Sigma_{\mathbf{S}}$ containing β-reduction and the equations of $\mathbf{S}$-module is inconsistent: $0 \simeq \infty_M + (-1).\infty_M \simeq (M + \infty_M) + (-1).\infty_M \simeq M$.*

The problem is of course the identity $0 \simeq \infty_M + (-1).\infty_M$. Another difficulty is that, if $\mathbf{S}$ has fractions then, up to $\mathbf{S}$-semimodule equations, one can split a single β-reduction step into infinitely many fractional steps: if $M \to_\beta M'$ then

$$M \simeq \tfrac{1}{2}.M + \tfrac{1}{2}.M \to_\beta \tfrac{1}{2}.M + \tfrac{1}{2}.M' \simeq \left(\tfrac{1}{4}.M + \tfrac{1}{4}.M\right) + \tfrac{1}{2}.M' \to_\beta \left(\tfrac{1}{4}.M + \tfrac{1}{4}.M'\right) + \tfrac{1}{2}.M' \simeq \cdots$$

It is not our purpose here to explore the various possible fixes to the rewriting theory of β-reduction on vector λ-terms. We rather refer the reader to the literature on algebraic λ-calculi [2; AD08; 23; Día11] for various proposals. Our focus being on Taylor expansion, we propose to consider vector λ-terms as intermediate objects: the reduction relation induced on resource vectors by β-reduction through Taylor expansion contains β-reduction on vector terms — which is mainly useful to understand what may go wrong.

We still need to introduce some form of quotient in the syntax, though, if only to allow formal sums to retain a computational meaning: otherwise, for instance, no β-redex can be fired in $(\lambda x\, M + \lambda x\, N)\, P$; and more generally there are β-normal terms whose Taylor expansion is not normal, and conversely (consider, *e.g.*, $(\lambda x\, 0)\, P$).

Write $\Lambda_{\mathbf{S}}$ for the quotient of $\Sigma_{\mathbf{S}}$ by the least compatible equivalence $\simeq_+$ containing the following six equations:

$$\begin{array}{ccc} \lambda x\, 0 \simeq_+ 0 & \lambda x\,(a.M) \simeq_+ a.\lambda x\, M & \lambda x\,(M+N) \simeq_+ \lambda x\, M + \lambda x\, N \\ (0)\, P \simeq_+ 0 & (a.M)\, P \simeq_+ a.(M)\, P & (M+N)\, P \simeq_+ (M)\, P + (N)\, P \end{array}$$

We call *algebraic λ-terms* the elements of $\Lambda_{\mathbf{S}}$. We will abuse notation and denote an algebraic λ-term by any of its representatives.

Observe that $\mathcal{T}(M)$ is preserved under $\simeq_+$ so it is well defined on algebraic terms, although not on vector terms.

Fact 2.4.17. An algebraic λ-term M is β-normal (*i.e.* each of its representatives is β-normal) iff $\mathcal{T}(M)$ contains only normal resource terms.

We do not claim that $\simeq_+$ is minimal with the above property (for this, the bottom three equations are sufficient) but it is quite natural for anyone familiar with the decomposition of λ-calculus in linear logic, as it reflects the linearity of λ-abstraction and the function position in an application. Moreover it retains the two-level structure of vector λ-terms, seen as sums of base terms.

It is indeed a routine exercise to show that orienting the defining equations of $\simeq_+$ from left to right defines a confluent and terminating rewriting system. We call canonical terms the normal forms of this system, which we can describe as follows. The sets $\Sigma^{\mathrm{c}}_{\mathbf{S}}$ of canonical terms and $\Sigma^{\mathrm{s}}_{\mathbf{S}}$ of simple canonical terms are mutually generated by the following grammars:

$$\begin{array}{lcll} \Sigma^{\mathrm{s}}_{\mathbf{S}} & \ni & S, T & ::= & x \mid \lambda x\, S \mid (S)\, M \\ \Sigma^{\mathrm{c}}_{\mathbf{S}} & \ni & M, N, P & ::= & S \mid 0 \mid a.M \mid M + N \end{array}$$

so that each algebraic term M admits a unique canonical $\simeq_+$-representative.

In the remaining of this paper we will systematically identify algebraic terms with their canonical representatives and keep $\simeq_+$ implicit. Moreover, we write $\Lambda^{\mathrm{s}}_{\mathbf{S}}$ for the set of simple algebraic λ-terms, *i.e.* those that admit a simple canonical representative.

Fact 2.4.18. Every simple term $S \in \Lambda^{\mathrm{s}}_{\mathbf{S}}$ is of one of the following two forms:

- $S = \lambda x_1 \cdots \lambda x_n\, (x)\, M_1 \cdots M_k$: S is a head normal form;
- $S = \lambda x_1 \cdots \lambda x_n\, (\lambda x\, T)\, M_0 \cdots M_k$: $(\lambda x\, T)\, M_0$ is the head redex of S.

So each algebraic λ-term can be considered as a formal linear combination of head normal forms and head reducible simple terms, which will structure the notions of weak solvability and hereditarily determinable terms in section 2.9.

2.5 On the reduction of resource vectors

Observe that

$$\tau((\lambda x\, M)\, N) = \langle \lambda x\, \tau(M) \rangle\, \tau(N)^! = \sum_{\substack{s \in \Delta \\ \bar{t} \in !\Delta}} \tau(M)_s \tau(N)^!_{\bar{t}}.\langle \lambda x\, s \rangle\, \bar{t}$$

and

$$\tau(M[N/x]) = \partial_x \tau(M) \cdot \tau(N)^! = \sum_{\substack{s \in \Delta \\ \bar{t} \in !\Delta}} \tau(M)_s \tau(N)^!_{\bar{t}}.\partial_x s \cdot \bar{t}$$

In order to simulate β-reduction through Taylor expansion we might be tempted to consider the reduction given by $\epsilon \to \epsilon'$ as soon as $\epsilon = \sum_{i \in I} a_i.e_i$ and $\epsilon' = \sum_{i \in I} a_i.\epsilon'_i$ with $e_i \to_\partial \epsilon'_i$ for all $i \in I$.[16]

Observe indeed that, as soon as $(a_i.e_i)_{i \in I} \in (!)\Delta$ is summable (*i.e.* for all $e \in (!)\Delta$, there are finitely many $i \in I$ such that $a_i \neq 0$ and $e_i = e$), the family $(a_i.\epsilon'_i)_{i \in I}$ is summable too: if $e' \in |a_i.\epsilon'_i|$ then $a_i \neq 0$ and $e' \in |\epsilon'_i|$ hence by Lemma 2.3.12, $\mathbf{fv}(e_i) = \mathbf{fv}(e')$ and $\mathbf{s}(e_i) \leq 2\mathbf{s}(e') + 2$; e' being fixed, there are thus finitely many possible values for e_i hence for i. So we do not need any additional condition for this reduction step to be well defined.

This reduction, however, is not suitable for simulating β-reduction because whenever the reduced β-redex is not in linear position, we need to reduce arbitrarily many resource redexes.

16. We must of course require that $\bigcup_{i \in I} \mathbf{fv}(e_i)$ is finite but, again, we will keep such requirements implicit in the following.

Example 2.5.1. *Observe that*

$$\tau((y)(\lambda x\, x)\, z) = \sum_{n,k_1,\dots,k_n \in \mathbf{N}} \frac{1}{n!k_1!\cdots k_n!} \langle y\rangle \left[\langle \lambda x\, x\rangle\, z^{k_1}, \dots, \langle \lambda x\, x\rangle\, z^{k_n}\right]$$

and

$$\tau((y)\, z) = \sum_{n\in\mathbf{N}} \frac{1}{n!} \langle y\rangle\, z^n.$$

Then the reduction from $\left[\langle \lambda x\, x\rangle\, z^{k_1}, \dots, \langle \lambda x\, x\rangle\, z^{k_n}\right]$ *to* z^n *if each* $k_i = 1$ *(resp. to* 0 *if one* $k_i \neq 1$*) requires firing* n *independent redexes (resp. one of those* n *redexes).*

2.5.1 Parallel resource reduction

One possible fix would be to replace $\to_\partial$ with $\to_\partial^*$ in the above definition, *i.e.* set $\epsilon \to \epsilon'$ as soon as $\epsilon = \sum_{i\in I} a_i.e_i$ and $\epsilon' = \sum_{i\in I} a_i.\epsilon_i'$ with $e_i \to_\partial^* \epsilon_i'$ for all $i \in I$, but then the study of the reduction subsumes that of normalization, which we treat in Section 2.8, and this relies on the possibility to simulate β-reduction steps.

A reasonable middle ground is to consider a parallel variant $\Rightarrow_\partial$ of $\to_\partial$, where any number of redexes can be reduced simultaneously in one step. The parallelism involved in the translation of a β-reduction step is actually quite constrained: like in the previous example, the redexes that need to be reduced in the Taylor expansion are always pairwise independent and no nesting is involved. However, in order to prove the confluence of the reduction on resource vectors, or its conservativity w.r.t. β-reduction, it is much more convenient to work with a fully parallel reduction relation, both on algebraic λ-terms and on resource vectors. Indeed, parallel reduction relations generally allow, e.g., to close confluence diagrams in one step or to define a maximal parallel reduction step: the relevance of this technical choice will be made clear all through Section 2.6.

Definition 2.5.2. We define *parallel resource reduction* $\Rightarrow_\partial \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$ inductively as follows:

- $x \Rightarrow_\partial x$;
- $\langle \lambda x\, s\rangle\, \bar{t} \Rightarrow_\partial \partial_x \sigma' \cdot \overline{\tau'}$ as soon as $s \Rightarrow_\partial \sigma'$ and $\bar{t} \Rightarrow_\partial \overline{\tau'}$;
- $\lambda x\, s \Rightarrow_\partial \lambda x\, \sigma'$ as soon as $s \Rightarrow_\partial \sigma'$;
- $\langle s\rangle\, \bar{t} \Rightarrow_\partial \langle \sigma'\rangle\, \overline{\tau'}$ as soon as $s \Rightarrow_\partial \sigma'$ and $\bar{t} \Rightarrow_\partial \overline{\tau'}$;
- $[s_1, \dots, s_n] \Rightarrow_\partial [\sigma_1', \dots, \sigma_n']$ as soon as $s_i \Rightarrow_\partial \sigma_i'$ for each $i \in \{1, \dots, n\}$.

We extend this reduction to sums of resource expressions by linearity: $\epsilon \Rightarrow_\partial \epsilon'$ if $\epsilon = \sum_{i=1}^n e_i$ and $\epsilon' = \sum_{i=1}^n \epsilon_i'$ with $e_i \Rightarrow_\partial \epsilon_i'$ for all $i \in \{1, \dots, n\}$.

It should be clear that $\to_\partial \subseteq \Rightarrow_\partial \subset \to_\partial^*$, Moreover observe that, because all term constructors are linear, the reduction rules extend naturally to finite sums of resource expressions: for instance, $\lambda x\, \sigma \Rightarrow_\partial \lambda x\, \sigma'$ as soon as $\sigma \Rightarrow_\partial \sigma'$.

We will prove in Sections 2.6 and 2.7 that this solution is indeed a good one: parallel resource reduction is strongly confluent, and there is a way to extend it to resource vectors so that not only the resulting reduction is strongly confluent and allows to simulate β-reduction, but any reduction step from the Taylor expansion of an algebraic term can be completed into a parallel β-reduction step. There are two pitfalls with this approach, though.

2.5.2 Size collapse

First, parallel reduction $\Rightarrow_\partial$ (like iterated reduction $\to_\partial^*$) lacks the combinatorial regularity properties of $\to_\partial$ given by Lemma 2.3.12: write $e \ggcurly_\partial e'$ if $e \Rightarrow_\partial \epsilon'$ with $e' \in |\epsilon'|$; $e' \in (!)\Delta$ being fixed, there is no bound on the size of the $\Rightarrow_\partial$-antecedents of e', *i.e.* those $e \in (!)\Delta$ such that $e \ggcurly_\partial e'$.

Example 2.5.3. *Fix $s \in \Delta$. Consider the sequences $\overrightarrow{u}(s)$ and $\overrightarrow{v}(s)$ of resource terms given by:*

$$\begin{cases} u_0(s) := s \\ u_{n+1}(s) := \langle \lambda y\, y \rangle\, [u_n(s)] \end{cases} \quad \textit{and} \quad \begin{cases} v_0(s) := s \\ v_{n+1}(s) := \langle \lambda y\, v_n(s) \rangle\, [] \end{cases}.$$

Observe that for all $n \in \mathbf{N}$, $u_{n+1}(s) \to_\partial u_n(s)$ and $v_{n+1}(s) \to_\partial v_n(s)$, and more generally, for all $n' \leq n$, $u_n(s) \Rightarrow_\partial u_{n'}(s)$ and $v_n(s) \Rightarrow_\partial v_{n'}(s)$. In particular $u_n(s) \Rightarrow_\partial s$ and $v_n(s) \Rightarrow_\partial s$ for all $n \in \mathbf{N}$.

Reducing all resource expressions in a resource vector simultaneously is thus no longer possible in general: consider, *e.g.*, $\sum_{n \in \mathbf{N}} u_n(x)$. As a consequence, when we introduce a reduction relation on resource vectors by extending a reduction relation on resource expressions as above, we must in general impose the summability of the family of reducts as a side condition:

Definition 2.5.4. Fix an arbitrary relation $\to\, \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$. For all $\epsilon, \epsilon' \in \mathbf{S}^{(!)\Delta}$, we write $\epsilon \widetilde{\to} \epsilon'$ whenever there exist families $(a_i)_{i\in I} \in \mathbf{S}^I$, $(e_i)_{i\in I} \in (!)\Delta^I$ and $(\epsilon'_i)_{i\in I} \in \mathbf{N}[(!)\Delta]^I$ such that:

- $(e_i)_{i\in I}$ is summable and $\epsilon = \sum_{i\in I} a_i.e_i$;
- $(\epsilon'_i)_{i\in I}$ is summable and $\epsilon' = \sum_{i\in I} a_i.\epsilon'_i$;
- for all $i \in I$, $e_i \to^? \epsilon'_i$.

The necessity of such a side condition forbids confluence. Indeed:

Example 2.5.5. *Let $\sigma = \sum_n u_n(v_n(x))$. Then $\sigma \widetilde{\Rightarrow}_\partial \sum u_n(x)$ and $\sigma \widetilde{\Rightarrow}_\partial \sum v_n(x)$, but since the only common reduct of $u_p(x)$ and $v_q(x)$ is x, there is no way*[17] *to close this pair of reductions: $(x)_{n\in\mathbf{N}}$ is not summable.*

These considerations lead us to study the combinatorics of parallel resource reduction more closely: in Section 2.6, we introduce successive variants of parallel reduction, based on restrictions on the nesting of fired redexes, and provide bounds for the size of antecedents of a resource expression. We moreover consider sufficient conditions for these restrictions to be preserved under reduction.

17. In fact, this argument is only valid if $\mathbf{S}$ is zerosumfree (*i.e.* if $a + b = 0 \in \mathbf{S}$ entails $a = b = 0$; see below, in particular Lemma 2.5.7), for instance if $\mathbf{S} = \mathbf{N}$: we rely on the fact that if $\sum_{i\in I} a_i.s_i = \sum_{n\in\mathbf{N}} u_n(x)$ then for all $i \in I$ such that $a_i \neq 0$, there is $n \in \mathbf{N}$ such that $s_i = u_n(x)$.

We then observe in Section 2.7 that, when applied to Taylor expansions, parallel reduction is automatically of the most restricted form, which allows us to provide uniform bounds and obtain the desired confluence and simulation properties.

2.5.3 Reduction structures

The other, *a priori* unrelated pitfall is the fact that the reduction can interact badly with the semimodule structure of $\mathbf{S}^{(!)\Delta}$: we can reproduce Example 2.4.16 in $\mathbf{S}^{(!)\Delta}$ through Taylor expansion (see the discussion in Section 2.7, p.58). Even more simply, we can use the terms of Example 2.5.3:

Example 2.5.6. *Let $s \in \Delta$ and $\sigma = \sum_{n\in\mathbf{N}} u_{n+1}(s) \in \mathbf{S}^{\Delta}$. Assuming $\mathbf{S}$ is a ring: $0 = \sigma + (-1).\sigma \mathrel{\widetilde{\Rightarrow}_\partial} \sum_{n\in\mathbf{N}} u_n(s) + (-1).\sigma = s$.*

Of course, this kind of issue does not arise when the semiring of coefficients is *zerosumfree*: recall that $\mathbf{S}$ is zerosumfree if $a + b = 0$ implies $a = b = 0$, which holds for all semirings of non-negative numbers, as well as for booleans. This prevents interferences between reductions and the semimodule structure:

Lemma 2.5.7. *Assume $\mathbf{S}$ is zerosumfree and fix a relation $\rightarrow \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$. If $\epsilon \mathrel{\widetilde{\rightarrow}} \epsilon'$ then, for all $e' \in |\epsilon'|$ there exists $e \in |\epsilon|$ and $\epsilon_0 \in \mathbf{N}[(!)\Delta]$ such that $e \rightarrow^? \epsilon_0$ and $e' \in |\epsilon_0|$.*

Proof. Assume $\epsilon = \sum_{i\in I} a_i.e_i$ and $\epsilon' = \sum_{i\in I} a_i.\epsilon'_i$ with $e_i \rightarrow^? \epsilon'_i$ for all $i \in I$. If $e' \in |\epsilon'|$ then there is $i \in I$ such that $e' \in |a_i.\epsilon'_i|$ hence $a_i \neq 0$ and $e' \in |\epsilon'_i|$. Then, since $\mathbf{S}$ is zerosumfree, $e_i \in |\epsilon|$. $\square$

Various alternative approaches to get rid of this restriction in the setting of the algebraic λ-calculus can be adapted to the reduction of resource vectors: we refer the reader to the literature on algebraic λ-calculi [2; AD08; 23; Día11] for several proposals. The linear-continuity of the resource λ-calculus allows us to propose a novel approach: consider possible restrictions on the families of resource expressions simultaneously reduced in a $\widetilde{\rightarrow}$-step.

Definition 2.5.8. We call *resource support* any set $\mathcal{E} \subseteq (!)\Delta$ of resource expressions such that $\mathbf{fv}(\mathcal{E}) = \bigcup_{e\in\mathcal{E}} \mathbf{fv}(e)$ is finite. Then a *resource structure* is any set $\mathfrak{E} \subseteq \mathfrak{P}((!)\Delta)$ of resource supports such that:

- $\mathfrak{E}$ contains all finite resource supports;
- $\mathfrak{E}$ is closed under finite unions;
- $\mathfrak{E}$ is downwards closed for inclusion.

The maximal resource structure is $(!)\mathfrak{F}_{\mathbf{fv}} := \{\mathcal{E} \subseteq (!)\Delta \; ; \; \mathbf{fv}(\mathcal{E}) \text{ is finite}\}$, which is also a finiteness structure [Ehr10]. Observe that any finiteness structure $\mathfrak{F} \subseteq (!)\mathfrak{F}_{\mathbf{fv}}$ is a resource structure: all three additional conditions are automatically satisfied.

Definition 2.5.9. Fix a relation $\rightarrow \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$. For all resource support $\mathcal{E}$, we write $\widetilde{\rightarrow}_{\mathcal{E}}$ for $\widetilde{\rightharpoonup}_{\mathcal{E}}$ where $\rightharpoonup_{\mathcal{E}}$ denotes $\rightarrow \cap (\mathcal{E} \times \mathbf{N}[(!)\Delta])$. For all resource structure $\mathfrak{E}$, we then write $\widetilde{\rightarrow}_{\mathfrak{E}}$ for $\bigcup_{\mathcal{E}\in\mathfrak{E}} \widetilde{\rightarrow}_{\mathcal{E}}$.

We have $\widetilde{\to}_{\mathcal{E}} \subseteq \widetilde{\to} \cap (\mathbf{S}^{\mathcal{E}} \times \mathbf{S}^{(!)\Delta})$, but in general the reverse inclusion holds only if $\mathbf{S}$ is zerosumfree: in this latter case $\epsilon \mathrel{\widetilde{\Rightarrow}_{\partial}} \epsilon'$ iff $\epsilon \mathrel{\widetilde{\Rightarrow}_{\partial_{|\epsilon|}}} \epsilon'$.

Definition 2.5.10. We call $\to$-*reduction structure* any resource structure $\mathfrak{E}$ such that if $\mathcal{E} \in \mathfrak{E}$ then $\bigcup \left\{ |\epsilon'| \ ; \ e \in \mathcal{E} \text{ and } e \to^? \epsilon' \right\} \in \mathfrak{E}$.

We will consider some particular choices of reduction structure in the following, but the point is that our approach is completely generic. The results of Section 2.7 will imply that if $\mathfrak{S} \subseteq \mathfrak{P}(\mathbf{S}^{\Delta})$ is a $\Rightarrow_\partial$-reduction structure containing $|\tau(M)|$ then one can translate any $\Rightarrow_\beta$-reduction sequence from M into a $\widetilde{\Rightarrow}_{\partial_{\mathfrak{S}}}$-reduction sequence from $\tau(M)$. Additional properties such as the confluence of $\widetilde{\Rightarrow}_{\partial_{\mathfrak{S}}}$, its conservativity over $\Rightarrow_\beta$, or its compatibility with normalization will depend on additional conditions on $\mathfrak{S}$.

2.6 Taming the size collapse of parallel resource reduction

In this section, we study successive families of restrictions of the parallel resource reduction $\Rightarrow_\partial$. Our purpose is to enforce some control on the size collapse induced by $\Rightarrow_\partial$, so as to obtain a confluent restriction of $\widetilde{\Rightarrow}_\partial$, all the while retaining enough parallelism to simulate parallel β-reduction on algebraic λ-terms, ideally in a conservative way.

First observe that parallel resource reduction itself is strongly confluent as expected: following a classic argument, we define $\mathsf{F}(e)$ as the result of firing all redexes in e and then, whenever $e \Rightarrow_\partial \epsilon'$, we have $\epsilon' \Rightarrow_\partial \mathsf{F}(e)$. Formally:

Definition 2.6.1. For all $e \in (!)\Delta$ we define the *full parallel reduct* $\mathsf{F}(e)$ of e by induction on e as follows:

$$
\begin{aligned}
\mathsf{F}(x) &:= x \\
\mathsf{F}(\lambda x\, s) &:= \lambda x\, \mathsf{F}(s) \\
\mathsf{F}\big(\langle \lambda x\, s\rangle\, \bar{t}\big) &:= \partial_x \mathsf{F}(s) \cdot \mathsf{F}\big(\bar{t}\big) \\
\mathsf{F}\big(\langle s\rangle\, \bar{t}\big) &:= \langle \mathsf{F}(s)\rangle\, \mathsf{F}\big(\bar{t}\big) && \text{(if } s \text{ is not an abstraction)} \\
\mathsf{F}([s_1, \dots, s_n]) &:= [\mathsf{F}(s_1), \dots, \mathsf{F}(s_n)].
\end{aligned}
$$

Then if $\epsilon = \sum_{i=1}^n e_i \in \mathbf{N}[(!)\Delta]$, we set $\mathsf{F}(\epsilon) = \sum_{i=1}^n \mathsf{F}(e_i)$.

Lemma 2.6.2. *For all $\epsilon, \epsilon' \in \mathbf{N}[(!)\Delta]$, if $\epsilon \Rightarrow_\partial \epsilon'$ then $\epsilon' \Rightarrow_\partial \mathsf{F}(\epsilon)$.*

Proof. Follows directly from the definitions. □

In general, however, if we fix $e' \in (!)\Delta$ then there is no bound on those $e \in (!)\Delta$ such that $e' \in |\mathsf{F}(e)|$, so we cannot extend F on $\mathbf{S}^{(!)\Delta}$, nor generalize Lemma 2.6.2 to $\widetilde{\Rightarrow}_\partial$. Indeed, we have shown that $\widetilde{\Rightarrow}_\partial$ is not even confluent.

In order to understand what restrictions are necessary to recover confluence, we first provide a close inspection of the combinatorial effect of $\Rightarrow_\partial$ on the size of resource expressions: we show in subsection 2.6.1 that bounding the length of chains of immediately nested fired redexes is enough to bound the size of $\Rightarrow_\partial$-antecedents of a fixed resource expression.

In order to close a pair of reductions $e \Rightarrow_\partial \epsilon'$ and $e \Rightarrow_\partial \epsilon''$, we have to reduce at least the residuals in ϵ' of the redexes fired in the reduction $e \Rightarrow_\partial \epsilon''$ (and vice versa). So we want the above bounds to be stable under taking the unions of sets of redexes in a term: it is not the case if we consider chains of immediately nested redexes. In Subsection 2.6.2, we extend the boundedness condition to all chains of nested fired redexes and introduce the family $\left(\Rightarrow_{(b)}\right)_{b\in\mathbf{N}}$ of boundedly nested parallel reductions. We then show that this family enjoys a kind of diamond property (Lemma 2.6.14), which can then be extended to $\widetilde{\Rightarrow_{(\partial)}} = \bigcup_{b\in\mathbf{N}} \widetilde{\Rightarrow_{(b)}}$. We must require that $\mathbf{S}$ enjoys an additional *additive splitting* property (see Definition 2.6.15), in order to "align" the $\Rightarrow_{(b)}$-reductions involved in both sides of a pair of $\widetilde{\Rightarrow_{(b)}}$-reductions from the same resource vector (see the proof of Lemma 2.6.17).

To get rid of the additive splitting hypothesis we must further restrict resource reduction so as to recover a notion of full reduct *at bounded depth*. It is not sufficient to bound the depth of fired redexes because this is not stable under reduction. In Subsection 2.6.3, we rather introduce the parallel reduction $\Rightarrow_{\lfloor d\rfloor}$ where substituted variables occur at depth at most d. We then show that $\widetilde{\Rightarrow_{\lfloor\partial\rfloor}} = \bigcup_{d\in\mathbf{N}} \widetilde{\Rightarrow_{\lfloor d\rfloor}}$ is strongly confluent by proving that any $\Rightarrow_{\lfloor d\rfloor}$-step from ϵ can be followed by a $\Rightarrow_{\lfloor d'\rfloor}$-reduction to $\mathsf{F}_{\lfloor d\rfloor}(\epsilon)$, where $\mathsf{F}_{\lfloor d\rfloor}(\epsilon)$ is obtained by firing all redexes in ϵ for which the bound variables occur at depth at most d, and d' depends only on d.

Finally, we consider resource vectors of bounded height: these contain the Taylor expansions of algebraic λ-terms. We show that all the above restrictions actually coincide with $\widetilde{\Rightarrow_\partial}$ on bounded resource vectors. In this particular case, we can actually extend F by linear-continuity and obtain a proof of the diamond property for $\widetilde{\Rightarrow_\partial}$. [18]

At this point of the discussion, it is worth noting that, if we extend a relation $\rightarrow\ \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$ to a binary relation on finite sums of resource expressions so that $\epsilon \rightarrow \epsilon'$ iff $\epsilon = \sum_{i=1}^{n} e_i$ and $\epsilon' = \sum_{i=1}^{n} \epsilon'_i$ with $e_i \rightarrow \epsilon'_i$ for all $i \in \{1, \dots, n\}$, then for all $\rightarrow$-reduction structure $\mathfrak{C}$ and all resource vectors $\epsilon, \epsilon' \in \mathbf{S}^{(!)\Delta}$, we have $\epsilon \mathrel{\widetilde{\rightarrow}_{\mathfrak{C}}} \epsilon'$ iff there exist a set I of indices, a resource support $\mathcal{E} \in \mathfrak{C}$, a family $(a_i)_{i\in I} \in \mathbf{S}^I$ of scalars and families $(\epsilon_i)_{i\in I} \in \mathbf{N}[\mathcal{E}]^I$ and $(\epsilon'_i)_{i\in I} \in \mathbf{N}[(!)\Delta]^I$ such that:

- $(\epsilon_i)_{i\in I}$ is summable and $\epsilon = \sum_{i\in I} a_i.\epsilon_i$;
- $(\epsilon'_i)_{i\in I}$ is summable and $\epsilon' = \sum_{i\in I} a_i.\epsilon'_i$;
- for all $i \in I$, $\epsilon_i \rightarrow \epsilon'_i$.

We will use this fact for confluence proofs: $\Rightarrow_\partial$ and its variants are all of this form.

18. Note that, although they involve increasing constraints on parallel reduction, Subsections 2.6.1 to 2.6.3 are essentially pairwise independent. Moreover, we obtain the diamond property for $\widetilde{\Rightarrow_\partial}$ on bounded resource vectors as a consequence of the results of Subsection 2.6.3, but it could as well be proved directly, using similar techniques (see Footnote 21, p.58). So, the reader who only wants the proofs necessary for the main results of the paper can skip Subsections 2.6.1 and 2.6.2; the reader who is not interested in checking proofs can also skip subsection 2.6.3.

We chose to present the successive families of restrictions anyway, because their construction provides a precise understanding of the combinatorics of parallel resource reduction, and of the various ingredients involved in designing a strongly confluent version of $\widetilde{\Rightarrow_\partial}$: we start by avoiding the size collapse by putting a restriction on families of redexes that can be fired in parallel; then we ensure that this restriction is stable under reduction.

This understanding plays a key rôle in enabling the generalization of our approach to linear logic proof nets or infinitary λ-calculus: with Chouquet, we have recently established that our restrictions on the nesting of redexes, as well as their preservation under reduction, can be adapted to the setting of proof nets [14]; and preliminary work on infinitary λ-calculus indicates that it could be amenable to the technique of Subsection 2.6.2, whereas it does not make sense to restrict the depth of substituted variables in this setting.

2.6.1 Bounded chains of redexes

Definition 2.6.3. We define a family of relations $\Rightarrow_{(m|k)} \subseteq (!)\Delta \times \mathbf{N}[(!)\Delta]$ for $m \leq k \in \mathbf{N}$ inductively as follows:

- $x \Rightarrow_{(m|k)} x$ for all $m \leq k \in \mathbf{N}$;
- $\lambda x\, s \Rightarrow_{(m|k)} \lambda x\, \sigma'$ if $m \leq k$ and $s \Rightarrow_{(m_1|k)} \sigma'$ for some $m_1 \leq k$;
- $\langle s\rangle\, \bar{t} \Rightarrow_{(m|k)} \langle \sigma'\rangle\, \overline{\tau'}$ if $m \leq k$, $s \Rightarrow_{(m_1|k)} \sigma'$ and $\bar{t} \Rightarrow_{(m_2|k)} \overline{\tau'}$ for some $m_1 \leq k$ and $m_2 \leq k$;
- $[s_1, \ldots, s_r] \Rightarrow_{(m|k)} [\sigma'_1, \ldots, \sigma'_r]$ if $s_i \Rightarrow_{(m|k)} \sigma'_i$ for all $i \in \{1, \ldots, r\}$;
- $\langle \lambda x\, s\rangle\, \bar{t} \Rightarrow_{(m|k)} \partial_x \sigma' \cdot \overline{\tau'}$ if $0 < m \leq k$, $s \Rightarrow_{(m-1|k)} \sigma'$ and $\bar{t} \Rightarrow_{(m-1|k)} \overline{\tau'}$.

Intuitively, we have $e \Rightarrow_{(m|k)} \epsilon'$ iff $e \Rightarrow_\partial \epsilon'$ and that reduction fires chains of redexes of length at most k, those starting at top level being of length at most m. In particular, it should be clear that if $e \Rightarrow_{(m|k)} \epsilon'$ then $e \Rightarrow_\partial \epsilon'$, and $e \Rightarrow_{(m'|k')} \epsilon'$ as soon as $m \leq m' \leq k'$ and $k \leq k'$. Moreover, $e \Rightarrow_\partial \epsilon'$ iff $e \Rightarrow_{(\mathbf{h}(e)|\mathbf{h}(e))} \epsilon'$.

Definition 2.6.4. We define $\mathbf{gb}_k(l, m) \in \mathbf{N}$ for all $k, l, m \in \mathbf{N}$, by induction on the lexicographically ordered pair (l, m):

$$\begin{array}{rcl} \mathbf{gb}_k(0,0) & := & 0 \\ \mathbf{gb}_k(l+1,0) & := & \mathbf{gb}_k(l,k)+1 \\ \mathbf{gb}_k(l,m+1) & := & 4\mathbf{gb}_k(l,m). \end{array}$$

We then write $\mathbf{gb}_k(l) := \mathbf{gb}_k(l, k)$.

For all $k, l, m \in \mathbf{N}$, the following identities follow straightforwardly from the definition and will be used throughout this subsection:

$$\begin{array}{rcl} \mathbf{gb}_k(l,m) & = & 4^m \mathbf{gb}_k(l,0) \\ \mathbf{gb}_k(0,m) & = & 0 \\ \mathbf{gb}_k(1,m) & = & 4^m. \end{array}$$

Lemma 2.6.5. *For all $k, l, l', m \in \mathbf{N}$, $\mathbf{gb}_k(l + l', m) \geq \mathbf{gb}_k(l, m) + \mathbf{gb}_k(l', m)$.*

Proof. By induction on l'. The case $l' = 0$ is direct. Assume the result holds for l', we prove it for $l' + 1$:

$$\begin{array}{rcl} \mathbf{gb}_k\big(l+l'+1,m\big) & = & 4^m\big(\mathbf{gb}_k\big(l+l'+1,0\big)\big) \\ & = & 4^m\big(\mathbf{gb}_k\big(l+l',k\big)+1\big) \\ & \geq & 4^m\big(\mathbf{gb}_k(l,k)+\mathbf{gb}_k\big(l',k\big)+1\big) \\ & = & 4^m\Big(4^k\mathbf{gb}_k(l,0)+\mathbf{gb}_k\big(l'+1,0\big)\Big) \\ & \geq & 4^m\mathbf{gb}_k(l,0)+4^m\mathbf{gb}_k\big(l'+1,0\big) \\ & = & \mathbf{gb}_k(l,m)+\mathbf{gb}_k\big(l'+1,m\big). \end{array}$$

∎

The following generalization follows directly:

Corollary 2.6.6. *For all $l_1 \ldots, l_n \in \mathbf{N}$*

$$\mathbf{gb}_k\left(\sum_{i=1}^{n} l_i, m\right) \geq \sum_{i=1}^{n} \mathbf{gb}_k(l_i, m).$$

Lemma 2.6.7. *For all $k, l, m \in \mathbf{N}$, $\mathbf{gb}_k(l, m) \geq l$.*

Proof. By Corollary 2.6.6, $\mathbf{gb}_k(l, m) \geq l \times \mathbf{gb}_k(1, m) = l \times 4^m$. □

Lemma 2.6.8. *For all $k, k', l, l', m, m' \in \mathbf{N}$ if $k \leq k'$, $l \leq l'$ and $m \leq m'$, then:*

$$\mathbf{gb}_k(l, m) \leq \mathbf{gb}_{k'}(l', m').$$

Proof. We prove the monotonicity of $\mathbf{gb}_k(l, m)$ in m, l and then k, separately.

First, if $m \leq m'$ then $\mathbf{gb}_k(l, m) = 4^m \mathbf{gb}_k(l, 0) \leq 4^{m'} \mathbf{gb}_k(l, 0) = \mathbf{gb}_k(l, m')$.

By Lemma 2.6.5, if $l \leq l'$, $\mathbf{gb}_k(l', m) \geq \mathbf{gb}_k(l, m) + \mathbf{gb}_k(l' - l, m) \geq \mathbf{gb}_k(l, m)$.

Finally, we prove that if $k \leq k'$ then $\mathbf{gb}_k(l, m) \leq \mathbf{gb}_{k'}(l, m)$ by induction on the lexicographically ordered pair (l, m):

$$\begin{array}{rcl}
\mathbf{gb}_k(0,0) & = & 0 \\
& = & \mathbf{gb}_{k'}(0,0) \\
\mathbf{gb}_k(l+1,0) & = & \mathbf{gb}_k(l,k)+1 \\
& \leq & \mathbf{gb}_{k'}(l,k)+1 \\
& \leq & \mathbf{gb}_{k'}(l,k')+1 \\
& = & \mathbf{gb}_{k'}(l+1,0) \\
\mathbf{gb}_k(l,m+1) & = & 4\mathbf{gb}_k(l,m) \\
& \leq & 4\mathbf{gb}_{k'}(l,m) \\
& = & \mathbf{gb}_{k'}(l,m+1)
\end{array}$$

□

Write $e \ggcurly_{(m|k)} e'$ if $e \Rightarrow_{(m|k)} \epsilon'$ with $e' \in |\epsilon'|$.

Lemma 2.6.9. *If $e \ggcurly_{(m|k)} e'$ then $\mathbf{s}(e) \leq \mathbf{gb}_k(\mathbf{s}(e'), m)$.*

Proof. By induction on the reduction $e \Rightarrow_{(m|k)} \epsilon'$ such that $e' \in \epsilon'$.

If $e = x = \epsilon'$ then $e' = x$ and $\mathbf{s}(e) = 1 = \mathbf{gb}_0(1, 0) \leq \mathbf{gb}_k(\mathbf{s}(e'), m)$.

If $e = \lambda x\, s$, $\epsilon' = \lambda x\, \sigma'$, $m \leq k$ and $s \Rightarrow_{(m_1|k)} \sigma'$ with $m_1 \leq k$, then $e' = \lambda x\, s'$ with $s \ggcurly_{(m_1|k)} s'$. We obtain:

$$\begin{aligned}
\mathbf{s}(e) &= \mathbf{s}(s) + 1 \\
&\leq \mathbf{gb}_k(\mathbf{s}(s'), m_1) + 1 && \text{(by induction hypothesis)} \\
&\leq \mathbf{gb}_k(\mathbf{s}(s'), k) + 1 \\
&= \mathbf{gb}_k(\mathbf{s}(s') + 1, 0) \\
&\leq \mathbf{gb}_k(\mathbf{s}(e'), m).
\end{aligned}$$

If $e = \langle s \rangle \bar{t}$, $\epsilon' = \langle \sigma' \rangle \overline{\tau'}$, $m \leq k$, $s \Rightarrow_{(m_1|k)} \sigma'$ and $\bar{t} \Rightarrow_{(m_2|k)} \overline{\tau'}$ with $m_i \leq k$ for all $i \in \{1, 2\}$, then $e' = \langle s' \rangle \overline{t'}$ with $s \ggg_{(m_1|k)} s'$ and $\bar{t} \ggg_{(m_2|k)} \overline{t'}$. We obtain:

$$\begin{aligned}
\mathbf{s}(e) &= \mathbf{s}(s) + \mathbf{s}(\bar{t}) + 1 \\
&\leq \mathbf{gb}_k(\mathbf{s}(s'), m_1) + \mathbf{gb}_k(\mathbf{s}(\overline{t'}), m_2) + 1 && \text{(by induction hypothesis)} \\
&\leq \mathbf{gb}_k(\mathbf{s}(s'), k) + \mathbf{gb}_k(\mathbf{s}(\overline{t'}), k) + 1 \\
&\leq \mathbf{gb}_k(\mathbf{s}(s') + \mathbf{s}(\overline{t'}), k) + 1 \\
&= \mathbf{gb}_k(\mathbf{s}(s') + \mathbf{s}(\overline{t'}) + 1, 0) \\
&\leq \mathbf{gb}_k(\mathbf{s}(e'), m).
\end{aligned}$$

If $e = [s_1, \ldots, s_r]$, $\epsilon' = [\sigma'_1, \ldots, \sigma'_r]$ and $s_i \Rightarrow_{(m|k)} \sigma'_i$ for all $i \in \{1, \ldots, r\}$, then $e' = [s'_1, \ldots, s'_r]$ with $s_i \ggg_{(m|k)} s'_i$ for all $i \in \{1, \ldots, r\}$. We obtain:

$$\begin{aligned}
\mathbf{s}(e) &= \sum_{i=1}^{r} \mathbf{s}(s_i) \\
&\leq \sum_{i=1}^{r} \mathbf{gb}_k(\mathbf{s}(s'_i), m) && \text{(by induction hypothesis)} \\
&\leq \mathbf{gb}_k\left(\sum_{i=1}^{r} \mathbf{s}(s'_i), m\right) \\
&= \mathbf{gb}_k(\mathbf{s}(e'), m).
\end{aligned}$$

If $e = \langle \lambda x\, s \rangle \bar{t}$, $\epsilon' = \partial_x \sigma' \cdot \overline{\tau'}$, $0 < m \leq k$, $s \Rightarrow_{(m-1|k)} \sigma'$ and $\bar{t} \Rightarrow_{(m-1|k)} \overline{\tau'}$, then there are $s' \in |\sigma'|$ and $\overline{t'} \in |\overline{\tau'}|$ such that $e' \in \left|\partial_x s' \cdot \overline{t'}\right|$. In particular, $s \ggg_{(m-1|k)} s'$ and $\bar{t} \ggg_{(m-1|k)} \overline{t'}$ and we obtain:

$$\begin{aligned}
\mathbf{s}(e) &= \mathbf{s}(s) + \mathbf{s}(\bar{t}) + 2 \\
&\leq \mathbf{gb}_k(\mathbf{s}(s'), m-1) + \mathbf{gb}_k(\mathbf{s}(\overline{t'}), m-1) + 2 && \text{(by induction hypothesis)} \\
&\leq 2\mathbf{gb}_k(\mathbf{s}(e'), m-1) + 2 && (\mathbf{s}(e') \geq \max\{\mathbf{s}(s'), \mathbf{s}(\overline{t'})\}) \\
&\leq 4\mathbf{gb}_k(\mathbf{s}(e'), m-1) && (\mathbf{s}(e') \geq \mathbf{s}(s') \geq 1) \\
&= \mathbf{gb}_k(\mathbf{s}(e'), m).
\end{aligned}$$

□

As a direct consequence, for all $m \leq k \in \mathbf{N}$, for all summable family $(e_i)_{i \in I}$ and all family $(\epsilon'_i)_{i \in I}$ such that $e_i \Rightarrow_{(m|k)} \epsilon'_i$ for all $i \in I$, $(\epsilon'_i)_{i \in I}$ is also summable: we can thus drop the side condition in the definition of $\widetilde{\Rightarrow_{(m|k)}}$.

Observe however that those reduction relations are not stable under taking the unions of fired redexes in families of reduction steps: using, *e.g.*, the terms $u_n(s)$ from Example 2.5.3, for all $n \in \mathbf{N}$, we have $u_{2n}(s) \Rightarrow_{(1|1)} u_n(s)$ by firing all redexes at even depth, $u_{2n}(s) \Rightarrow_{(0|1)} u_n(s)$ by firing all redexes at odd depth, and $u_{2n}(s) \Rightarrow_{(2n|2n)} s$ by firing both families, but there is obviously no $k \in \mathbf{N}$ such that $u_{2n}(s) \Rightarrow_{(k|k)} s$ uniformly for all $n \in \mathbf{N}$. Although we can close the induced critical pair

$$\sum_{n\in\mathbf{N}} u_{2n}(s) \widetilde{\Rightarrow_{(0|1)}} \sum_{n\in\mathbf{N}} u_n(s) \text{ and } \sum_{n\in\mathbf{N}} u_{2n}(s) \widetilde{\Rightarrow_{(1|1)}} \sum_{n\in\mathbf{N}} u_n(s)$$

trivially in this case, this phenomenon is an obstacle to confluence:

Example 2.6.10. *Fix $s \in \Delta$ and consider the sequence $\overrightarrow{w}(s)$ of resource terms given by $w_0(s) = s$ and:*

$$\begin{array}{rcl} w_{2n+1}(s) & = & \langle \lambda y\, y \rangle\, [w_{2n}(s)] \\ w_{2n+2}(s) & = & \langle \lambda y\, w_{2n+1}(s) \rangle\, [] \end{array}$$

Then for all $n \in \mathbf{N}$, $w_{2n}(s) \Rightarrow_{(1|1)} u_n(s)$, $w_{2n+1}(s) \Rightarrow_{(0|1)} u_n(s)$, $w_{2n}(s) \Rightarrow_{(0|1)} v_n(s)$, and $w_{2n+1}(s) \Rightarrow_{(1|1)} v_n(s)$. Then for instance

$$\sum_{n\in\mathbf{N}} w_{2n}(s) \widetilde{\Rightarrow_{(1|1)}} \sum_{n\in\mathbf{N}} u_n(s) \quad \textit{and} \quad \sum_{n\in\mathbf{N}} w_{2n}(s) \widetilde{\Rightarrow_{(0|1)}} \sum_{n\in\mathbf{N}} v_n(s)$$

but we know from Example 2.5.5 that this pair of reductions cannot be closed in general.

2.6.2 Boundedly nested redexes

From the previous subsection, it follows that bounding the length of chains of immediately nested redexes allows to tame the size collapse of resource expressions under reduction, but we need to further restrict this notion in order to keep it stable under unions of fired redex sets. A natural answer is to require a bound on the depth of the nesting of fired redexes, regardless of the distance between them:

Definition 2.6.11. We define a family of relations $\left(\Rightarrow_{(b)}\right)_{b\in\mathbf{N}}$ inductively as follows:

- $x \Rightarrow_{(b)} x$ for all $b \in \mathbf{N}$;
- $\lambda x\, s \Rightarrow_{(b)} \lambda x\, \sigma'$ if $s \Rightarrow_{(b)} \sigma'$;
- $\langle s \rangle\, \bar{t} \Rightarrow_{(b)} \langle \sigma' \rangle\, \overline{\tau'}$ if $s \Rightarrow_{(b)} \sigma'$ and $\bar{t} \Rightarrow_{(b)} \overline{\tau'}$;
- $[s_1, \dots, s_r] \Rightarrow_{(b)} [\sigma'_1, \dots, \sigma'_r]$ if $s_i \Rightarrow_{(b)} \sigma'_i$ for all $i \in \{1, \dots, r\}$;
- $\langle \lambda x\, s \rangle\, \bar{t} \Rightarrow_{(b)} \partial_x \sigma' \cdot \overline{\tau'}$ if $b \geq 1$, $s \Rightarrow_{(b-1)} \sigma'$ and $\bar{t} \Rightarrow_{(b-1)} \overline{\tau'}$.

Intuitively, we have $e \Rightarrow_{(b)} \epsilon'$ iff $e \Rightarrow_\partial \epsilon'$ and every branch of e (seen as a rooted tree) crosses at most b fired redexes. In particular it should be clear that if $e \Rightarrow_{(b)} \epsilon'$ then $e \Rightarrow_{(b|b)} \epsilon'$, and moreover $e \Rightarrow_{(b')} \epsilon'$ for all $b' \geq b$. Moreover observe that $e \Rightarrow_{(\mathbf{h}(e))} \epsilon'$ whenever $e \Rightarrow_\partial \epsilon'$, hence $\Rightarrow_\partial = \bigcup_{b\in\mathbf{N}} \Rightarrow_{(b)}$.

Write $e \ggcurly_{(b)} e'$ if $e \Rightarrow_{(b)} \epsilon'$ with $e' \in |\epsilon'|$. If $e \ggcurly_{(b)} e'$, then $e \ggcurly_{(b|b)} e'$ and we thus know that $\mathbf{s}(e) \leq \mathbf{gb}_b(\mathbf{s}(e'))$. In this special case, we can in fact provide a much better bound:

Lemma 2.6.12. *If $e \ggcurly_{(b)} e'$ then $\mathbf{s}(e) \leq 4^b \mathbf{s}(e')$.*

Proof. By induction on the reduction $e \Rightarrow_{(b)} \epsilon'$ such that $e' \in |\epsilon'|$.

If $e = x = \epsilon'$ then $e' = x$ and $\mathbf{s}(e) = 1 \leq 4^b = 4^b\mathbf{s}(e')$.

If $e = \lambda x\, s$ and $\epsilon' = \lambda x\, \sigma'$ with $s \Rightarrow_{(b)} \sigma'$, then $e' = \lambda x\, s'$ with $s \ggg_{(b)} s'$. By induction hypothesis, $\mathbf{s}(s) \leq 4^b\mathbf{s}(s')$. Then $\mathbf{s}(e) = \mathbf{s}(s) + 1 \leq 4^b\mathbf{s}(s') + 1 \leq 4^b(\mathbf{s}(s') + 1) = 4^b\mathbf{s}(e')$.

If $e = \langle s\rangle\, \overline{t}$, $\epsilon' = \langle \sigma'\rangle\, \overline{\tau'}$, $s \Rightarrow_{(b)} \sigma'$ and $\overline{t} \Rightarrow_{(b)} \overline{\tau'}$, then $e' = \langle s'\rangle\, \overline{t'}$ with $s \ggg_{(b)} s'$ and $\overline{t} \ggg_{(b)} \overline{t'}$. By induction hypothesis, $\mathbf{s}(s) \leq 4^b\mathbf{s}(s')$ and $\mathbf{s}(\overline{t}) \leq 4^b\mathbf{s}(\overline{t'})$. Then $\mathbf{s}(e) = \mathbf{s}(s) + \mathbf{s}(\overline{t}) + 1 \leq 4^b\mathbf{s}(s') + 4^b\mathbf{s}(\overline{t'}) + 1 \leq 4^b\big(\mathbf{s}(s') + \mathbf{s}(\overline{t'}) + 1\big) = 4^b\mathbf{s}(e')$.

If $e = [s_1, \dots, s_r]$, $\epsilon' = [\sigma'_1, \dots, \sigma'_r]$ and $s_i \Rightarrow_{(b)} \sigma'_i$ for all $i \in \{1, \dots, r\}$, then $e' = [s'_1, \dots, s'_r]$ with $s_i \ggg_{(b)} s'_i$ for all $i \in \{1, \dots, r\}$. By induction hypothesis, $\mathbf{s}(s_i) \leq 4^b\mathbf{s}(s'_i)$ for all $i \in \{1, \dots, r\}$ and then $\mathbf{s}(e) = \sum_{i=1}^r \mathbf{s}(s_i) \leq \sum_{i=1}^r 4^b\mathbf{s}(s'_i) = 4^b\mathbf{s}(e')$.

If $e = \langle \lambda x\, s\rangle\, \overline{t}$, $\epsilon' = \partial_x\sigma' \cdot \overline{\tau'}$, $b > 0$, $s \Rightarrow_{(b-1)} \sigma'$ and $\overline{t} \Rightarrow_{(b-1)} \overline{\tau'}$, then there are $s' \in |\sigma'|$ and $\overline{t'} \in \left|\overline{\tau'}\right|$ such that $e' \in \left|\partial_x s' \cdot \overline{t'}\right|$. In particular, $s \ggg_{(b-1)} s'$ and $\overline{t} \ggg_{(b-1)} \overline{t'}$ and, by induction hypothesis, $\mathbf{s}(s) \leq 4^{b-1}\mathbf{s}(s')$ and $\mathbf{s}(\overline{t}) \leq 4^{b-1}\mathbf{s}(\overline{t'})$. Writing $n = \mathbf{n}_x(s') = \#\overline{t'}$, we have:

$$\begin{aligned}
4^b\mathbf{s}(e') &= 4^b\big(\mathbf{s}(s') + \mathbf{s}(\overline{t'}) - n\big) \\
&= 4^{b-1}\big(\mathbf{s}(s') + \mathbf{s}(\overline{t'}) + 3\mathbf{s}(s') + 3\mathbf{s}(\overline{t'}) - 4n\big) && (n \leq \mathbf{s}(s') \text{ and } n \leq \mathbf{s}(\overline{t'})) \\
&\geq 4^{b-1}\big(\mathbf{s}(s') + \mathbf{s}(\overline{t'}) + 2\mathbf{s}(s')\big) && (\mathbf{s}(s') \geq 1) \\
&\geq 4^{b-1}\big(\mathbf{s}(s') + \mathbf{s}(\overline{t'})\big) + 2 \\
&\geq \mathbf{s}(s) + \mathbf{s}(\overline{t}) + 2 \\
&= \mathbf{s}(e).
\end{aligned}$$

□

Like for parallel reduction (Definition 2.5.2), we extend each $\Rightarrow_{(b)}$ to sums of resource expressions by linearity: $\epsilon \Rightarrow_{(b)} \epsilon'$ if $\epsilon = \sum_{i=1}^n e_i$ and $\epsilon' = \sum_{i=1}^n \epsilon'_i$ with $e_i \Rightarrow_{(b)} \epsilon'_i$ for all $i \in \{1, \dots, n\}$. Again, because all term constructors are linear, the reduction rules extend naturally to finite sums of resource expressions: for instance, $\langle \lambda x\, \sigma\rangle\, \overline{\tau} \Rightarrow_{(b)} \partial_x\sigma' \cdot \overline{\tau'}$ as soon as $b \geq 1$, $\sigma \Rightarrow_{(b-1)} \sigma'$ and $\overline{\tau} \Rightarrow_{(b-1)} \overline{\tau'}$.

The relations $\Rightarrow_{(b)}$ are then stable under unions of families of fired redexes, avoiding pitfalls such as that of Example 2.6.10.

Lemma 2.6.13. *If $e \Rightarrow_{(b_0)} \epsilon'$ and $\overline{u} \Rightarrow_{(b_1)} \overline{\upsilon'}$ then $\partial_x e \cdot \overline{u} \Rightarrow_{(b_0+b_1)} \partial_x\epsilon' \cdot \overline{\upsilon'}$.*

Proof. Write $\overline{u} = [u_1, \dots, u_n]$. Then we can write $\overline{\upsilon'} = [\upsilon'_1, \dots, \upsilon'_n]$ with $u_i \Rightarrow_{(b_1)} \upsilon'_i$ for all $i \in \{1, \dots, n\}$. Recall that whenever $I = \{i_1, \dots, i_k\} \subseteq \{1, \dots, n\}$ with $\#I = k$, we write $\overline{u}_I = [u_{i_1}, \dots, u_{i_k}]$ and $\overline{\upsilon'}_I = \left[\upsilon'_{i_1}, \dots, \upsilon'_{i_k}\right]$.

The proof is by induction on the reduction $e \Rightarrow_{(b_0)} \epsilon'$. If $e = y = \epsilon'$ then:

- if $y = x$ and $n = 1$ then $\partial_x e \cdot \overline{u} = u_1 \Rightarrow_{(b_1)} \upsilon'_1 = \partial_x\epsilon' \cdot \overline{\upsilon'}'$;
- if $y \neq x$ and $\overline{u} = []$ then $\partial_x e \cdot \overline{u} = y \Rightarrow_{(0)} y = \partial_x\epsilon' \cdot \overline{\upsilon'}$;
- otherwise, $\partial_x e \cdot \overline{u} = 0 \Rightarrow_{(0)} 0 = \partial_x\epsilon' \cdot \overline{\upsilon'}$.

If $e = \lambda y\, s$ (choosing $y \neq x$ and $y \notin \mathbf{fv}(\overline{u})$), $\epsilon' = \lambda y\, \sigma'$ and $s \Rightarrow_{(b_0)} \sigma'$ then, by induction hypothesis, $\partial_x s \cdot \overline{u} \Rightarrow_{(b_0+b_1)} \partial_x \sigma' \cdot \overline{v'}$. We obtain: $\partial_x e \cdot \overline{u} = \lambda y\, (\partial_x s \cdot \overline{u}) \Rightarrow_{(b_0+b_1)} \lambda y \left(\partial_x \sigma' \cdot \overline{v'}\right) = \partial_x \epsilon' \cdot \overline{v'}$.

If $e = \langle s \rangle\, \overline{t}$, $\epsilon' = \langle \sigma' \rangle\, \overline{\tau'}$, $s \Rightarrow_{(b_0)} \sigma'$ and $\overline{t} \Rightarrow_{(b_0)} \overline{\tau'}$ then, by induction hypothesis, $\partial_x s \cdot \overline{u}_I \Rightarrow_{(b_0+b_1)} \partial_x \sigma' \cdot \overline{v'}_I$ and $\partial_x \overline{t} \cdot \overline{u}_I \Rightarrow_{(b_0+b_1)} \partial_x \overline{\tau'} \cdot \overline{v'}_I$, for all $I \subseteq \{1, \ldots, n\}$. We obtain:

$$\partial_x e \cdot \overline{u} = \sum_{\substack{(I,J) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} \langle \partial_x s \cdot \overline{u}_I \rangle\, \partial_x \overline{t} \cdot \overline{u}_J \Rightarrow_{(b_0+b_1)} \sum_{\substack{(I,J) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} \left\langle \partial_x \sigma' \cdot \overline{v'}_I \right\rangle \partial_x \overline{\tau'} \cdot \overline{v'}_J = \partial_x \epsilon' \cdot \overline{v'}.$$

If $e = [s_1, \ldots, s_r]$, $\epsilon' = [\sigma'_1, \ldots, \sigma'_r]$ and $s_i \Rightarrow_{(b_0)} \sigma'_i$ for all $i \in \{1, \ldots, r\}$ then, by induction hypothesis, $\partial_x s_i \cdot \overline{u}_I \Rightarrow_{(b_0+b_1)} \partial_x \sigma'_i \cdot \overline{v'}_I$ for all $i \in \{1, \ldots, r\}$ and all $I \subseteq \{1, \ldots, n\}$. We obtain:

$$\begin{aligned}\partial_x e \cdot \overline{u} &= \sum_{\substack{(I_1,\ldots,I_r) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} [\partial_x s_1 \cdot \overline{u}_{I_1}, \ldots, \partial_x s_r \cdot \overline{u}_{I_r}] \\ &\Rightarrow_{(b_0+b_1)} \sum_{\substack{(I_1,\ldots,I_r) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} \left[\partial_x \sigma'_1 \cdot \overline{v'}_{I_1}, \ldots, \partial_x \sigma'_r \cdot \overline{v'}_{I_r}\right] = \partial_x \epsilon' \cdot \overline{v'}.\end{aligned}$$

If $e = \langle \lambda y\, s \rangle\, \overline{t}$ (choosing $y \neq x$ and $y \notin \mathbf{fv}\left(\overline{t}\right) \cup \mathbf{fv}(\overline{u})$), $\epsilon' = \partial_y \sigma' \cdot \overline{\tau'}$, $b_0 \geq 1$, $s \Rightarrow_{(b_0-1)} \sigma'$ and $\overline{t} \Rightarrow_{(b_0-1)} \overline{\tau'}$ then, by induction hypothesis, $\partial_x s \cdot \overline{u}_I \Rightarrow_{(b_0+b_1-1)} \partial_x \sigma' \cdot \overline{v'}_I$ and $\partial_x \overline{t} \cdot \overline{u}_I \Rightarrow_{(b_0+b_1-1)} \partial_x \overline{\tau'} \cdot \overline{v'}_I$, for all $I \subseteq \{1, \ldots, n\}$. We obtain:

$$\begin{aligned}\partial_x e \cdot \overline{u} &= \sum_{\substack{(I,J) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} \langle \lambda y\, \partial_x s \cdot \overline{u}_I \rangle\, \partial_x \overline{t} \cdot \overline{u}_J \\ &\Rightarrow_{(b_0+b_1)} \sum_{\substack{(I,J) \text{ partition} \\ \text{of } \{1,\ldots,n\}}} \partial_y \left(\partial_x \sigma' \cdot \overline{v'}_I\right) \cdot \left(\partial_x \overline{\tau'} \cdot \overline{v'}_J\right) = \partial_x \left(\partial_y \sigma' \cdot \overline{\tau'}\right) \cdot \overline{v'} = \partial_x \epsilon' \cdot \overline{v'}\end{aligned}$$

using Lemma 2.3.9. □

Lemma 2.6.14. *Let K be a finite set, and assume $\epsilon \Rightarrow_{(b_k)} \epsilon'_k$ for all $k \in K$. Then, setting $b = \sum_{k \in K} b_k$, there is ϵ'' such that $\epsilon'_k \Rightarrow_{(2^{b_k} b)} \epsilon''$ for all $k \in K$.*

Proof. By the linearity of the definition of reduction on finite sums, it is sufficient to address the case of $\epsilon = e \in (!)\Delta$. The proof is then by induction on the family of reductions $e \Rightarrow_{(b_k)} \epsilon'_k$.

If $e = x = \epsilon'_k$ for all $k \in K$, then we set $\epsilon'' = x$.

If $e = \lambda x\, s$, and $\epsilon'_k = \lambda x\, \sigma'_k$ with $s \Rightarrow_{(b_k)} \sigma'_k$ for all $k \in K$ then, by induction hypothesis, we have σ'' such that $\sigma'_k \Rightarrow_{(2^{b_k} b)} \sigma''$ for all $k \in K$, and then we set $\epsilon'' = \lambda x\, \sigma''$.

If $e = [s_1, \ldots, s_r]$ and $\epsilon'_k = \left[\sigma'_{1,k}, \ldots, \sigma'_{r,k}\right]$ with $s_j \Rightarrow_{(b_k)} \sigma'_{j,k}$ for all $j \in \{1, \ldots, r\}$ and $k \in K$ then, by induction hypothesis, we have σ''_j such that $\sigma'_{j,k} \Rightarrow_{(2^{b_k} b)} \sigma''_j$ for all $j \in \{1, \ldots, r\}$ and $k \in K$, and then we set $\epsilon'' = [\sigma''_1, \ldots, \sigma''_r]$.

Finally, assume $K = K_0 + K_1$, $e = \langle \lambda x\, s \rangle\, \overline{t}$ and:

– for all $k \in K_0$, $\epsilon'_k = \langle \lambda x\, \sigma'_k \rangle\, \overline{\tau'}_k$ with $s \Rightarrow_{(b_k)} \sigma'_k$ and $\bar{t} \Rightarrow_{(b_k)} \overline{\tau'}_k$;

– for all $k \in K_1$, $b_k \geq 1$ and $\epsilon'_k = \partial_x \sigma'_k \cdot \overline{\tau'}_k$ with $s \Rightarrow_{(b_k - 1)} \sigma'_k$ and $\bar{t} \Rightarrow_{(b_k-1)} \overline{\tau'}_k$.

Write $b' = b - \# K_1$. By induction hypothesis, there are σ'' and $\overline{\tau''}$ such that, for all $k \in K_0$, $\sigma'_k \Rightarrow_{(2^{b_k} b')} \sigma''$ and $\overline{\tau'}_k \Rightarrow_{(2^{b_k} b')} \overline{\tau''}$, and for all $k \in K_1$, $\sigma'_k \Rightarrow_{\left(2^{(b_k-1)} b'\right)} \sigma''$ and $\overline{\tau'}_k \Rightarrow_{\left(2^{(b_k-1)} b'\right)} \overline{\tau''}$.

If $K_1 = \emptyset$ then $b = b'$ and we set $\epsilon'' = \langle \lambda x\, \sigma'' \rangle\, \overline{\tau''}$: we obtain $\epsilon'_k \Rightarrow_{(2^{b_k} b)} \epsilon''$, for all $k \in K = K_0$.

Otherwise, $b > b'$ and we set $\epsilon'' = \partial_x \sigma'' \cdot \overline{\tau''}$ so that:

– for all $k \in K_0$, $\epsilon'_k = \langle \lambda x\, \sigma'_k \rangle\, \overline{\tau'}_k \Rightarrow_{(2^{b_k} b' + 1)} \sigma'_k$ with $2^{b_k} b' + 1 \leq 2^{b_k} b$;

– for all $k \in K_1$, by the previous lemma, $\epsilon'_k = \partial_x \sigma'_k \cdot \overline{\tau'}_k \Rightarrow_{(2^{b_k} b')} \sigma'_k$ and $2^{b_k} b' < 2^{b_k} b$.

□

We already know $\widetilde{\Rightarrow_\partial}$ is not confluent, and the counter examples we provided actually show that no single $\widetilde{\Rightarrow_{(b)}}$ is confluent either. Setting [19]

$$\widetilde{\Rightarrow_{(\partial)}} := \left(\bigcup_{b \in \mathbf{N}} \widetilde{\Rightarrow_{(b)}} \right) \subseteq \mathbf{S}^{(!)\Delta} \times \mathbf{S}^{(!)\Delta}$$

however, we will obtain a strongly confluent reduction relation, under the assumption that $\mathbf{S}$ has the following additive splitting property: [20]

Definition 2.6.15. We say $\mathbf{S}$ has the *additive splitting property* if: whenever $a_1 + a_2 = b_1 + b_2 \in \mathbf{S}$, there exists $c_{1,1}, c_{1,2}, c_{2,1}, c_{2,2} \in \mathbf{S}$ such that $a_i = c_{i,1} + c_{i,2}$ and $b_j = c_{1,j} + c_{2,j}$ for $i, j \in \{1, 2\}$.

This property is satisfied by any ring, but also by the usual semirings of non-negative numbers ($\mathbf{N}$, $\mathbf{Q}^+$, *etc.*) as well as booleans. We will in fact rely on the following generalization of the property to finite families of finite sums of any size:

Lemma 2.6.16. *Assume $\mathbf{S}$ has the additive splitting property. Let $a \in \mathbf{S}$, $J_1, \ldots, J_n$ be finite sets and, for all $i \in \{1, \ldots, n\}$, let $(b_{i,j})_{j \in J_i} \in \mathbf{S}^{J_i}$ be a family such that $a = \sum_{j \in J_i} b_{i,j}$. Write $J = J_1 \times \cdots \times J_n$ and, for all $i \in \{1, \ldots, n\}$, write $J'_i = J_1 \times \cdots \times J_{i-1} \times J_{i+1} \times \cdots \times J_n$. Whenever $\vec{\jmath}' = (j_1, \ldots, j_{i-1}, j_{i+1}, \ldots, j_n) \in J'_i$ and $j_i \in J_i$, write $\vec{\jmath}' \cdot_i j_i = (j_1, \ldots, j_n) \in J$. Then there exists a family $\left(c_{\vec{\jmath}}\right) \in \mathbf{S}^J$ such that, for all $i \in \{1, \ldots, n\}$ and all $j \in J_i$, $b_{i,j} = \sum_{\vec{\jmath}' \in J'_i} c_{\vec{\jmath}' \cdot_i j}$.*

Proof. By induction on n, and then on $\# J_n$ for $n > 0$, using the binary additive splitting property to enable the induction. □

19. Our notation is somehow abusive as $\widetilde{\Rightarrow_{(\partial)}}$ is not of the form described in Definition 2.5.9: there should not be any ambiguity as we have not defined any relation $\Rightarrow_{(\partial)}$. Similarly, we may also write $\widetilde{\Rightarrow_{(\partial)}}_{\mathfrak{C}}$ for $\bigcup_{b \in \mathbf{N}} \widetilde{\Rightarrow_{(b)}}_{\mathfrak{C}}$ in the following.

20. The additive splitting property was previously used by Carraro, Ehrhard and Salibra [CES10; Car11] in their study of linear logic exponentials with infinite multiplicities. There is no clear connection between that work and our present contributions, though.

Lemma 2.6.17. *Assume* $\mathbf{S}$ *has the additive splitting property and fix a* $\Rightarrow_\partial$*-reduction structure* $\mathfrak{E}$*. For all finite set* K *and all reductions* $\epsilon \mathrel{\widetilde{\Rightarrow_{(\partial)}}}_{\mathfrak{E}} \epsilon'_k$ *for* $k \in K$*, there is* ϵ'' *such that* $\epsilon'_k \mathrel{\widetilde{\Rightarrow_{(\partial)}}}_{\mathfrak{E}} \epsilon''$ *for all* $k \in K$*.*

Proof. For all $k \in K$, there are $b_k \in \mathbf{N}$, a resource support $\mathcal{E}_k \in \mathfrak{E}$, a set I_k of indices, a family $(a_{k,i})_{i\in I_k}$ of scalars, and summable families $(e_{k,i})_{i\in I_k} \in \mathcal{E}_k^{I_k}$ and $\left(\epsilon'_{k,i}\right)_{i\in I_k} \in \mathbf{N}[(!)\Delta]^{I_k}$ such that $\epsilon = \sum_{i\in I_k} a_{k,i}.e_{k,i}$, $\epsilon'_k = \sum_{i\in I_k} a_{k,i}.\epsilon'_{k,i}$ and $e_{k,i} \Rightarrow_{(b_k)} \epsilon'_{k,i}$ for all $i \in I_k$.

Write $\mathcal{E} = \{e_{k,i} \ ; \ k \in K,\ i \in I_k\}$: since $\mathcal{E} \subseteq \bigcup_{k\in K} \mathcal{E}_k$ and $\mathfrak{E}$ is a resource structure, we have $\mathcal{E} \in \mathfrak{E}$. Write $\mathcal{E}' = \bigcup\left\{\left|\epsilon'_{k,i}\right| \ ; \ k \in K,\ i \in I_k\right\}$: since $\mathfrak{E}$ is a reduction structure, we also have $\mathcal{E}' \in \mathfrak{E}$.

Now fix $e \in (!)\Delta$ and write $a = \epsilon_e$. For all $k \in K$, the set $I_{e,k} = \{i_k \in I_k \ ; \ e_{k,i_k} = e\}$ is finite, and then $\sum_{i_k\in I_{e,k}} a_{k,i_k} = a$. Write $I_e = \prod_{k\in K} I_{e,k}$ and, for all $k \in K$, $K'_k = K \setminus \{k\}$ and $I'_{e,k} = \prod_{l\in K'_k} I_{e,l}$. If $\overrightarrow{\imath} = (i_l)_{l\in K'_k} \in I'_{e,k}$ and $i_k \in I_{e,k}$, write $\overrightarrow{\imath} \cdot_k i_k = (i_k)_{k\in K} \in I_e$. By Lemma 2.6.16, we obtain a family of scalars $\left(a'_{e,\overrightarrow{\imath}}\right)_{\overrightarrow{\imath}\in I_e}$ such that, for all $k \in K$ and all $i_k \in I_{e,k}$, $a_{k,i_k} = \sum_{\overrightarrow{\imath}\in I'_{e,k}} a'_{e,\overrightarrow{\imath}\cdot_k i_k}$. Moreover, $a = \sum_{\overrightarrow{\imath}\in I_e} a'_{e,\overrightarrow{\imath}}$.

Since each I_e is finite, the family $(e)_{e\in(!)\Delta,\overrightarrow{\imath}\in I_e}$ is summable. Moreover, if we fix $k \in K$ and $i_k \in I_k$, there are finitely many $e \in (!)\Delta$ and $\overrightarrow{\imath} \in I'_{e,k}$ such that $\overrightarrow{\imath} \cdot_k i_k \in I_e$: indeed in this case $e = e_{k,i_k}$. Since $\left(\epsilon'_{k,i_k}\right)_{i_k\in I_k}$ is summable too, it follows that $\left(\epsilon'_{k,i_k}\right)_{e\in(!)\Delta,\overrightarrow{\imath}\in I_e}$ is summable. By associativity, we obtain

$$\sum_{\substack{e\in(!)\Delta\\ \overrightarrow{\imath}\in I_e}} a'_{e,\overrightarrow{\imath}}.e = \sum_{e\in(!)\Delta}\left(\sum_{\overrightarrow{\imath}\in I_e} a'_{e,\overrightarrow{\imath}}\right) e = \epsilon$$

and

$$\sum_{\substack{e\in(!)\Delta\\ \overrightarrow{\imath}\in I_e}} a'_{e,\overrightarrow{\imath}}.\epsilon'_{k,i_k} = \sum_{i_k\in I_k}\left(\sum_{\overrightarrow{\imath}\in I'_{e_{k,i_k},k}} a'_{e_{k,i_k},\overrightarrow{\imath}\cdot_k i_k}\right)\epsilon'_{k,i_k} = \epsilon'_k$$

for all $k \in K$.

Write $b = \sum_{k\in K} b_k$. For all $e \in (!)\Delta$ and all $\overrightarrow{\imath} = (i_k)_{k\in K} \in I_e$, we have $e \Rightarrow_{(b_k)} \epsilon'_{k,i_k}$ for all $k \in K$ hence Lemma 2.6.14 gives $\epsilon''_{e,\overrightarrow{\imath}} \in \mathbf{N}[(!)\Delta]$ such that $\epsilon'_{k,i_k} \Rightarrow_{(2^{b_k}b)} \epsilon''_{e,\overrightarrow{\imath}}$ for all $k \in K$. Moreover, for all $k \in K$ and $e'' \in (!)\Delta$, if $e'' \in \left|\epsilon''_{e,\overrightarrow{\imath}}\right|$ then there is $e' \in \left|\epsilon'_{k,i_k}\right|$ such that $e' \ggcurly_{(2^{b_k}b)} e''$, and then $e \ggcurly_{(b_k)} e'$: it follows that $\mathbf{s}(e) \leq 4^{b_k+2^{b_k}b}\mathbf{s}(e'')$ and $\mathbf{fv}(e) = \mathbf{fv}(e'')$. Since each I_e is finite, there are finitely many pairs $(e, \overrightarrow{\imath}) \in \sum_{e\in(!)\Delta} I_e$ such that $e'' \in \left|\epsilon''_{e,\overrightarrow{\imath}}\right|$. Hence the family $\left(\epsilon''_{e,\overrightarrow{\imath}}\right)_{e\in(!)\Delta,\overrightarrow{\imath}\in I_e}$ is summable. Recall moreover that $\epsilon'_{k,i_k} \in \mathbf{N}[\mathcal{E}']$ for all $k \in K$ and $i_k \in I_k$: we obtain

$$\epsilon'_k \mathrel{\widetilde{\Rightarrow_{(2^{b_k}b)}}}_{\mathcal{E}'} \sum_{\substack{e\in(!)\Delta\\ \overrightarrow{\imath}\in I_e}} a'_{e,\overrightarrow{\imath}}.\epsilon''_{e,\overrightarrow{\imath}}$$

for all $k \in K$, which concludes the proof. $\square$

2.6.3 Bounded depth of substitution

In the previous subsection, we relied on the additive splitting property to establish the confluence of $\widetilde{\Rightarrow_{(\partial)}}$: this is because there is no maximal way to $\widetilde{\Rightarrow_{(b)}}$-reduce a resource vector, hence we must track precisely the different redexes that are fired in each reduction of a critical pair.

We can get rid of this hypothesis by considering a more uniform bound on reductions. A first intuition would be to bound the depth at which redexes are fired, but as with $\Rightarrow_{(m|k)}$ this boundedness condition is not preserved in residuals: rather, we have to bound the depth at which variables are substituted. First recall from Definition 2.3.2 that $\mathbf{md}_x(s) = \max \mathbf{d}_x(s)$ is the maximum depth of an occurrence of x in s. Then:

Definition 2.6.18. We define a family of relations $\left(\Rightarrow_{\lfloor d \rfloor}\right)_{d \in \mathbf{N}}$ inductively as follows:

- $e \Rightarrow_{\lfloor 0 \rfloor} e$ for all $e \in (!)\Delta$;
- $x \Rightarrow_{\lfloor d+1 \rfloor} x$ for all $x \in \mathcal{V}$;
- $\lambda x\, s \Rightarrow_{\lfloor d+1 \rfloor} \lambda x\, \sigma'$ if $s \Rightarrow_{\lfloor d \rfloor} \sigma'$;
- $\langle s \rangle\, \bar{t} \Rightarrow_{\lfloor d+1 \rfloor} \langle \sigma' \rangle\, \overline{\tau'}$ if $s \Rightarrow_{\lfloor d+1 \rfloor} \sigma'$ and $\bar{t} \Rightarrow_{\lfloor d \rfloor} \overline{\tau'}$;
- $[s_1, \dots, s_r] \Rightarrow_{\lfloor d+1 \rfloor} [\sigma'_1, \dots, \sigma'_r]$ if $s_i \Rightarrow_{\lfloor d+1 \rfloor} \sigma'_i$ for all $i \in \{1, \dots, r\}$;
- $\langle \lambda x\, s \rangle\, \bar{t} \Rightarrow_{\lfloor d+1 \rfloor} \partial_x \sigma' \cdot \overline{\tau'}$ if $\mathbf{md}_x(s) \le d$, $s \Rightarrow_{\lfloor d \rfloor} \sigma'$ and $\bar{t} \Rightarrow_{\lfloor d \rfloor} \overline{\tau'}$.

It should be clear that if $e \Rightarrow_{\lfloor d \rfloor} \epsilon'$ then $e \Rightarrow_{(d)} \epsilon'$, and moreover $e \Rightarrow_{\lfloor d' \rfloor} \epsilon'$ for all $d' \ge d$. We also have $e \Rightarrow_{\lfloor \mathbf{h}(e) \rfloor} \epsilon'$ as soon as $e \Rightarrow_\partial \epsilon'$.

Definition 2.6.19. For all $e \in (!)\Delta$ we define the *full parallel reduct* $\mathsf{F}_{\lfloor d \rfloor}(e)$ *at substitution depth* d of e by induction on the pair (d, e) as follows:

$$
\begin{aligned}
\mathsf{F}_{\lfloor 0 \rfloor}(e) &:= e \\
\mathsf{F}_{\lfloor d+1 \rfloor}(x) &:= x \\
\mathsf{F}_{\lfloor d+1 \rfloor}(\lambda x\, s) &:= \lambda x\, \mathsf{F}_{\lfloor d \rfloor}(s) \\
\mathsf{F}_{\lfloor d+1 \rfloor}\left(\langle \lambda x\, s \rangle\, \bar{t}\right) &:= \partial_x \mathsf{F}_{\lfloor d \rfloor}(s) \cdot \mathsf{F}_{\lfloor d \rfloor}\left(\bar{t}\right) && \text{(if } \mathbf{md}_x(s) \le d\text{)} \\
\mathsf{F}_{\lfloor d+1 \rfloor}\left(\langle s \rangle\, \bar{t}\right) &:= \left\langle \mathsf{F}_{\lfloor d+1 \rfloor}(s) \right\rangle \mathsf{F}_{\lfloor d \rfloor}\left(\bar{t}\right) && \text{(in the other cases)} \\
\mathsf{F}_{\lfloor d+1 \rfloor}([s_1, \dots, s_n]) &:= \left[\mathsf{F}_{\lfloor d+1 \rfloor}(s_1), \dots, \mathsf{F}_{\lfloor d+1 \rfloor}(s_n)\right]
\end{aligned}
$$

Then if $\epsilon = \sum_{i=1}^n e_i \in \mathbf{N}[(!)\Delta]$, we set $\mathsf{F}_{\lfloor d \rfloor}(\epsilon) := \sum_{i=1}^n \mathsf{F}_{\lfloor d \rfloor}(e_i)$.

Lemma 2.6.20. *For all* $e \in (!)\Delta$, $e \Rightarrow_{\lfloor d \rfloor} \mathsf{F}_{\lfloor d \rfloor}(e)$.

Proof. By a straightforward induction on d then on e. $\square$

It follows that $e \Rightarrow_{(d)} \mathsf{F}_{\lfloor d \rfloor}(e)$, hence if $e' \in \left|\mathsf{F}_{\lfloor d \rfloor}(e)\right|$ then $\mathbf{s}(e) \le 4^d \mathbf{s}(e')$. In particular $\mathsf{F}_{\lfloor d \rfloor}$ defines a linear-continuous function on $\mathbf{S}^{(!)\Delta}$.

Lemma 2.6.21. *If* $e \Rightarrow_{\lfloor d_0 \rfloor} \epsilon'$, $\overline{u} \Rightarrow_{\lfloor d_1 \rfloor} \overline{\upsilon}'$ *and* $d \geq \max\left(\{d_0\} \cup \{d_x + d_1 - 1 \; ; \; d_x \in \mathbf{d}_x(e)\}\right)$ *then* $\partial_x e \cdot \overline{u} \Rightarrow_{\lfloor d \rfloor} \partial_x \epsilon' \cdot \overline{\upsilon}'$.

Proof. Write $n = \#\overline{u}$, $\overline{u} = [u_1, \ldots, u_n]$ and $\overline{\upsilon}' = [\upsilon'_1, \ldots, \upsilon'_n]$ so that $u_i \Rightarrow_{\lfloor d_1 \rfloor} \upsilon'_i$ for $i \in \{1, \ldots, n\}$.

The proof is by induction on the reduction $e \Rightarrow_{\lfloor d_0 \rfloor} \epsilon'$. We treat the cases $d_0 = 0$ and $d_0 > 0$ uniformly by a further induction on e, setting $d'_0 = \max\{0, d_0 - 1\}$.

If $d_0 = d'_0 + 1$, $e = \langle \lambda y \, s \rangle \, \overline{t}$ and $\epsilon' = \partial_y \sigma' \cdot \overline{\tau}'$ with $y \notin \{x\} \cup \mathbf{fv}(\overline{t}) \cup \mathbf{fv}(\overline{u})$, $\mathbf{md}_y(s) \leq d'_0$, $s \Rightarrow_{\lfloor d'_0 \rfloor} \sigma'$ and $\overline{t} \Rightarrow_{\lfloor d'_0 \rfloor} \overline{\tau}'$, then we have

$$\partial_x e \cdot \overline{u} = \sum_{(I,J) \text{ partition of } \{1,\ldots,n\}} \langle \lambda y \, (\partial_x s \cdot \overline{u}_I) \rangle \, \partial_x \overline{t} \cdot \overline{u}_J$$

and

$$\partial_x \epsilon' \cdot \overline{\upsilon}' = \sum_{(I,J) \text{ partition of } \{1,\ldots,n\}} \partial_y \big(\partial_x s \cdot \overline{\upsilon}'_I\big) \cdot \big(\partial_x \overline{t} \cdot \overline{\upsilon}'_J\big).$$

Observe that $d > 0$ and $d - 1 \geq \max\{d'_0\} \cup \big\{d'_x + d_1 - 1 \; ; \; d'_x \in \mathbf{d}_x(s) \cup \mathbf{d}_x(\overline{t})\big\}$. By induction hypothesis, we obtain $\partial_x s \cdot \overline{u}_I \Rightarrow_{\lfloor d-1 \rfloor} \partial_x \sigma' \cdot \overline{\upsilon}'_I$ and $\partial_x \overline{t} \cdot \overline{u}_J \Rightarrow_{\lfloor d-1 \rfloor} \partial_x \overline{\tau}' \cdot \overline{\upsilon}'_J$, and we conclude since $\mathbf{md}_y(\partial_x s \cdot \overline{u}_I) = \mathbf{md}_y(s) \leq d'_0 \leq d - 1$.

If $e = y = \epsilon'$, with $y \neq x$, then $\partial_x e \cdot \overline{u} = \partial_x \epsilon' \cdot \overline{\upsilon}' = y$ and we conclude directly by the definition of $\Rightarrow_{\lfloor d \rfloor}$.

If $e = x = \epsilon'$, then $\mathbf{d}_x(e) = \{1\}$ hence $d \geq d_1$ and we conclude since $\partial_x e \cdot \overline{u} = \overline{u}$, $\partial_x \epsilon' \cdot \overline{\upsilon}' = \overline{\upsilon}'$ and $\overline{u} \Rightarrow_{\lfloor d_1 \rfloor} \overline{\upsilon}'$.

If $e = \lambda y \, s$ and $\epsilon' = \lambda y \, \sigma'$ with $y \notin \{x\} \cup \mathbf{fv}(\overline{u})$ and $s \Rightarrow_{\lfloor d'_0 \rfloor} \sigma'$, then write $d' = \max\{d'_0\} \cup \{d'_x + d_1 - 1 \; ; \; d'_x \in \mathbf{d}_x(s)\}$. By induction hypothesis, we obtain $\partial_x s \cdot \overline{u} \Rightarrow_{\lfloor d' \rfloor} \partial_x \sigma' \cdot \overline{\upsilon}'$. Observe that either $d = d' + 1$ or $d = d' = 0$ (in that latter case, $\partial_x s \cdot \overline{u} = \partial_x \sigma' \cdot \overline{\upsilon}'$), and then we conclude since $\partial_x e \cdot \overline{u} = \lambda y \, (\partial_x s \cdot \overline{u})$ and $\partial_x \epsilon' \cdot \overline{\upsilon}' = \lambda y \, (\partial_x \sigma' \cdot \overline{\upsilon}')$.

If $e = \langle s \rangle \, \overline{t}$ and $\epsilon' = \langle \sigma' \rangle \, \overline{\tau}'$, with $s \Rightarrow_{\lfloor d_0 \rfloor} \sigma'$ and $\overline{t} \Rightarrow_{\lfloor d'_0 \rfloor} \overline{\tau}'$, then we have

$$\partial_x e \cdot \overline{u} = \sum_{(I,J) \text{ partition of } \{1,\ldots,n\}} \langle \partial_x s \cdot \overline{u}_I \rangle \, \partial_x \overline{t} \cdot \overline{u}_J$$

and

$$\partial_x \epsilon' \cdot \overline{\upsilon}' = \sum_{(I,J) \text{ partition of } \{1,\ldots,n\}} \big\langle \partial_x s \cdot \overline{\upsilon}'_I \big\rangle \, \big(\partial_x \overline{t} \cdot \overline{\upsilon}'_J\big).$$

Write $d' = \max\{d'_0\} \cup \big\{d'_x + d_1 - 1 \; ; \; d'_x \in \mathbf{d}_x(\overline{t})\big\}$. By induction hypothesis, we obtain $\partial_x s \cdot \overline{u}_I \Rightarrow_{\lfloor d \rfloor} \partial_x \sigma' \cdot \overline{\upsilon}'_I$ and $\partial_x \overline{t} \cdot \overline{u}_J \Rightarrow_{\lfloor d' \rfloor} \partial_x \overline{\tau}' \cdot \overline{\upsilon}'_J$. Then we conclude observing that $d = d' + 1$ or $d = d' = 0$ (in that latter case, $\partial_x s \cdot \overline{u}_I = \partial_x \sigma' \cdot \overline{\upsilon}'_I$ and $\partial_x \overline{t} \cdot \overline{u}_J = \partial_x \overline{\tau}' \cdot \overline{u}'_J$).

If $e = [s_1, \ldots, s_k]$ and $\epsilon' = [\sigma'_1, \ldots, \sigma'_k]$, with $s_i \Rightarrow_{\lfloor d_0 \rfloor} \sigma'_i$ for $i \in \{1, \ldots, k\}$, then we have

$$\partial_x e \cdot \overline{u} = \sum_{(I_1,\ldots,I_k) \text{ partition of } \{1,\ldots,n\}} [\partial_x s_1 \cdot \overline{u}_{I_1} \ldots, \partial_x s_k \cdot \overline{u}_{I_k}]$$

and

$$\partial_x \epsilon' \cdot \overline{v}' = \sum_{(I_1,\ldots,I_k) \text{ partition of } \{1,\ldots,n\}} \left[\partial_x \sigma'_1 \cdot \overline{v'}_{I_1} \ldots, \partial_x \sigma'_k \cdot \overline{v'}_{I_k}\right].$$

By induction hypothesis, we obtain

$$[\partial_x s_1 \cdot \overline{u}_{I_1} \ldots, \partial_x s_k \cdot \overline{u}_{I_k}] \Rightarrow_{\lfloor d \rfloor} \left[\partial_x \sigma'_1 \cdot \overline{v'}_{I_1} \ldots, \partial_x \sigma'_k \cdot \overline{v'}_{I_k}\right]$$

for all partition $(I_1, \ldots, I_k)$ of $\{1, \ldots, n\}$ and we conclude. $\square$

Lemma 2.6.22. *If $e \Rightarrow_{\lfloor d \rfloor} \epsilon'$ and $e' \in |\epsilon'|$ then $\mathbf{md}_x(e') \leq 2^d \max\{d, \mathbf{md}_x(e)\}$.*

Proof. By induction on the reduction $e \Rightarrow_{\lfloor d \rfloor} \epsilon'$.

If $d = 0$, then $e' = \epsilon' = e$ and the result is trivial. For the other inductive cases, write $d = d' + 1$.

If $e = \langle \lambda y\, s \rangle\, \overline{t}$ and $\epsilon' = \partial_y \sigma' \cdot \overline{\tau}'$ with $\mathbf{md}_y(s) \leq d'$, $s \Rightarrow_{\lfloor d' \rfloor} \sigma'$ and $\overline{t} \Rightarrow_{\lfloor d' \rfloor} \overline{\tau}'$, choosing $y \notin \{x\} \cup \mathbf{fv}(\overline{t})$, then $e' \in \left|\partial_y s' \cdot \overline{t}'\right|$ with $s' \in |\sigma'|$ and $\overline{t}' \in |\overline{\tau}'|$. By induction hypothesis, $\mathbf{md}_z(s') \leq 2^{d'} \max\{d', \mathbf{md}_z(s)\}$ and $\mathbf{md}_z(\overline{t}') \leq 2^{d'} \max\left\{d', \mathbf{md}_z\left(\overline{t}\right)\right\}$ for any $z \in \mathcal{V}$. By Lemma 2.3.7,

$$\begin{aligned}
\mathbf{md}_x(e') &\leq \max\left(\mathbf{d}_x(s') \cup \left\{d'_y + d'_x - 1\, ;\, d'_y \in \mathbf{d}_y(s'),\ d'_x \in \mathbf{d}_x(\overline{t}')\right\}\right) \\
&\leq \max\left\{2^{d'} \max\left\{d', \mathbf{md}_x(s)\right\}, 2^{d'} \max\left\{d', \mathbf{md}_y(s)\right\} + 2^{d'} \max\left\{d', \mathbf{md}_x\left(\overline{t}\right)\right\}\right\} \\
&\leq 2^{d'+1} \max\left\{d', \mathbf{md}_x(s), \mathbf{md}_y(s), \mathbf{md}_x\left(\overline{t}\right)\right\} \\
&\leq 2^d \max\{d, \mathbf{md}_x(e)\}.
\end{aligned}$$

If $e = \lambda y\, s$ and $\epsilon' = \lambda y\, \sigma'$ with $s \Rightarrow_{\lfloor d' \rfloor} \sigma'$, choosing $y \neq x$, then $e' = \lambda x\, s'$ with $s' \in |\sigma'|$. By induction hypothesis, $\mathbf{md}_x(s') \leq 2^{d'} \max\{d', \mathbf{md}_x(s)\}$. Then $\mathbf{md}_x(e') \leq \mathbf{md}_x(s') + 1 \leq 2^{d'} \max\{d', \mathbf{md}_x(s)\} + 1 \leq 2^d \max\{d, \mathbf{md}_x(s)\} \leq 2^d \max\{d, \mathbf{md}_x(e)\}$.

If $e = \langle s \rangle\, \overline{t}$ and $\epsilon' = \langle \sigma' \rangle\, \overline{\tau}'$ with $s \Rightarrow_{\lfloor d \rfloor} \sigma'$ and $\overline{t} \Rightarrow_{\lfloor d' \rfloor} \overline{\tau}'$, then $e' = \langle s' \rangle\, \overline{t}'$ with $s' \in |\sigma'|$ and $\overline{t}' \in |\overline{\tau}'|$. By induction hypothesis, $\mathbf{md}_x(s') \leq 2^d \max\{d, \mathbf{md}_x(s)\}$ and $\mathbf{md}_x(\overline{t}') \leq 2^{d'} \max\left\{d', \mathbf{md}_x\left(\overline{t}\right)\right\}$. Then:

$$\begin{aligned}
\mathbf{md}_x(e') &\leq \max\left\{\mathbf{md}_x(s'), \mathbf{md}_x(\overline{t}') + 1\right\} \\
&\leq \max\left\{2^d \max\{d, \mathbf{md}_x(s)\}, 2^{d'} \max\left\{d', \mathbf{md}_x\left(\overline{t}\right)\right\} + 1\right\} \\
&\leq 2^d \max\left\{d, \mathbf{md}_x(s), \mathbf{md}_x\left(\overline{t}\right)\right\} \\
&\leq 2^d \max\{d, \mathbf{md}_x(e)\}.
\end{aligned}$$

If $e = [s_1, \ldots, s_k]$ and $\epsilon' = [\sigma'_1, \ldots, \sigma'_k]$, with $s_i \Rightarrow_{\lfloor d \rfloor} \sigma'_i$ for all $i \in \{1, \ldots, k\}$, then $e' = [s'_1, \ldots, s'_k]$ with $s'_i \in |\sigma'_i|$ for all $i \in \{1, \ldots, k\}$. By induction hypothesis, for all $i \in \{1, \ldots, k\}$, $\mathbf{md}_x(s'_i) \leq 2^d \max\{d, \mathbf{md}_x(s_i)\}$, hence

$$\begin{aligned}
\mathbf{md}_x(e') &= \max\left\{\mathbf{md}_x(s'_1), \ldots, \mathbf{md}_x(s'_k)\right\} \\
&\leq 2^d \max\{d, \mathbf{md}_x(s_1), \ldots, \mathbf{md}_x(s_k)\} \\
&= 2^d \max\{d, \mathbf{md}_x(e)\}.
\end{aligned}$$

□

Lemma 2.6.23. *If $e \Rightarrow_{\lfloor d\rfloor} \epsilon'$ then $\epsilon' \Rightarrow_{\lfloor 2^d d\rfloor} \mathsf{F}_{\lfloor d\rfloor}(e)$.*

Proof. By induction on the reduction $e \Rightarrow_{\lfloor d\rfloor} \epsilon'$.

If $d = 0$, then $\epsilon' = e$ and the result follows from Lemma 2.6.20. For the other inductive cases, set $d = d' + 1$.

If $e = \langle \lambda x\, s\rangle\, \bar{t}$ and $\epsilon' = \partial_x \sigma' \cdot \bar{\tau}'$ with $\mathbf{md}_x(s) \leq d'$, $s \Rightarrow_{\lfloor d'\rfloor} \sigma'$ and $\bar{t} \Rightarrow_{\lfloor d'\rfloor} \bar{\tau}'$ then by induction hypothesis, we have $\sigma' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(s)$ and $\bar{\tau}' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(\bar{t})$. By the previous lemma, we moreover have $\mathbf{md}_x(\sigma') \leq 2^{d'} \max\{d', \mathbf{md}_x(s)\} = 2^{d'} d'$. It follows that $2^d d \geq 2^{d'} d'$ and $2^d d \geq \mathbf{md}_x(\sigma') + 2^{d'} d' - 1$ hence we can apply Lemma 2.6.21 to obtain $\epsilon' \Rightarrow_{\lfloor 2^d d\rfloor} \partial_x \mathsf{F}_{\lfloor d'\rfloor}(s) \cdot \mathsf{F}_{\lfloor d'\rfloor}(\bar{t}) = \mathsf{F}_{\lfloor d\rfloor}(e)$.

If $e = \lambda y\, s$ and $\epsilon' = \lambda y\, \sigma'$ with $s \Rightarrow_{\lfloor d'\rfloor} \sigma'$, then by induction hypothesis, $\sigma' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(s)$, hence $\epsilon' \Rightarrow_{\lfloor 2^{d'} d'+1\rfloor} \lambda x\, \mathsf{F}_{\lfloor d'\rfloor}(s) = \mathsf{F}_{\lfloor d\rfloor}(e)$ and we conclude since $2^{d'} d' + 1 \leq 2^d d$.

If $e = \langle s\rangle\, \bar{t}$ and $\epsilon' = \langle \sigma'\rangle\, \bar{\tau}'$ with $s \Rightarrow_{\lfloor d\rfloor} \sigma'$ and $\bar{t} \Rightarrow_{\lfloor d'\rfloor} \bar{\tau}'$, there are two subcases:

- If moreover $s = \lambda x\, u$ and $\mathbf{md}_x(u) \leq d'$ then $\sigma' = \lambda x\, \upsilon'$ with $u \Rightarrow_{\lfloor d'\rfloor} \upsilon'$. Then by induction hypothesis, $\upsilon' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(u)$, and $\bar{\tau}' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(\bar{t})$. By the previous lemma, we moreover have $\mathbf{md}_x(\upsilon') \leq 2^{d'} \max\{d', \mathbf{md}_x(u)\} = 2^{d'} d'$, hence $\epsilon' = \langle \lambda x\, \upsilon'\rangle\, \bar{\tau}' \Rightarrow_{\lfloor 2^{d'} d'+1\rfloor} \partial_x \mathsf{F}_{\lfloor d'\rfloor}(u) \cdot \mathsf{F}_{\lfloor d'\rfloor}(\bar{t}) = \mathsf{F}_{\lfloor d\rfloor}(e)$, and we conclude since $2^{d'} d' + 1 \leq 2^d d$.
- Otherwise s is not an abstraction or $s = \lambda x\, u$ with $\mathbf{md}_x(u) > d'$. By induction hypothesis, $\sigma' \Rightarrow_{\lfloor 2^d d\rfloor} \mathsf{F}_{\lfloor d\rfloor}(\sigma')$, and $\bar{\tau}' \Rightarrow_{\lfloor 2^{d'} d'\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(\bar{t})$. Since $2^{d'} d' < 2^d d$, we obtain $\bar{\tau}' \Rightarrow_{\lfloor 2^d d-1\rfloor} \mathsf{F}_{\lfloor d'\rfloor}(\bar{t})$ and then $\epsilon' \Rightarrow_{\lfloor 2^d d\rfloor} \left\langle \mathsf{F}_{\lfloor d\rfloor}(s)\right\rangle \mathsf{F}_{\lfloor d'\rfloor}(\bar{t}) = \mathsf{F}_{\lfloor d\rfloor}(e)$.

If $e = [s_1, \dots, s_k]$ and $\epsilon' = [\sigma'_1, \dots, \sigma'_k]$, with $s_i \Rightarrow_{\lfloor d\rfloor} \sigma'_i$ for all $i \in \{1, \dots, k\}$, then by induction hypothesis, for all $i \in \{1, \dots, k\}$, $\sigma'_i \Rightarrow_{\lfloor 2^d d\rfloor} \mathsf{F}_{\lfloor d\rfloor}(s_i)$ and we conclude directly. □

Lemma 2.6.24. *For all $\Rightarrow_\partial$-reduction structure $\mathfrak{E}$, if $\epsilon \mathrel{\widetilde{\Rightarrow_{\lfloor d\rfloor}}}_{\mathfrak{E}} \epsilon'$ then $\epsilon' \mathrel{\widetilde{\Rightarrow_{\lfloor 2^d d\rfloor}}}_{\mathfrak{E}} \mathsf{F}_{\lfloor d\rfloor}(\epsilon')$.*

Proof. Assume there is $\mathcal{E} \in \mathfrak{E}$, summable families $(e_i)_{i\in I} \in \mathcal{E}^I$ and $(\epsilon'_i)_{i\in I} \mathbf{N}[(!)\Delta]^I$, and a family of scalars $(a_i)_{i\in I}$ such that $\epsilon = \sum_{i\in I} a_i.e_i$, $\epsilon' = \sum_{i\in I} a_i.\epsilon'_i$ and $e_i \Rightarrow_{\lfloor d\rfloor} \epsilon'_i$ for all $i \in I$. Write $\mathcal{E}' = \bigcup_{i\in I} |\epsilon'_i|$: since $\mathfrak{E}$ is a reduction structure, we obtain $\mathcal{E}' \in \mathfrak{E}$. The family $\left(\mathsf{F}_{\lfloor d\rfloor}(e_i)\right)_{i\in I}$ is summable, and by the previous lemma, $\epsilon'_i \Rightarrow_{\lfloor 2^d d\rfloor} \mathsf{F}_{\lfloor d\rfloor}(e_i)$ for all $i \in I$. We conclude that $\epsilon' \mathrel{\widetilde{\Rightarrow_{\lfloor 2^d d\rfloor}}} \sum_{i\in I} a_i.\mathsf{F}_{\lfloor d\rfloor}(e_i) = \mathsf{F}_{\lfloor d\rfloor}(\epsilon)$. □

Similarly to $\widetilde{\Rightarrow_{(\partial)}}$, we set

$$\widetilde{\Rightarrow_{\lfloor\partial\rfloor}} := \bigcup_{d\in\mathbf{N}} \widetilde{\Rightarrow_{\lfloor d\rfloor}}$$

and we obtain:

Corollary 2.6.25. *For all $\Rightarrow_\partial$-reduction structure $\mathfrak{E}$ and all $\epsilon, \epsilon'_1, \dots, \epsilon'_n \in \mathbf{S}^{(!)\Delta}$ such that $\epsilon \mathrel{\widetilde{\Rightarrow_{\lfloor\partial\rfloor}}}_{\mathfrak{E}} \epsilon'_i$ for $i \in \{1, \dots, n\}$, there exists $d \in \mathbf{N}$ such that $\epsilon'_i \mathrel{\widetilde{\Rightarrow_{\lfloor\partial\rfloor}}}_{\mathfrak{E}} \mathsf{F}_{\lfloor d\rfloor}(\epsilon)$ for $i \in \{1, \dots, n\}$.*

2.6.4 Parallel reduction of resource vectors of bounded height

Recall that we have $e \Rightarrow_\partial \epsilon'$ iff $e \Rightarrow_{(\mathbf{h}(e)|\mathbf{h}(e))} \epsilon'$ iff $e \Rightarrow_{(\mathbf{h}(e))} \epsilon'$ iff $e \Rightarrow_{\lfloor \mathbf{h}(e) \rfloor} \epsilon'$.

Definition 2.6.26. We say a resource vector $\epsilon \in \mathbf{S}^{(!)\Delta}$ is *bounded* if $\{\mathbf{h}(e) \; ; \; e \in |\epsilon|\}$ is finite. We then write $\mathbf{h}(\epsilon) = \max \{\mathbf{h}(e) \; ; \; e \in |\epsilon|\}$.

If $\mathcal{E} \subseteq (!)\Delta$, we also write $\mathbf{h}(\mathcal{E}) := \{\mathbf{h}(e) \; ; \; e \in \mathcal{E}\}$ and then

$$(!)\mathfrak{B} := \{\mathcal{E} \subseteq (!)\Delta \; ; \; \mathbf{h}(\mathcal{E}) \text{ and } \mathbf{fv}(\mathcal{E}) \text{ are finite}\}$$

which is a resource structure (see Definition 2.5.8). Indeed, $(!)\mathfrak{B} \subseteq (!)\mathfrak{F}_{\mathbf{fv}}$, and if we write

$$(!)\Delta_{h,V} := \{e \in (!)\Delta \; ; \; \mathbf{h}(e) \leq h \text{ and } \mathbf{fv}(e) \subseteq V\}$$

for all $h \in \mathbf{N}$ and all $V \subseteq \mathcal{V}$, we have $(!)\mathfrak{B} = \{(!)\Delta_{h,V} \; ; \; h \in \mathbf{N} \text{ and } V \in \mathfrak{P}_f(\mathcal{V})\}^{\perp\perp}$: this is a consequence of a generic *transport lemma* [4]. The semimodule of bounded resource vectors is then $\mathbf{S}\langle(!)\mathfrak{B}\rangle$.

Lemma 2.6.27. *For all $h \in \mathbf{N}$ and $V \in \mathfrak{P}_f(\mathcal{V})$, $(\mathsf{F}(e))_{e \in (!)\Delta_{h,V}}$ is summable. Moreover, for all $\epsilon \in \mathbf{S}\langle(!)\mathfrak{B}\rangle$, we have $|\epsilon| \subseteq (!)\Delta_{\mathbf{h}(\epsilon),\mathbf{fv}(\epsilon)}$ and then, setting $\mathsf{F}(\epsilon) := \sum_{e \in |\epsilon|} \epsilon_e.\mathsf{F}(e)$, we obtain $\epsilon \mathrel{\widetilde{\Rightarrow}}_{\partial|\epsilon|} \mathsf{F}(\epsilon)$.*

Proof. Follows from Lemmas 2.6.20 and 2.6.12 using the fact that, if $\mathbf{h}(e) \leq h$ then $\mathsf{F}(e) = \mathsf{F}_{\lfloor h \rfloor}(e)$. □

If $\mathbf{S}$ is zerosumfree, we have: $\epsilon \mathrel{\widetilde{\Rightarrow}}_\partial \epsilon'$ iff $\epsilon \mathrel{\widetilde{\Rightarrow_{\lfloor \mathbf{h}(\epsilon) \rfloor}}} \epsilon'$ as soon as ϵ is bounded. More generally, without any assumption on $\mathbf{S}$, we have $\epsilon \mathrel{\widetilde{\Rightarrow}}_{\partial(!)\Delta_{h,V}} \epsilon'$ iff $\epsilon \mathrel{\widetilde{\Rightarrow_{\lfloor h \rfloor}}}_{(!)\Delta_{h,V}} \epsilon'$. We can moreover show that bounded vectors are stable under $\mathrel{\widetilde{\Rightarrow}}_{\partial(!)\mathfrak{B}}$:

Lemma 2.6.28. *If $e \ggg_\partial e'$ then $\mathbf{h}(e') \leq 2^{\mathbf{h}(e)}\mathbf{h}(e)$.*

Proof. The proof is by induction on the reduction $e \Rightarrow_\partial \epsilon'$ such that $e' \in |\epsilon'|$, and is very similar to that of Lemma 2.6.22. We detail only the base case.

If $e = \langle \lambda x \, s \rangle \, \bar{t}$ and $\epsilon' = \partial_x \sigma' \cdot \overline{\tau}'$ with $s \Rightarrow_\partial \sigma'$ and $\bar{t} \Rightarrow_\partial \overline{\tau}'$. Then $e' \in \left|\partial_x s' \cdot \overline{t'}\right|$ with $s' \in |\sigma'|$ and $\overline{t'} \in |\overline{\tau}'|$. By induction, $\mathbf{h}(s') \leq 2^{\mathbf{h}(s)}\mathbf{h}(s)$ and $\mathbf{h}(\overline{t'}) \leq 2^{\mathbf{h}(\bar{t})}\mathbf{h}(\bar{t})$. By Lemma 2.3.6,

$$\begin{aligned}
\mathbf{h}(e') &\leq \mathbf{h}(s') + \mathbf{h}(\overline{t'}) \\
&\leq 2^{\mathbf{h}(s)}\mathbf{h}(s) + 2^{\mathbf{h}(\bar{t})}\mathbf{h}(\bar{t}) \\
&\leq 2 \times 2^{\max\{\mathbf{h}(s),\mathbf{h}(\bar{t})\}} \max\{\mathbf{h}(s), \mathbf{h}(\bar{t})\} \\
&< 2^{\max\{\mathbf{h}(s),\mathbf{h}(\bar{t})\}+1}\left(\max\{\mathbf{h}(s), \mathbf{h}(\bar{t})\} + 1\right) \\
&= 2^{\mathbf{h}(e)}\mathbf{h}(e).
\end{aligned}$$

□

It follows that $(!)\mathfrak{B}$ is a $\Rightarrow_\partial$-reduction structure: since $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$ coincides with $\widetilde{\Rightarrow_{\lfloor\partial\rfloor}}_{(!)\mathfrak{B}}$, Corollary 2.6.25 entails that $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$ is strongly confluent. We can even refine this result following Lemma 2.6.27. First, let us call *bounded reduction structure* any $\Rightarrow_\partial$-reduction structure $\mathfrak{E}$ such that $\mathfrak{E} \subseteq (!)\mathfrak{B}$. Then Lemma 2.6.24 entails:

Corollary 2.6.29. *For all bounded reduction structure $\mathfrak{E}$, and all reduction $\epsilon \mathrel{\widetilde{\Rightarrow}_{\partial\mathfrak{E}}} \epsilon'$, $\epsilon' \mathrel{\widetilde{\Rightarrow}_{\partial\mathfrak{E}}} \mathsf{F}(\epsilon)$.*

It should moreover be clear that $\tau(M)$ is bounded for all $M \in \Lambda_{\mathbf{S}}$. In the next section, we show that $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$ allows to simulate parallel β-reduction via Taylor expansion. [21]

2.7 Simulating β-reduction under Taylor expansion

From now on, for all $M, N \in \Lambda_{\mathbf{S}}$, we write $M \mathrel{\widetilde{\Rightarrow}_\partial} N$ if $\tau(M) \mathrel{\widetilde{\Rightarrow}_\partial} \tau(N)$. More generally, for all $M \in \Lambda_{\mathbf{S}}$ and all $\sigma \in \mathbf{S}^{(!)\Delta}$, we write $M \mathrel{\widetilde{\Rightarrow}_\partial} \sigma$ (resp. $\sigma \mathrel{\widetilde{\Rightarrow}_\partial} M$) if $\tau(M) \mathrel{\widetilde{\Rightarrow}_\partial} \sigma$ (resp. $\sigma \mathrel{\widetilde{\Rightarrow}_\partial} \tau(M)$). We will show in Subsection 2.7.1 that $M \mathrel{\widetilde{\Rightarrow}_\partial} N$ as soon as $M \Rightarrow_\beta N$ where $\Rightarrow_\beta$ is the parallel β-reduction defined as follows:

Definition 2.7.1. We define *parallel β-reduction* on algebraic terms $\Rightarrow_\beta \subseteq \Lambda_{\mathbf{S}} \times \Lambda_{\mathbf{S}}$ by the following inductive rules:

- $x \Rightarrow_\beta x$;
- if $S \Rightarrow_\beta M'$ then $\lambda x\, S \Rightarrow_\beta \lambda x\, M'$;
- if $S \Rightarrow_\beta M'$ and $N \Rightarrow_\beta N'$ then $(S)\, N \Rightarrow_\beta (M')\, N'$;
- if $S \Rightarrow_\beta M'$ and $N \Rightarrow_\beta N'$ then $(\lambda x\, S)\, N \Rightarrow_\beta M'[N'/x]$;
- $0 \Rightarrow_\beta 0$;
- if $M \Rightarrow_\beta M'$ then $a.M \Rightarrow_\beta a.M'$;
- if $M \Rightarrow_\beta M'$ and $N \Rightarrow_\beta N'$ then $M + N \Rightarrow_\beta M' + N'$.

In particular, if $1 \in \mathbf{S}$ admits an opposite element $-1 \in \mathbf{S}$ then $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$ is degenerate. Indeed, we can consider $\Rightarrow_\beta$ up to the equality of vector λ-terms by setting $M \Rightarrow_{\widetilde{\beta}} N$ if there are $M' \simeq_{\mathrm{v}} M$ and $N' \simeq_{\mathrm{v}} N$ such that $M' \Rightarrow_\beta N'$. Since $\simeq_\tau$ subsumes $\simeq_{\mathrm{v}}$, the results of Subsection 2.7.1 will imply that $M \mathrel{\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}} N$ as soon as $M \Rightarrow_{\widetilde{\beta}} N$. If $-1 \in \mathbf{S}$, we have $M \Rightarrow_{\widetilde{\beta}}^* N$ for all $M, N \in \Lambda_{\mathbf{S}}$ by Example 2.4.16, hence $M \mathrel{\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}^*} N$.

Using reduction structures, we will nonetheless be able to define a consistent reduction relation containing β-reduction, but restricted to those algebraic λ-terms that have a normalizable Taylor expansion, in the sense to be defined in Section 2.8.

On the other hand, even assuming $\mathbf{S}$ is zerosumfree, Taylor expansions are not stable under $\widetilde{\Rightarrow}_\partial$: if $M \mathrel{\widetilde{\Rightarrow}_{\partial\mathfrak{B}}} \sigma'$, we know from the previous section that σ' is bounded and $M \mathrel{\widetilde{\Rightarrow_{\lfloor\partial\rfloor}}} \sigma'$, but there is no reason why σ' would be the Taylor expansion of an algebraic λ-term.

21. Observe that it is possible to establish Corollary 2.6.29 quite directly, following the proof of Lemma 2.6.24, and using only Lemma 2.6.28 and a variant of Lemma 2.6.12 (replacing b with $\mathbf{h}(e)$). This is the path adopted in the extended abstract [13] presented at *CSL 2017*.

We do know, however, that $\sigma' \mathrel{\widetilde{\Rightarrow}_{\partial\mathfrak{B}}} \mathsf{F}(\tau(M))$, which will allow us to obtain a weak conservativity result w.r.t. parallel β-reduction: for all reduction $M \mathrel{\widetilde{\Rightarrow}^*_{\partial\mathfrak{B}}} \sigma'$ there is a reduction $M \Rightarrow^*_\beta M'$ such that $\sigma' \mathrel{\widetilde{\Rightarrow}^*_{\partial\mathfrak{B}}} M'$, *i.e.* any $\widetilde{\Rightarrow}_{\partial\mathfrak{B}}$-reduction sequence from a Taylor expansion can be completed into a parallel β-reduction sequence (Subsection 2.7.2). Restricted to normalizable pure λ-terms, this will enable us to obtain an actual conservativity result.

2.7.1 Simulation of parallel β-reduction

We show that $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$ allows to simulate $\Rightarrow_\beta$ on $\mathbf{S}^{(!)\Delta}$, without any particular assumption on $\mathbf{S}$.

Lemma 2.7.2. *If $\sigma \mathrel{\widetilde{\Rightarrow}_{\partial\mathcal{S}}} \sigma'$ and $\overline{\tau} \mathrel{\widetilde{\Rightarrow}_{\partial\overline{\mathcal{T}}}} \overline{\tau'}$ then $\langle\lambda x\, \sigma\rangle\, \overline{\tau} \mathrel{\widetilde{\Rightarrow}_{\partial\langle\lambda x\, \mathcal{S}\rangle\, \overline{\mathcal{T}}}} \partial_x\sigma' \cdot \overline{\tau'}$.*

Proof. Assume there are summable families $(s_i)_{i\in I}$, $(\sigma'_i)_{i\in I}$, $\left(\overline{t}_j\right)_{j\in J}$ and $\left(\overline{\tau}'_j\right)_{j\in J}$, and families of scalars $(a_i)_{i\in I} \in \mathcal{S}^I$ and $(b_j)_{j\in J} \in \overline{\mathcal{T}}^J$ such that:

- $\sigma = \sum_{i\in I} a_i.s_i$, $\sigma' = \sum_{i\in I} a_i.\sigma'_i$ and $s_i \Rightarrow_\partial \sigma'_i$ for all $i \in I$;
- $\overline{\tau} = \sum_{j\in J} b_j.\overline{t}_j$, $\overline{\tau}' = \sum_{j\in J} b_j.\overline{\tau}'_j$, and $\overline{t}_j \in \overline{\mathcal{T}}$ and $\overline{t}_j \Rightarrow_\partial \overline{\tau}'_j$ for all $j \in J$.

By multilinear-continuity, the families $\left(\langle\lambda x\, s_i\rangle\, \overline{t}_j\right)_{i\in I, j\in J}$ and $\left(\partial_x\sigma'_i \cdot \overline{\tau}'_j\right)_{i\in I, j\in J}$ are summable, $\langle\lambda x\, \sigma\rangle\, \overline{\tau} = \sum_{i\in I, j\in J} a_i b_j.\langle\lambda x\, s_i\rangle\, \overline{t}_j$ and $\partial_x\sigma' \cdot \overline{\tau}' = \sum_{i\in I, j\in J} a_i b_j.\partial_x\sigma'_i \cdot \overline{\tau}'_j$. It is then sufficient to observe that $\langle\lambda x\, s_i\rangle\, \overline{t}_j \Rightarrow_\partial \partial_x\sigma'_i \cdot \overline{\tau}'_j$ for all $(i,j) \in I \times J$.

The additional requirement on resource supports is straightforwardly satisfied, since $\langle\lambda x\, s_i\rangle\, \overline{t}_j \in \langle\lambda x\, \mathcal{S}\rangle\, \overline{\mathcal{T}}$ for all $(i,j) \in I \times J$. □

Lemma 2.7.3. *If $\sigma \mathrel{\widetilde{\Rightarrow}_{\partial\mathcal{S}}} \sigma'$ then $\lambda x\, \sigma \mathrel{\widetilde{\Rightarrow}_{\partial\lambda x\, \mathcal{S}}} \lambda x\, \sigma'$. If moreover $\overline{\tau} \mathrel{\widetilde{\Rightarrow}_{\partial\overline{\mathcal{T}}}} \overline{\tau'}$ then $\langle\sigma\rangle\, \overline{\tau} \mathrel{\widetilde{\Rightarrow}_{\partial\langle\mathcal{S}\rangle\, \overline{\mathcal{T}}}} \langle\sigma'\rangle\, \overline{\tau'}$.*

Proof. Similarly to the previous lemma, each result follows from the multilinear-continuity of syntactic operators, and the contextuality of $\Rightarrow_\partial$. □

Lemma 2.7.4. *If $\sigma \mathrel{\widetilde{\Rightarrow}_{\partial\mathcal{S}}} \sigma'$ then $\sigma^! \mathrel{\widetilde{\Rightarrow}_{\partial\mathcal{S}^!}} \sigma'^!$.*

Proof. Assume there are summable families $(s_i)_{i\in I}$ and $(\sigma'_i)_{i\in I}$, and a family of scalars $(a_i)_{i\in I}$ such that $\sigma = \sum_{i\in I} a_i.s_i$, $\sigma' = \sum_{i\in I} a_i.\sigma'_i$ and $s_i \Rightarrow_\partial \sigma'_i$ for all $i \in I$.

Then by multilinear-continuity of the monomial construction, for all $n \in \mathbf{N}$, the families $([s_{i_1}, \ldots, s_{i_n}])_{i_1,\ldots,i_n\in I}$ and $\left([\sigma'_{i_1}, \ldots, \sigma'_{i_n}]\right)_{i_1,\ldots,i_n\in I}$ are summable, and

$$\sigma^n = \sum_{i_1,\ldots,i_n\in I} a_{i_1} \cdots a_{i_n} [s_{i_1}, \ldots, s_{i_n}]$$

and

$$\sigma'^n = \sum_{i_1,\ldots,i_n\in I} a_{i_1} \cdots a_{i_n}.[\sigma'_{i_1}, \ldots, \sigma'_{i_n}].$$

Since the supports of the monomial vectors σ^n (resp. σ'^n) for $n \in \mathbf{N}$ are pairwise disjoint, we obtain that the families $([s_{i_1}, \ldots, s_{i_n}])_{\substack{n\in\mathbf{N} \\ i_1,\ldots,i_n \in I}}$ and $\left([\sigma'_{i_1}, \ldots, \sigma'_{i_n}]\right)_{\substack{n\in\mathbf{N} \\ i_1,\ldots,i_n \in I}}$ are summable, and

$$\sigma^! = \sum_{n\in\mathbf{N}} \frac{1}{n!}.\sigma^n = \sum_{\substack{n\in\mathbf{N} \\ i_1,\ldots,i_n \in I}} \frac{a_{i_1}\cdots a_{i_n}}{n!}.[s_{i_1}, \ldots, s_{i_n}]$$

and

$$\sigma'^! = \sum_{n\in\mathbf{N}} \frac{1}{n!}.\sigma'^n = \sum_{\substack{n\in\mathbf{N} \\ i_1,\ldots,i_n \in I}} \frac{a_{i_1}\cdots a_{i_n}}{n!}.[\sigma'_{i_1}, \ldots, \sigma'_{i_n}]$$

which concludes the proof since each $[s_{i_1}, \ldots, s_{i_n}] \Rightarrow_\partial [\sigma'_{i_1}, \ldots, \sigma'_{i_n}]$. □

Lemma 2.7.5. *If $\epsilon \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{E}} \epsilon'$ and $\varphi \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{F}} \varphi'$ then $a.\epsilon \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{E}} a.\epsilon'$ and $\epsilon + \varphi \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{E}\cup\mathcal{F}} \epsilon' + \varphi'$.*

Proof. Follows directly from the definitions, using the fact that summable families form a $\mathbf{S}$-semimodule. □

Lemma 2.7.6. *If $M \Rightarrow_\beta M'$ then $M \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{T}(M)} M'$.*

Proof. By induction on the reduction $M \Rightarrow_\beta M'$ using Lemmas 2.7.2 to 2.7.4 in the cases of reduction from a simple term, and Lemma 2.7.5 in the case of reduction from an algebraic term. □

Recalling that $\mathcal{T}(M) \in \mathfrak{B}$ we obtain:

Corollary 2.7.7. *If $M \Rightarrow_\beta M'$ then $M \mathrel{\widetilde{\Rightarrow}}_{\partial\mathfrak{B}} M'$.*

Observe that these results hold on Taylor supports as well, which will be useful in the treatment of Taylor normalizable terms in Section 2.8:

Lemma 2.7.8. *If $M \Rightarrow_\beta M'$ then $\mathcal{T}(M) \mathrel{\widetilde{\Rightarrow}}_{\partial\mathcal{T}(M)} \mathcal{T}(M')$ in $\mathbf{B}^\Delta$.*

Proof. The proof is again by induction on the reduction $M \Rightarrow_\beta M'$ using Lemmas 2.7.2 to Lemma 2.7.5 in $\mathbf{B}^\Delta$. □

2.7.2 Conservativity

Definition 2.7.9. We define the *full parallel reduct* of simple terms and algebraic terms inductively as follows:

$$\begin{aligned} \mathsf{F}(x) &:= x & \mathsf{F}(0) &:= 0 \\ \mathsf{F}(\lambda x\, S) &:= \lambda x\, \mathsf{F}(S) & \mathsf{F}(a.M) &:= a.\mathsf{F}(M) \\ \mathsf{F}((\lambda x\, S)\, N) &:= \mathsf{F}(S)[\mathsf{F}(N)/x] & \mathsf{F}(M+N) &:= \mathsf{F}(M) + \mathsf{F}(N) \\ \mathsf{F}((S)\, N) &:= (\mathsf{F}(S))\, \mathsf{F}(N) & & \text{(if } S \text{ is not an abstraction).} \end{aligned}$$

As can be expected, we have $M' \Rightarrow_\beta \mathsf{F}(M)$ as soon as $M \Rightarrow_\beta M'$. In this subsection, we will show that a similar property holds for $\widetilde{\Rightarrow}_{\partial(!)\mathfrak{B}}$.

Recall that, by Lemma 2.6.27, the full reduction operator F on resource expressions extends to bounded resource vectors. We obtain:

Lemma 2.7.10. *For all bounded $\sigma_0 \in \mathbf{S}^{\Delta}$, $\overline{\tau} \in \mathbf{S}^{!\Delta}$, $\epsilon, \varphi \in \mathbf{S}^{(!)\Delta}$,*

$$\mathsf{F}(x) = x \qquad \mathsf{F}\left(\sigma^!\right) = \mathsf{F}(\sigma)^!$$
$$\mathsf{F}(\lambda x\, \sigma) = \lambda x\, \mathsf{F}(\sigma) \qquad \mathsf{F}(a.\epsilon) = a.\mathsf{F}(\epsilon)$$
$$\mathsf{F}(\langle \lambda x\, \sigma\rangle\, \overline{\tau}) = \partial_x \mathsf{F}(\sigma) \cdot \mathsf{F}(\overline{\tau}) \qquad \mathsf{F}(\epsilon + \varphi) = \mathsf{F}(\epsilon) + \mathsf{F}(\varphi)$$
$$\mathsf{F}(\langle \sigma_0\rangle\, \overline{\tau}) = \langle \mathsf{F}(\sigma_0)\rangle\, \mathsf{F}(\overline{\tau}) \quad \textit{(if there is no abstraction term in } |\sigma_0|\textit{).}$$

Proof. The proofs of those identities are basically the same as those of Lemmas 2.7.2 to 2.7.5, the necessary summability conditions following from Lemma 2.6.27. □

Lemma 2.7.11. *For all $M \in \Lambda_{\mathbf{S}}$, $\mathsf{F}(\tau(M)) = \tau(\mathsf{F}(M))$.*

Proof. We know that $\tau(M)$ is bounded. The identity is then proved by induction on simple terms and algebraic terms, using the previous lemma in each case. □

Lemma 2.7.12. *For all bounded term reduction structure $\mathfrak{S}$ and all $M \in \Lambda_{\mathbf{S}}$, if $M \widetilde{\Rightarrow}_{\partial\mathfrak{S}} \sigma'$ then $\sigma' \widetilde{\Rightarrow}_{\partial\mathfrak{S}} \mathsf{F}(M)$.*

Proof. By Corollary 2.6.29, $\sigma' \widetilde{\Rightarrow}_{\partial\mathfrak{S}} \mathsf{F}(\tau(M))$ and we conclude by the previous lemma. □

This result can then be generalized to sequences of $\widetilde{\Rightarrow}_{\partial}$-reductions.

Lemma 2.7.13. *For all bounded term reduction structure $\mathfrak{S}$ and all $M \in \Lambda_{\mathbf{S}}$, if $M \widetilde{\Rightarrow}^n_{\partial\mathfrak{S}} \sigma'$ then $\sigma' \widetilde{\Rightarrow}^n_{\partial\mathfrak{S}} \mathsf{F}^n(M)$.*

Proof. By induction on n. The case $n = 0$ is trivial, and the inductive case follows from the previous lemma and strong confluence of $\widetilde{\Rightarrow}_{\partial\mathfrak{S}}$: if $M \widetilde{\Rightarrow}^n_{\partial\mathfrak{S}} \sigma' \widetilde{\Rightarrow}_{\partial\mathfrak{S}} \tau$ then by induction hypothesis $\sigma' \widetilde{\Rightarrow}^n_{\partial\mathfrak{S}} \mathsf{F}^n(M)$, hence by strong confluence, there exists τ' such that $\tau \widetilde{\Rightarrow}^n_{\partial\mathfrak{S}} \tau'$ and $\mathsf{F}^n(M) \widetilde{\Rightarrow}_{\partial\mathfrak{S}} \tau'$; by the previous lemma, $\tau' \widetilde{\Rightarrow}_{\partial} \mathsf{F}^{n+1}(M)$. □

We have thus obtained some weak kind of conservativity of $\widetilde{\Rightarrow}_{\partial\mathfrak{B}}$ w.r.t. β-reduction, but it is not very satisfactory: the same result would hold for the tautological relation $\mathbf{S}\langle\mathfrak{B}\rangle \times \mathbf{S}\langle\mathfrak{B}\rangle$, which is indeed the same as $\widetilde{\Rightarrow}_{\partial\mathfrak{B}}$ if 1 has an opposite element in $\mathbf{S}$. Even when $\mathbf{S}$ is zerosumfree, the converse to Lemma 2.7.6 cannot hold in general if only because there can be distinct β-normal forms $M \not\simeq_{\mathrm{v}} N$ such that $M \simeq_\tau N$ (see Example 2.4.14). Under this hypothesis, we can nonetheless obtain an actual conservativity result on normalizable pure λ-terms as follows.

We write $\simeq_\beta$ for the symmetric, reflexive and transitive closure of $\Rightarrow_\beta$. Similarly, if $\mathfrak{C}$ is a reduction structure, we write $\simeq_{\partial\mathfrak{C}}$ for the equivalence on $\mathbf{S}\langle\mathfrak{C}\rangle$ induced by $\widetilde{\Rightarrow}_{\partial\mathfrak{C}}$.

Lemma 2.7.14. *Assume $\mathbf{S}$ is zerosumfree. Let $M, N \in \Lambda$ be such that M is normalizable. Then $M \simeq_{\partial\mathfrak{B}} N$ iff $M \simeq_\beta N$.*

Proof. Corollary 2.6.29 entails that, if $\mathfrak{E}$ is a bounded reduction structure, then $\epsilon \simeq_{\partial\mathfrak{E}} \epsilon'$ iff $\epsilon \mathrel{\widetilde{\Rightarrow}}^*_{\partial\mathfrak{E}} \mathsf{F}^n(\epsilon')$ for some $n \in \mathbf{N}$. Now assume $M \in \Lambda_{\mathbf{S}}$ is normalizable and write $\mathsf{NF}(M)$ for its normal form: in particular $M \mathrel{\widetilde{\Rightarrow}}^*_{\partial\mathfrak{B}} \mathsf{NF}(M)$, by Corollary 2.7.7. If $M \simeq_{\partial\mathfrak{B}} N$, we thus have $\mathsf{NF}(M) \simeq_{\partial\mathfrak{B}} N$, hence $\mathsf{NF}(M) \mathrel{\widetilde{\Rightarrow}}^*_{\partial\mathfrak{B}} \mathsf{F}^n(N)$ for some $n \in \mathbf{N}$. In particular, if $\mathbf{S}$ is zerosumfree, we obtain $\mathsf{NF}(M) \simeq_\tau \mathsf{F}^n(N)$. If moreover $M, N \in \Lambda$, we deduce $M \simeq_\beta N$ by the injectivity of τ on Λ. □

The next section will allow us to establish a similar conservativity result, without any assumption on $\mathbf{S}$, at the cost of restricting the reduction relation to normalizable resource vectors.

2.8 Normalizing Taylor expansions

Previous works on the normalization of Taylor expansions were restricted *a priori*, to a strict subsystem of the algebraic λ-calculus:

- the uniform setting of pure λ-terms [ER08; ER06a];
- the typed setting of an extension of system F to the algebraic λ-calculus [Ehr10];
- a λ-calculus extended with formal finite sums, rather than linear combinations [12; TAO17].

In all these, pathological terms were avoided, e.g. those involved in the inconsistency Example 2.4.16. Moreover observe that the very notion of normalizability is not compatible with $\simeq_{\mathrm{v}}$, and in particular the identity $0 \simeq_{\mathrm{v}} 0.M$: those previous works circumvented this incompatibility, either by imposing normalizability via typing, or by excluding the formation of the term $0.M$.

Our approach is substantially different. We introduce a notion of normalizability on resource vectors such that:

- both pure λ-terms and normalizable algebraic λ-terms (in particular typed algebraic λ-terms and normalizable λ-terms with sums) have a normalizable Taylor expansion;
- the restriction of $\widetilde{\Rightarrow}_\partial$ to normalizable resource vectors is a consistent extension of both β-reduction on pure λ-terms and normalization on algebraic λ-terms, without any assumption on the underlying semiring of scalars.

2.8.1 Normalizable resource vectors

We say $\epsilon \in \mathbf{S}^{(!)\Delta}$ is *normalizable* whenever the family $(\mathsf{NF}(e))_{e\in|\epsilon|}$ is summable. In this case, we write $\mathsf{NF}(\epsilon) := \sum_{e\in(!)\Delta} \epsilon_e.\mathsf{NF}(e)$.

Normalizable vectors form a finiteness space. Recall indeed from Subsection 2.3.1 that $e \geq_\partial e'$ iff $e \to^*_\partial \epsilon'$ with $e' \in |\epsilon'|$. If $e \in (!)\Delta$, we write $\uparrow e := \{e' \in (!)\Delta \ ; \ e' \geq_\partial e\}$. Then ϵ is normalizable iff for each normal resource expression e, $|\epsilon| \cap \uparrow e$ is finite: writing $(!)\mathcal{N} = \{e \in (!)\Delta \ ; \ e \text{ is normal}\}$ and $(!)\mathfrak{N} = \{\uparrow e \ ; \ e \in (!)\mathcal{N}\}^\perp \cap (!)\mathfrak{F}_{\mathbf{fv}}$, we obtain that $\mathbf{S}\langle(!)\mathfrak{N}\rangle$ is the set of normalizable resource vectors. Observe that NF is defined on all $\mathbf{S}\langle(!)\mathfrak{N}\rangle$ but is guaranteed to be linear-continuous only when restricted to subsemimodules of the form $\mathbf{S}^{\mathcal{E}}$ with $\mathcal{E} \in (!)\mathfrak{N}$.

For our study of hereditarily determinable terms in Section 2.9, it will be useful to decompose $(!)\mathfrak{N}$ into a decreasing sequence of finiteness structures.

Definition 2.8.1. We define the *monomial depth* $\mathbf{d}(e) \in \mathbf{N}$ of a resource expression $e \in (!)\Delta$ as follows:

$$\mathbf{d}(x) := 0 \qquad \mathbf{d}(\langle s\rangle\,\bar{t}) := \max(\mathbf{d}(s), \mathbf{d}(\bar{t}))$$
$$\mathbf{d}(\lambda x\, s) := \mathbf{d}(s) \qquad \mathbf{d}([t_1, \dots, t_n]) := 1 + \max\{\mathbf{d}(t_i)\ ;\ 1 \le i \le n\}$$

We write $(!)\mathcal{N}_d = \{e \in (!)\mathcal{N}\ ;\ \mathbf{d}(e) \le d\}$ so that $(!)\mathcal{N} = \bigcup_{d\in\mathbf{N}} (!)\mathcal{N}_d$. We then write $(!)\mathfrak{N}_d = \{\uparrow e\ ;\ e \in (!)\mathcal{N}_d\}^{\perp} \cap (!)\mathfrak{F}_{\mathbf{fv}}$ so that $(!)\mathfrak{N} = \bigcap_{d\in\mathbf{N}} (!)\mathfrak{N}_d$. Each finiteness structure $(!)\mathfrak{N}_d$ is moreover a reduction structure for any reduction relation contained in $\to_\partial^*$ (and so is $(!)\mathfrak{N}$). Indeed, writing $\downarrow e = \{e' \in (!)\Delta\ ;\ e \ge_\partial e'\}$ and $\downarrow\mathcal{E} = \bigcup_{e\in\mathcal{E}} \downarrow e$, we obtain:

Lemma 2.8.2. *If $\mathcal{E} \in (!)\mathfrak{N}_d$ then $\downarrow\mathcal{E} \in (!)\mathfrak{N}_d$.*

Proof. Let $e'' \in (!)\mathcal{N}_d$ and $e' \in \downarrow\mathcal{E} \cap \uparrow e''$. Necessarily, there is $e \in \mathcal{E}$ such that $e \ge_\partial e'$. Then $e \in \mathcal{E} \cap \uparrow e''$: since $\mathcal{E} \in (!)\mathfrak{N}_d$, there are finitely many values for e hence for e' by Lemma 2.3.13. □

It follows that normalizable vectors are stable under reduction:

Lemma 2.8.3. *If $\epsilon \mathrel{\widetilde{\Rightarrow}}_{\partial (!)\mathfrak{N}} \epsilon'$ then $\epsilon' \in \mathbf{S}\langle(!)\mathfrak{N}\rangle$ and $\mathsf{NF}(\epsilon) = \mathsf{NF}(\epsilon')$.*

Proof. Assume there exists $\mathcal{E} \in (!)\mathfrak{N}$ and families $(a_i)_{i\in I} \in \mathbf{S}^I$, $(e_i)_{i\in I} \in (!)\Delta^I$ and $(\epsilon'_i)_{i\in I} \in \mathbf{N}[(!)\Delta]^I$ such that:

- $(e_i)_{i\in I}$ is summable and $\epsilon = \sum_{i\in I} a_i.e_i$;
- $(\epsilon'_i)_{i\in I}$ is summable and $\epsilon' = \sum_{i\in I} a_i.\epsilon'_i$;
- for all $i \in I$, $e_i \in \mathcal{E}$ and $e_i \mathrel{\widetilde{\Rightarrow}}_\partial \epsilon'_i$.

We obtain that $\mathcal{E}' := \bigcup_{i\in I} |\epsilon'_i| \in (!)\mathfrak{N}$ by Lemma 2.8.2, hence $\epsilon' \in \mathbf{S}\langle(!)\mathfrak{N}\rangle$ since $|\epsilon'| \subseteq \mathcal{E}'$. Then, by the linear-continuity of NF on $\mathbf{S}^{\mathcal{E}'}$,

$$\mathsf{NF}(\epsilon) = \sum_{i\in I} a_i.\mathsf{NF}(e_i) = \sum_{i\in I} a_i.\mathsf{NF}(\epsilon'_i) = \mathsf{NF}\left(\sum_{i\in I} a_i.\epsilon'_i\right) = \mathsf{NF}(\epsilon').$$

□

As a direct consequence, we obtain that $\simeq_{\partial (!)\mathfrak{N}}$ is consistent, without any additional condition on the semiring $\mathbf{S}$:

Corollary 2.8.4. *If $\epsilon \simeq_{\partial (!)\mathfrak{N}} \epsilon'$ (in particular $\epsilon, \epsilon' \in \mathbf{S}\langle(!)\mathfrak{N}\rangle$) then $\mathsf{NF}(\epsilon) = \mathsf{NF}(\epsilon')$.*

We can moreover show that the normal form of a Taylor normalizable term is obtained as the limit of the parallel left reduction strategy. Let us first precise the kind of convergence we consider. With the notations of Subsection 2.2.3, we say a sequence $\overrightarrow{\xi} = (\xi_n)_{n\in\mathbf{N}} \in \left(\mathbf{S}^X\right)^{\mathbf{N}}$ of vectors converges to ξ' if, for all $x \in X$ there exists $n_x \in \mathbf{N}$ such that, for all $n \ge n_x$, $\xi_{n,x} = \xi'_x$. In other words we consider the product topology on $\mathbf{S}^X$, $\mathbf{S}$ being endowed with the discrete topology. Similarly to the notion of summability, this notion of convergence coincides with that induced by the linear topology on $\mathbf{S}^X$ associated with the maximal finiteness structure $\mathfrak{P}(X)$ on X: in this particular case, a base of neighbourhoods of 0 is given by the sets $\left\{\xi \in \mathbf{S}^X \,;\, |\xi| \cap \mathcal{X}' = \emptyset\right\}$ for $\mathcal{X}' \in \mathfrak{P}(X)^\perp = \mathfrak{P}_f(X)$, or equivalently by the the sets $\left\{\xi \in \mathbf{S}^X \,;\, x \notin |\xi|\right\}$ for $x \in X$.

The parallel left reduction strategy on resource vectors is defined as follows.

Definition 2.8.5. We define the *left reduct of a resource expression* inductively as follows:

$$\begin{aligned}
\mathsf{L}(\lambda x\, s) &:= \lambda x\, \mathsf{L}(s)\\
\mathsf{L}([t_1, \dots, t_n]) &:= [\mathsf{L}(t_1), \dots, \mathsf{L}(t_n)]\\
\mathsf{L}\big(\langle x\rangle\, \bar{t}_1 \cdots \bar{t}_n\big) &:= \langle x\rangle\, \mathsf{L}\big(\bar{t}_1\big) \cdots \mathsf{L}\big(\bar{t}_n\big)\\
\mathsf{L}\big(\langle \lambda x\, s\rangle\, \bar{t}_0\, \bar{t}_1 \cdots \bar{t}_n\big) &:= \big\langle \partial_x s \cdot \bar{t}_0\big\rangle\, \bar{t}_1 \cdots \bar{t}_n.
\end{aligned}$$

This is extended to finite sums of resource expressions by linearity: $\mathsf{L}(\sum_{i=1}^n e_i) = \sum_{i=1}^n \mathsf{L}(e_i)$.

Lemma 2.8.6. *For all resource expression $e \in (!)\Delta$, $e \Rightarrow_{(1)} \mathsf{L}(e)$.*

Proof. Easy by induction on e. □

In particular $\mathsf{NF}(e) = \mathsf{NF}(\mathsf{L}(e))$ for all $e \in (!)\Delta$. By Lemma 2.6.12, we moreover obtain that if $e' \in |\mathsf{L}(e)|$ then $\mathbf{s}(e) \leq 4\mathbf{s}(e')$ and $\mathbf{fv}(e) = \mathbf{fv}(e')$. As a consequence $(\mathsf{L}(e))_{e\in(!)\Delta}$ is summable. For all $\epsilon \in \mathbf{S}^{(!)\Delta}$, we set

$$\mathsf{L}(\epsilon) := \sum_{e\in(!)\Delta} \epsilon_e.\mathsf{L}(e)$$

and obtain a linear-continuous map on resource vectors.

For all $\epsilon \in \mathbf{S}^{(!)\Delta}$, we write $\epsilon\restriction_{(!)\mathcal{N}}$ for the *projection of ϵ on normal resource expressions*: $\epsilon\restriction_{(!)\mathcal{N}} := \sum_{e\in(!)\mathcal{N}} \epsilon_e.e \in \mathbf{S}^{(!)\mathcal{N}}$. We obtain:

Theorem 2.8.7. *For all normalizable resource vector $\epsilon \in \mathbf{S}\langle(!)\mathfrak{N}\rangle$, $\left(\mathsf{L}^k(\epsilon)\restriction_{(!)\mathcal{N}}\right)_{k\in\mathbf{N}}$ converges to $\mathsf{NF}(\epsilon)$ in $\mathbf{S}^{(!)\mathcal{N}}$.*

Proof. Fix $e' \in \mathcal{N}$. Since $|\epsilon| \in (!)\mathfrak{N}$, $\mathcal{E} := |\epsilon| \cap {\uparrow} e'$ is finite. Let k' be such that $\mathsf{L}^{k'}(e)$ is normal for all $e \in \mathcal{E}$. Then $\mathsf{NF}(\epsilon)_{e'} = \sum_{e\in|\epsilon|} \epsilon_e.\mathsf{NF}(e)_{e'} = \sum_{e\in\mathcal{E}} \epsilon_e.\mathsf{NF}(e)_{e'} = \sum_{e\in\mathcal{E}} \epsilon_e.\mathsf{L}^{k'}(e)_{e'}$. Moreover, by the linear-continuity of L^k on resource vectors, $\left(\mathsf{L}^k(\epsilon)\restriction_{(!)\mathcal{N}}\right)_{e'} = \mathsf{L}^k(\epsilon)_{e'} = \sum_{e\in\Delta} \epsilon_e.\mathsf{L}^k(e)_{e'} = \sum_{e\in\mathcal{E}} \epsilon_e.\mathsf{L}^k(e)_{e'} = \sum_{e\in\mathcal{E}} \epsilon_e.\mathsf{L}^{k'}(e)_{e'}$. □

Observe that the projection on normal expressions is essential:

Example 2.8.8. *Consider the looping term $\Omega := (\lambda x\,(x)\,x)\,\lambda x\,(x)\,x$: one can check that $\mathsf{NF}(\tau(\Omega)) = \tau(\Omega)\restriction_{\mathcal{N}} = 0$, but it will follow from the results of subsection 2.8.2 that $\mathsf{L}^k(\tau(\Omega)) = \tau(\Omega) \neq 0$ for all $k \in \mathbf{N}$.*

Analyzing this phenomenon was fundamental in the characterization of strongly normalizable λ-terms by a finiteness structure on resource terms, obtained by Pagani, Tasson and the author [12].

2.8.2 Taylor normalizable terms

It is possible to transfer some of the good properties of reduction on normalizable vectors to those algebraic λ-terms that have a normalizable Taylor expansion. More precisely, we say $M \in \Lambda_{\mathbf{S}}$ is *Taylor normalizable* if $\mathcal{T}(M) \in (!)\mathfrak{N}$. Then:

Lemma 2.8.9. *Assume $M, M' \in \Lambda_{\mathbf{S}}$ are such that $M \Rightarrow_\beta M'$. Then M is Taylor normalizable iff M' is Taylor normalizable.*

Proof. First observe that by Lemma 2.7.8, we have $\mathcal{T}(M) \,\widetilde{\Rrightarrow}_{\partial \mathcal{T}(M)}\, \mathcal{T}(M')$ in $\mathbf{B}^\Delta$. Moreover observe that $\mathbf{B}\langle\mathfrak{N}\rangle$ is nothing but $\mathfrak{N}$.

Assume M is Taylor normalizable, *i.e.* $\mathcal{T}(M) \in \mathfrak{N}$: by Lemma 2.8.3, $\mathcal{T}(M') \in \mathbf{B}\langle\mathfrak{N}\rangle$, *i.e.* M' is Taylor normalizable.

Conversely, assume M' is Taylor normalizable and let $s'' \in \mathcal{N}$ and $\mathcal{S} := \mathcal{T}(M) \cap {\uparrow} s''$: we prove $\mathcal{S}$ is finite. Fix an enumeration $(s_k)_{k\in K} \in \mathcal{S}^K$ of $\mathcal{S}$: $\mathcal{S} = \{s_k \,;\, k \in K\}$. Since $\mathcal{T}(M) \,\widetilde{\Rrightarrow}_{\partial \mathcal{T}(M)}\, \mathcal{T}(M')$, we have $\mathcal{T}(M) = \{t_i \,;\, i \in I\}$ and $\mathcal{T}(M') = \bigcup_{i\in I} |\tau'_i|$ with $t_i \Rrightarrow_\partial \tau'_i$ for all $i \in I$. Now for all $k \in K$, there exists $i \in I$ such that $s_k = t_i$. Since $s_k \geq_\partial s''$, $\tau'_i \neq 0$ and we can fix $s'_k \in |\tau'_i| \subseteq \mathcal{T}(M')$ such that $s_k \ggcurly_\partial s'_k \geq_\partial s''$. Since $\mathcal{T}(M') \in \mathfrak{N}$, the set $\{s'_k \,;\, k \in K\}$ is finite. Then $\mathcal{S} \subseteq \left\{s \in \Delta \,;\, k \in K,\, s \ggcurly_{(\mathbf{h}(M))} s'_k\right\}$ which is finite by Lemma 2.6.12. $\square$

The consistency of β-reduction on Taylor normalizable terms follows.

Theorem 2.8.10. *Assume $M, M' \in \Lambda_{\mathbf{S}}$ are such that $M \simeq_\beta M'$. Then M is Taylor normalizable iff M' is Taylor normalizable, and in this case $\mathsf{NF}(\tau(M)) = \mathsf{NF}(\tau(M'))$.*[22]

Proof. The first part is a direct corollary of Lemma 2.8.9. By Lemma 2.7.6, it follows that $M \simeq_{\partial\,(!)\mathfrak{N}} M'$, and then we conclude by Corollary 2.8.4. $\square$

In other words, when restricted to Taylor normalizable terms, the normal form of Taylor expansion is a valid notion of denotation. Remark that, in general, it is not possible to generalize this result to those terms M such that $\tau(M)$ is normalizable because of the interaction with coefficients: consider, *e.g.*, $0 \simeq_\tau (I)\,\infty_x + (-1).(I)\,\infty_x \Rightarrow_\beta \infty_x + (-1).(I)\,\infty_x$, and observe that $\tau(\infty_x + (-1).(I)\,\infty_x) \notin \mathbf{S}\langle\mathfrak{N}\rangle$.

Definition 2.8.11. We define the *left reduct of an algebraic λ-term* inductively as follows:

$$\begin{aligned} \mathsf{L}(\lambda x\, S) &:= \lambda x\, \mathsf{L}(S) & \mathsf{L}(0) &:= 0 \\ \mathsf{L}((x)\, M_1 \cdots M_n) &:= (x)\, \mathsf{L}(M_1) \cdots \mathsf{L}(M_n) & \mathsf{L}(a.M) &:= a.\mathsf{L}(M) \\ \mathsf{L}((\lambda x\, S)\, M_0\, M_1 \cdots M_n) &:= (S[M_0/x])\, M_1 \cdots M_n & \mathsf{L}(M+N) &:= \mathsf{L}(M) + \mathsf{L}(N) \end{aligned}$$

Observe that this definition is exhaustive by Fact 2.4.18. It should be clear that $M \Rightarrow_\beta \mathsf{L}(M)$ for all term M, and that $\mathsf{L}(M) = M$ when M is in normal form (although the converse may not hold). Now we can establish that L commutes with Taylor expansion.

22. In the standard terminology of denotational semantics, Theorem 2.8.10 expresses the soundness of $\mathsf{NF}(\tau(\cdot))$ on Taylor normalizable terms.

Lemma 2.8.12. *For all* $\sigma \in \mathbf{S}^{\Delta}$, $\mathsf{L}(\sigma^!) = \mathsf{L}(\sigma)^!$.

Proof. First observe that by the definition of L and the linear-continuity of both L and the monomial construction, for all $\sigma_1, \dots, \sigma_n \in \mathbf{S}^{\Delta}$, we have $\mathsf{L}([\sigma_1, \dots, \sigma_k]) = [\mathsf{L}(\sigma_1), \dots, \mathsf{L}(\sigma_k)]$. In particular, $\mathsf{L}(\sigma^k) = \mathsf{L}(\sigma)^k$. We deduce that $\mathsf{L}(\sigma^!) = \mathsf{L}(\sum_{k\in\mathbf{N}} \frac{1}{k!}.\sigma^k) = \sum_{k\in\mathbf{N}} \frac{1}{k!}.\mathsf{L}(\sigma)^k = \mathsf{L}(\sigma)^!$, by the linear-continuity of L. $\square$

Lemma 2.8.13. *For all* $M \in \Lambda_{\mathbf{S}}$, $\mathsf{L}(\tau(M)) = \tau(\mathsf{L}(M))$.

Proof. By induction on the definition of $\mathsf{L}(M)$: in addition to the inductive hypothesis and the linear-continuity of L, we use Lemma 2.8.12 in the case of a head variable, and Lemmas 2.4.7, 2.4.10 and 2.8.12 in the case of a head β-redex. $\square$

As a direct corollary of Theorem 2.8.7, we obtain:

Theorem 2.8.14. *For all Taylor normalizable term* M*, the sequence of normal resource vectors* $\left(\tau(\mathsf{L}^k(M))\restriction_{\mathcal{N}}\right)_{k\in\mathbf{N}}$ *converges to* $\mathsf{NF}(\tau(M))$ *in* $\mathbf{S}^{\mathcal{N}}$.

This property is very much akin to the fact that the Böhm tree $\mathsf{BT}(M)$ of a pure λ-term M is obtained as the limit (in an order theoretic sense) of normal form approximants of the left reducts of M. This analogy will be made explicit in Section 2.9. Before that, we apply our results to normalizable algebraic λ-terms.

2.8.3 Taylor expansion and normalization commute on the nose

By a general standardization argument, we can show that parallel reduction is a normalization strategy:

Lemma 2.8.15. *An algebraic* λ*-term* M *is normalizable iff there exists* $k \in \mathbf{N}$*, such that* $\mathsf{L}^k(M) = \mathsf{NF}(M)$.

Proof. Recall that we consider algebraic λ-terms up to $\simeq_+$ only. Then one can for instance use the general standardization technique developed by Leventis for a slightly different presentation of the calculus [24]. $\square$

A direct consequence is that M normalizes iff the judgement $M \Downarrow$ can be derived inductively by the following rules: [23]

$$\frac{S \Downarrow}{\lambda x\, S \Downarrow} \qquad \frac{M_1 \Downarrow \quad \cdots \quad M_n \Downarrow}{(x)\, M_1 \cdots M_n \Downarrow} \qquad \frac{(S[M_0/x])\, M_1 \cdots M_n \Downarrow}{(\lambda x\, S)\, M_0\, M_1 \cdots M_n \Downarrow} \qquad \frac{}{0 \Downarrow} \qquad \frac{M \Downarrow}{a.M \Downarrow} \qquad \frac{M \Downarrow \quad N \Downarrow}{(M+N) \Downarrow}$$

In the remaining of this subsection, we prove that normalizable algebraic λ-terms are Taylor normalizable, using a reducibility technique: like in Ehrhard's work for the typed case [Ehr10], or our previous work for the strongly normalizable case [12], $(!)\mathfrak{N}$ is the analogue of a reducibility candidate. We prove each key property (Lemmas 2.8.16 to 2.8.20) using the family of structures $(!)\mathfrak{N}_d$ rather than $(!)\mathfrak{N}$ directly: this will be useful in section 2.9, while the corresponding results for $(!)\mathfrak{N}$ are immediately derived from those.

23. Moreover, it seems natural to conjecture that if $M \Downarrow$ then M (or, rather, its $\simeq_{\mathrm{v}}$-class) is normalizable in the sense of Alberti [23], and then the obtained normal forms are the same (up to $\simeq_{\mathrm{v}}$).

Lemma 2.8.16. *If $\mathcal{S} \in \mathfrak{N}_d$ then $\lambda x\, \mathcal{S} \in \mathfrak{N}_d$.*

Proof. Let $t' \in \mathcal{N}_d$ and $t \in (\lambda x\, \mathcal{S}) \cap \uparrow t'$. Necessarily, $t = \lambda x\, s$ and $t' = \lambda x\, s'$ with $s \in \mathcal{S} \cap \uparrow s'$ which is finite by assumption. $\square$

Lemma 2.8.17. *If $\mathcal{S} \in \mathfrak{N}_d$ then $\mathcal{S}^! \in !\mathfrak{N}_{d+1}$.*

Proof. Let $\bar{t}' \in !\mathcal{N}_{d+1}$ and $\bar{t} \in \mathcal{S}^! \cap \uparrow \bar{t}'$. Write $n = \#\bar{t}'$. Without loss of generality, we can write $\bar{t} = [t_1, \dots, t_n]$ and $\bar{t}' = [t'_1, \dots, t'_n]$ so that $t_i \geq_\partial t'_i$ and $t'_i \in \mathcal{N}_d$, for all $i \in \{1, \dots, n\}$. Since $\bar{t} \in \mathcal{S}^!$, each $t_i \in \mathcal{S}$. Since $\mathcal{S} \in \mathfrak{N}_d$, t'_i being fixed, there are finitely many possible values for each t_i. $\square$

Lemma 2.8.18. *If $\overline{\mathcal{T}}_1, \dots, \overline{\mathcal{T}}_n \in !\mathfrak{N}_d$ then $\langle x \rangle\, \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n \in \mathfrak{N}_d$.*

Proof. Let $t' \in \mathcal{N}_d$ and $t \in \left(\langle x \rangle\, \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n\right) \cap \uparrow t'$. Necessarily, $t = \langle x \rangle\, \bar{t}_1 \cdots \bar{t}_n$ and $t' = \langle x \rangle\, \bar{t}'_1 \cdots \bar{t}'_n$ and, for each $i \in \{1, \dots, n\}$, $\bar{t}_i \in \overline{\mathcal{T}}_i$, $\bar{t}_i \geq_\partial \bar{t}'_i$ and $\bar{t}'_i \in !\mathcal{N}_d$: since $\overline{\mathcal{T}}_i \in !\mathfrak{N}_d$, there are finitely many possible values for each $\bar{t}_i$. $\square$

Corollary 2.8.19. *If $\mathcal{T}_1, \dots, \mathcal{T}_n \in \mathfrak{N}_d$ then $\langle x \rangle\, \mathcal{T}_1^! \cdots \mathcal{T}_n^! \in \mathfrak{N}_{d+1}$.*

Lemma 2.8.20. *If $\left\langle \partial_x \mathcal{S} \cdot \overline{\mathcal{T}}_0 \right\rangle \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n \in \mathfrak{N}_d$ then $\langle \lambda x\, \mathcal{S} \rangle\, \overline{\mathcal{T}}_0\, \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n \in \mathfrak{N}_d$.*

Proof. Let $u' \in \mathcal{N}_d$, and let $u \in \left(\langle \lambda x\, \mathcal{S} \rangle\, \overline{\mathcal{T}}_0\, \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n\right) \cap \uparrow u'$. In other words, $u' \in |\mathsf{NF}(u)|$ and we can write $u = \langle \lambda x\, s \rangle\, \bar{t}_0\, \bar{t}_1 \cdots \bar{t}_n$ with $s \in |\mathcal{S}|$ and $\bar{t}_i \in \left|\overline{\mathcal{T}}_i\right|$ for $i \in \{0, \dots, n\}$. Write $v = \left\langle \partial_x s \cdot \bar{t}_0 \right\rangle \bar{t}_1 \cdots \bar{t}_n$: Corollary 2.3.14 entails $v \geq_\partial u'$, hence we have $v \in \left(\left\langle \partial_x \mathcal{S} \cdot \overline{\mathcal{T}}_0 \right\rangle \overline{\mathcal{T}}_1 \cdots \overline{\mathcal{T}}_n\right) \cap \uparrow u'$. By assumption, there are finitely many possible values for v. Then, v being fixed, by Lemma 2.3.12, we have $\mathbf{fv}(u) = \mathbf{fv}(v)$ and $\mathbf{s}(u) \leq 2\mathbf{s}(v) + 2$, hence there are finitely many possible values for u. $\square$

Theorem 2.8.21. *If M is normalizable, then $\mathcal{T}(M) \in \mathfrak{N}$, and $\tau(M) \in \mathbf{S}\langle \mathfrak{N} \rangle$.*

Proof. By induction on the derivation of $M \Downarrow$: Lemma 2.8.16, Corollary 2.8.19 and Lemma 2.8.20 respectively entail the translation of the first three inductive rules through Taylor expansion. The other three follow from the fact that $\mathfrak{N}$ is a resource structure (because it is a finiteness structure). $\square$

It remains to prove that in this case, $\tau(\mathsf{NF}(M))$ is indeed the normal form of $\tau(M)$.

Theorem 2.8.22. *If M is normalizable, then* $\mathsf{NF}(\tau(M)) = \tau(\mathsf{NF}(M))$.

Proof. By Theorem 2.8.21, M is Taylor normalizable. Then Theorem 2.8.10 entails $\mathsf{NF}(\tau(M)) = \mathsf{NF}(\tau(\mathsf{NF}(M))) = \tau(\mathsf{NF}(M))$. $\square$

2.8.4 Conservativity

The restriction to normalizable vectors allows us to prove an analogue of Lemma 2.7.14, without any assumption on the semiring of scalars.

Lemma 2.8.23. *Let $M, N \in \Lambda$ be normalizable. Then $M \simeq_{\partial\mathfrak{N}} N$ iff $M \simeq_\beta N$.*

Proof. Assume $M \simeq_{\partial\mathfrak{N}} N$. By Corollary 2.8.4, we have $\mathsf{NF}(\tau(M)) = \mathsf{NF}(\tau(M'))$. By Theorem 2.8.22, we obtain $\mathsf{NF}(M) \simeq_\tau \mathsf{NF}(M')$. Since M and N are pure λ-terms, we deduce $\mathsf{NF}(M) = \mathsf{NF}(N)$ from the injectivity of τ on Λ.

The reverse direction is similar to Theorem 2.8.10 and does not depend on M and N being pure λ-terms: apply Lemmas 2.7.6, 2.8.3 and 2.8.9 to the reduction path from M to N $\square$

We can adapt this result to non-normalizing pure λ-terms thanks to previous work by Ehrhard and Regnier: [24]

Theorem 2.8.24 ([ER08; ER06a])**.** *For all pure λ-term $M \in \Lambda$, $\mathcal{T}(M) \in \mathfrak{N}$ and $\mathsf{NF}(\tau(M)) = \tau(\mathsf{BT}(M))$ where $\mathsf{BT}(M)$ denotes the Böhm tree of M.*

Here *Böhm tree* is to be understood as *generalized normal form for left β-reduction.* In particular it does not involve η-expansion. More formally, the Böhm tree of a λ-term is the possibly infinite tree obtained coinductively as follows:

- if M is head normalizable and its head normal form is $\lambda x_1 \cdots \lambda x_n\,(x)\,N_1 \cdots N_k$ then $\mathsf{BT}(M) := \lambda x_1 \cdots \lambda x_n\,(x)\,\mathsf{BT}(N_1) \cdots \mathsf{BT}(N_k)$
- otherwise $\mathsf{BT}(M) := \perp$, where $\perp$ is a constant representing unsolvability.

Taylor expansion can be generalized to Böhm trees [ER06a], setting in particular $\tau(\perp) = 0$: this is still injective.

Lemma 2.8.25. *If $M, N \in \Lambda$ and $M \simeq_{\partial\mathfrak{N}} N$ then $\mathsf{BT}(M) = \mathsf{BT}(N)$.*

Proof. By Corollary 2.8.4, we have $\mathsf{NF}(\tau(M)) = \mathsf{NF}(\tau(M'))$. By Theorem 2.8.24, we obtain $\tau(\mathsf{BT}(M)) = \tau(\mathsf{BT}(N))$. We conclude since τ is injective on Böhm trees. $\square$

In the next and final section, we prove a generalization of Theorem 2.8.24 to the non-uniform setting which is made possible by the results we have achieved so far.

2.9 Normal form of Taylor expansion, *façon* Böhm trees

The Böhm tree construction is often introduced as the limit of an increasing sequence $(\mathsf{BT}_d(M))_{d\in\mathbf{N}}$ of finite normal form approximants, *aka* finite Böhm trees, where $\mathsf{BT}_d(M)$ is defined inductively as follows:

- $\mathsf{BT}_0(M) = \perp$;
- if M is head normalizable and its head normal form is $\lambda x_1 \cdots \lambda x_n\,(x)\,N_1 \cdots N_k$ then $\mathsf{BT}_{d+1}(M) := \lambda x_1 \cdots \lambda x_n\,(x)\,\mathsf{BT}_d(N_1) \cdots \mathsf{BT}_d(N_k)$

24. We could as well rely on Theorem 2.9.14, to be proved in the next section.

– otherwise $\mathsf{BT}_{d+1}(M) := \bot$;

and the order on Böhm trees is the contextual closure of the inequality $\bot \leq M$ for all M.

In this final section of our paper, we show that the *normal form of Taylor expansion* operator generalizes this construction to the class of *hereditarily determinable terms*: these encompass both all pure λ-terms and all normalizable algebraic λ-terms, but exclude terms such as ∞_x, that produce unbounded sums of head normal forms. More precisely, we show that any hereditarily determinable term M is Taylor normalizable, and moreover admits a sequence of approximants $(\mathsf{NA}_d(M))_{d\in\mathbf{N}}$, such that each $\mathsf{NA}_d(M)$ is an algebraic λ-term in normal form, and the sequence of normal term vectors $(\tau(\mathsf{NA}_d(M)))_{d\in\mathbf{N}}$ converges to $\mathsf{NF}(\tau(M))$.

The results in this section should not hide the fact that the more fundamental notion is that of Taylor normalizable term, which arises naturally by combining Taylor expansion with the normalization of resource terms, subject to a summability condition. We believe this approach is quite robust, and may be adapted modularly following both parameters: to other systems admitting Taylor expansion; and to variants of summability, possibly associated with topological conditions of the semiring of scalars.

By contrast, the definition of hereditarily determinable terms is essentially *ad-hoc*. Its only purpose is to allow us to generalize Theorem 2.8.24 and support our claim that: *the normal form of Taylor expansion extends the notion of Böhm tree to the non-uniform setting.*

2.9.1 Taylor unsolvability

In the ordinary λ-calculus, head normalizable terms are exactly those with a non trivial Böhm tree. This is reflected via Taylor expansion: it is easy to check that $\mathsf{NF}(\tau(M)) = 0$ iff M has no head normal form. In the non uniform setting, a similar result holds, although we need to be more careful about the interplay between reduction and coefficients.

Definition 2.9.1. We say an algebraic λ-term M (resp. simple term S) is *weakly solvable* if the judgement $M \Downarrow_{\mathrm{w}}$ can be derived inductively by the following rules:

$$\frac{}{(x)\, M_1 \cdots M_n \Downarrow_{\mathrm{w}}} \qquad \frac{S \Downarrow_{\mathrm{w}}}{\lambda x\, S \Downarrow_{\mathrm{w}}} \qquad \frac{(S[M_0/x])\, M_1 \cdots M_n \Downarrow_{\mathrm{w}}}{(\lambda x\, S)\, M_0\, M_1 \cdots M_n \Downarrow_{\mathrm{w}}} \qquad \frac{M \Downarrow_{\mathrm{w}}}{a.M \Downarrow_{\mathrm{w}}} \qquad \frac{M \Downarrow_{\mathrm{w}}}{M + N \Downarrow_{\mathrm{w}}} \qquad \frac{N \Downarrow_{\mathrm{w}}}{M + N \Downarrow_{\mathrm{w}}}$$

It should be clear that, if M is a pure λ-term, $M \Downarrow_{\mathrm{w}}$ iff M is head normalizable. In the general case, we show that $M \Downarrow_{\mathrm{w}}$ iff normalizing the Taylor expansion of M yields a non trivial result. More formally:

Definition 2.9.2. We say an algebraic λ-term $M \in \Lambda_{\mathbf{S}}$ is *Taylor unsolvable* and write $M \Uparrow$ if $\mathsf{NF}(s) = 0$ for all $s \in \mathcal{T}(M)$.

In particular, if $M \Uparrow$ then $\tau(M) \in \mathbf{S}\langle\mathfrak{N}\rangle$ and $\mathsf{NF}(\tau(M)) = 0$: indeed, $|\tau(M)| \subseteq \mathcal{T}(M)$. Beware that the reverse implication does not hold in general. We can then show that $M \Downarrow_{\mathrm{w}}$ iff M is Taylor solvable (Lemmas 2.9.3 and 2.9.4).

Lemma 2.9.3. *If there exists $s \in \mathcal{T}(M)$ such that $\mathsf{NF}(s) \neq 0$ then $M \Downarrow_{\mathrm{w}}$.*

Proof. We prove by induction on $k \in \mathbf{N}$ then on $M \in \Lambda_{\mathbf{S}}$ that if $\left|\mathsf{L}^k(s)\right|$ contains a normal resource term and $s \in \mathcal{T}(M)$ then $M \Downarrow_{\mathrm{w}}$.

If $M = (x)\, M_1 \cdots M_n$ we conclude directly.

If $M = \lambda x\, T$ then $s = \lambda x\, t$ with $t \in \mathcal{T}(T)$: necessarily $\left|\mathsf{L}^k(t)\right|$ contains a normal resource term and by induction hypothesis we obtain $T \Downarrow_{\mathrm{w}}$ hence $M \Downarrow_{\mathrm{w}}$.

If $M = (\lambda x\, T)\, M_0\, M_1 \cdots M_n$ then $s = \langle \lambda x\, t\rangle\, \overline{s}_0\, \overline{s}_1 \cdots \overline{s}_n$ with $t \in \mathcal{T}(T)$ and $\overline{s}_i \in \mathcal{T}(M_i)^!$ for $i \in \{0, \ldots, n\}$. Necessarily $k > 0$ and there is $s' \in |\mathsf{L}(s)| = |\langle \partial_x t \cdot \overline{s}_0\rangle\, \overline{s}_1 \cdots \overline{s}_n|$ such that $\left|\mathsf{L}^{k-1}(s')\right|$ contains a normal resource term. By Lemma 2.4.12, $s' \in \mathcal{T}((T[M_0/x])\, M_1 \cdots M_n)$: we obtain $(T[M_0/x])\, M_1 \cdots M_n \Downarrow_{\mathrm{w}}$ by induction hypothesis, and then $M \Downarrow_{\mathrm{w}}$.

If $M = a.N$, $M = N + P$ or $M = P + N$ with $s \in \mathcal{T}(N)$ then we obtain $N \Downarrow_{\mathrm{w}}$ by induction hypothesis, and then $M \Downarrow_{\mathrm{w}}$. ☐

Lemma 2.9.4. *If $M \Downarrow_{\mathrm{w}}$, then there exists $s \in \mathcal{T}(M)$ such that* $\mathsf{NF}(s) \neq 0$.

Proof. By induction on the derivation of $M \Downarrow_{\mathrm{w}}$.

If $M = (x)\, M_1 \cdots M_n$, set $s = \langle x\rangle\, [] \cdots []$ (x applied n times to the empty monomial): $s \in \mathcal{T}(M)$ and s is normal.

If $M = \lambda x\, T$ with $T \Downarrow_{\mathrm{w}}$: by induction hypothesis, we obtain $t \in \mathcal{T}(T)$ with $\mathsf{NF}(t) \neq 0$ and set $s = \lambda x\, t$.

If $M = \lambda x\, T M_0\, M_1 \cdots M_n$ and $M' = (T[M_0/x])\, M_1 \cdots M_n$ with $M' \Downarrow_{\mathrm{w}}$, the induction hypothesis gives $s' \in \mathcal{T}(M')$ such that $\mathsf{NF}(s') \neq 0$. By Lemma 2.4.12, there exist $t \in \mathcal{T}(T)$ and $\overline{u}_i \in \mathcal{T}(M_i)^!$ for $i \in \{0, \ldots, n\}$ such that $s' \in |\langle \partial_x t \cdot \overline{u}_0\rangle\, \overline{u}_1 \cdots \overline{u}_n|$. We then set $s = \langle \lambda x\, t\rangle\, \overline{u}_0 \cdots \overline{u}_n$.

If $M = a.N$, $M = N + P$ or $M = P + N$ with $N \Downarrow_{\mathrm{w}}$: the induction hypothesis gives $s \in \mathcal{T}(N) \subseteq \mathcal{T}(M)$ with $\mathsf{NF}(s) \neq 0$ directly. ☐

Taylor unsolvable terms are thus exactly those that are not weakly solvable. [25] They are moreover stable under $\simeq_\beta$:

Lemma 2.9.5. *If $M \Rightarrow_\beta M'$ then $M \Uparrow$ iff $M' \Uparrow$.*

Proof. If $\mathcal{E} \subseteq (!)\Delta$, we write $\mathsf{NF}(\mathcal{E}) := \bigcup \{|\mathsf{NF}(e)| \,;\, e \in \mathcal{E}\}$. We leave as an exercise to the reader the proof that $\mathsf{NF}(\mathcal{T}(M)) = \mathsf{NF}(\mathcal{T}(M'))$ as soon as $M \Rightarrow_\beta M'$: this is the analogue of Lemma 2.8.3 on Taylor supports (in particular there is no summability condition, and scalars play absolutely no rôle). ☐

2.9.2 Hereditarily determinable terms

The Böhm tree construction is based on the fact that, for a pure λ-term M, either M is unsolvable, or it reduces to a head normal form; and then the same holds for the arguments of the head variable. We will be able to follow a similar construction for the class of *hereditarily determinable terms*: intuitively, a simple term is in *determinate form* if it is either unsolvable or a

25. If we restrict to non-deterministic λ-terms (*i.e.* only add a sum operator to the usual λ-term constructs) then we obtain $M \Uparrow$ iff $\mathsf{NF}(\mathcal{T}(M)) = \emptyset$, which states the adequacy of $\mathsf{NF}(\mathcal{T}(\cdot))$ for the observational equivalence associated with may-style head normalization.

head normal form; and a term is hereditarily determinable if it reduces to a sum of determinate forms, and this holds hereditarily in the arguments of head variables. Formally:

Definition 2.9.6. Let $M \in \Lambda_{\mathbf{S}}$ be an algebraic λ-term. We say M is *d-determinable* if the judgement $M \Downarrow_d$ can be derived inductively from the following rules:

$$\frac{}{M \Downarrow_0} \qquad \frac{M \Uparrow}{M \Downarrow_d} \qquad \frac{S \Downarrow_d}{\lambda x\, S \Downarrow_d} \qquad \frac{M_1 \Downarrow_d \quad \cdots \quad M_n \Downarrow_d}{(x)\, M_1 \cdots M_n \Downarrow_{d+1}}$$

$$\frac{M \Downarrow_d}{a.M \Downarrow_d} \qquad \frac{M \Downarrow_d \quad N \Downarrow_d}{M + N \Downarrow_d} \qquad \frac{(S[M_0/x])\, M_1 \cdots M_n \Downarrow_d}{(\lambda x\, S)\, M_0\, M_1 \cdots M_n \Downarrow_d}$$

We say M is *hereditarily determinable* and write $M \Downarrow_\omega$ if $M \Downarrow_d$ for all $d \in \mathbf{N}$. We say M is in *d-determinate form* and write $M\ \mathsf{df}_d$ if $M \Downarrow_d$ is derivable from the above rules excluding the last one.

It should be clear that $M \Downarrow$ implies $M \Downarrow_\omega$. Observing that $M \Uparrow$ for all unsolvable pure λ-terms (*i.e.* those pure λ-terms having no head normal form), we moreover obtain $M \Downarrow_\omega$ for all $M \in \Lambda$.

We can already prove that hereditarily determinable terms are Taylor normalizable: [26]

Lemma 2.9.7. *If $M \Downarrow_d$ then $\mathcal{T}(M) \in \mathfrak{N}_d$. If moreover $M \Downarrow_\omega$ then $\mathcal{T}(M) \in \mathfrak{N}$.*

Proof. The second fact follows directly from the first one, which we prove by induction on the derivation of $M \Downarrow_d$: we use the definition of $M \Uparrow$ for the base case, and rely on Lemma 2.8.16, Corollary 2.8.19, Lemma 2.8.20, or the fact that $\mathfrak{N}_d$ is a resource structure to establish the induction in the other cases. □

On the other hand, there are Taylor normalizable terms that do not follow this pattern: intuitively, hereditarily determinable terms rule out any representation of an infinite sum of head normal forms, whereas Taylor normalizability allows to represent an infinite sum of normal forms as long as their Taylor expansions are pairwise disjoint. More formally:

Example 2.9.8. *Write $s_0 := \lambda x\, x$, and $s_{n+1} := \lambda x\, s_n$. Let $M_{step} = \lambda y\, \lambda z\, z + \lambda y\, \lambda z\, \lambda x\, (y)\, y\, z$ and then $M_{loop} = (M_{step})\, M_{step}\, \lambda x\, x$. Write $u = \lambda y\, \lambda z\, z$ and $v_{n,k} = \lambda y\, \lambda z\, \lambda x\, \langle y \rangle\, y^n\, z^k$ so that $\mathcal{T}(M_{step}) = \{u\} \cup \{v_{n,k}\ ;\ n, k \in \mathbf{N}\}$. Let $s \in \mathcal{T}(M_{loop})$ be such that $\mathsf{NF}(s) \neq 0$: a simple inspection shows that either $s = \langle u \rangle\, []\, [s_0]$ and then $\mathsf{NF}(s) = s_0$, or $s = \langle v_{n,1} \rangle\, [v_{0,1}, \ldots, v_{n-1,1}, u]\, [s_0]$ and then $\mathsf{NF}(s) = s_{n+1}$. It follows that M_{loop} is Taylor normalizable. On the other hand, observe that $\mathsf{L}^2(M_{loop}) = \lambda x\, x + \lambda x\, M_{loop}$, which is not 1-determinate: hence no $\mathsf{L}^{2k}(M_{loop})$ is 1-determinate and it will follow from Lemma 2.9.10 that M_{loop} is not 1-determinable.*

Hence hereditarily determinable terms form a strict subclass of Taylor normalizable terms, containing both pure λ-terms and normalizable algebraic λ-terms. For each level $d \in \mathbf{N}$, the class of d-determinable terms (resp. of d-determinate terms) is moreover stable under left reduction:

26. Observe that this fails if we replace $\mathcal{T}(M)$ with $|\tau(M)|$ in the definition of $M \Uparrow$: write $I := \lambda x\, x$ and consider, *e.g.*, $M = (\lambda x\, (I)\, (x + (-1).\infty_y))\, \infty_y$ which head-reduces to $(I)\, (\infty_y + (-1).\infty_y) \simeq_\tau (I)\, 0$, with $(I)\, 0 \Uparrow$ but of course $\tau(M) \notin \mathfrak{N}_0$. The very same problem would occur if we were to consider terms up to $\simeq_{\mathrm{v}}$.

Lemma 2.9.9. *If $M \Downarrow_d$ (resp. $M\ \mathsf{df}_d$) then $\mathsf{L}(M) \Downarrow_d$ (resp. $\mathsf{L}(M)\ \mathsf{df}_d$).*

Proof. We give the proof for d-determinable terms, by induction on the derivation of $M \Downarrow_d$: the case of d-determinate terms is similar, except that we do not consider head redexes.

If $d = 0$ the result is direct. Otherwise, write $d = d' + 1$.

If $M \Uparrow$ then $\mathsf{L}(M) \Uparrow$ by Lemma 2.9.5, and we conclude directly.

If $M = \lambda x\, S$ with $S \Downarrow_d$: by induction hypothesis $\mathsf{L}(S) \Downarrow_d$, and then $\lambda x\, \mathsf{L}(S) \Downarrow_d$.

If $M = (x)\, M_1 \cdots M_n$ with $M_i \Downarrow_{d'}$ for $i \in \{1, \ldots, n\}$: by induction hypothesis $\mathsf{L}(M_i) \Downarrow_{d'}$ for $i \in \{1, \ldots, n\}$, and then $(x)\, \mathsf{L}(M_1) \cdots \mathsf{L}(M_n) \Downarrow_d$.

If $M = (\lambda x\, S)\, M_0\, M_1 \cdots M_n$ with $(S[M_0/x])\, M_1 \cdots M_n \Downarrow_d$ then we conclude directly since $\mathsf{L}(M) = (S[M_0/x])\, M_1 \cdots M_n$.

If $M = a.N$ with $N \Downarrow_d$: by induction hypothesis $\mathsf{L}(N) \Downarrow_d$, and then $a.\mathsf{L}(N) \Downarrow_d$.

If $M = N + P$ with $N \Downarrow_d$ and $P \Downarrow_d$: by induction hypothesis $\mathsf{L}(N) \Downarrow_d$ and $\mathsf{L}(P) \Downarrow_d$, and then $\mathsf{L}(N) + \mathsf{L}(P) \Downarrow_d$. □

Now we can formally prove that applying the parallel left reduction strategy to d-determinable terms does reach d-determinate forms.

Lemma 2.9.10. *If $M \Downarrow_d$ then there exists $k \in \mathbf{N}$ such that $\mathsf{L}^k(M)\ \mathsf{df}_d$.*

Proof. By induction on the derivation of $M \Downarrow_d$.

If $d = 0$ or $M \Uparrow$, then $M\ \mathsf{df}_d$.

If $M = \lambda x\, S$ with $S \Downarrow_d$: by induction hypothesis, we have $k \in \mathbf{N}$ such that $\mathsf{L}^k(S)\ \mathsf{df}_d$ and then $\mathsf{L}^k(M) = \lambda x\, \mathsf{L}^k(S)$ hence $\mathsf{L}^k(M)\ \mathsf{df}_d$.

If $M = (x)\, M_1 \cdots M_n$ with $d > 0$ and $M_i \Downarrow_{d-1}$ for each $i \in \{1, \ldots, n\}$: by induction hypothesis, we obtain $k_i \in \mathbf{N}$ such that $\mathsf{L}^{k_i}(M_i)\ \mathsf{df}_{d-1}$ for each $i \in \{1, \ldots, n\}$. Let $k = \max \{k_i\ ;\ 1 \leq i \leq n\}$: by Lemma 2.9.9, we also have $\mathsf{L}^k(M_i)\ \mathsf{df}_{d-1}$ for all $i \in \{1, \ldots, n\}$. Since $\mathsf{L}^k(M) = \mathsf{L}^k((x)\, M_1 \cdots M_n) = (x)\, \mathsf{L}^k(M_1) \cdots \mathsf{L}^k(M_n)$ we conclude that $\mathsf{L}^k(M)\ \mathsf{df}_d$.

If $M = (\lambda x\, S)\, M_0\, M_1 \cdots M_n$ with $(S[M_0/x])\, M_1 \cdots M_n \Downarrow_d$: by induction hypothesis, we have $k_0 \in \mathbf{N}$ such that $\mathsf{L}^{k_0}((S[M_0/x])\, M_1 \cdots M_n)\ \mathsf{df}_d$. It is then sufficient to observe that $\mathsf{L}(M) = (S[M_0/x])\, M_1 \cdots M_n$ and set $k = k_0 + 1$.

If $M = a.N$ with $N \Downarrow_d$: by induction hypothesis, we have $k \in \mathbf{N}$ such that $\mathsf{L}^k(S)\ \mathsf{df}_d$ and then $\mathsf{L}^k(M) = a.\mathsf{L}^k(S)$ hence $\mathsf{L}^k(M)\ \mathsf{df}_d$.

If $M = N + P$ with $N \Downarrow_d$ and $P \Downarrow_d$: by induction hypothesis, we have $k_0, k_1 \in \mathbf{N}$ such that $\mathsf{L}^{k_0}(N)\ \mathsf{df}_d$ and $\mathsf{L}^{k_1}(P)\ \mathsf{df}_d$ and then, setting $k = \max(k_0, k_1)$, $\mathsf{L}^k(M) = \mathsf{L}^k(N) + \mathsf{L}^k(P)$ hence $\mathsf{L}^k(M)\ \mathsf{df}_d$ by the previous lemma. □

2.9.3 Approximants of the normal form of Taylor expansion

Now we introduce the analogue of finite Böhm trees for hereditarily determinable terms:

Definition 2.9.11. If $M \Downarrow_d$ then we define the *normal d-approximant* $\mathsf{NA}_d(M)$ of M inductively as follows: $\mathsf{NA}_d(M) := 0$ if $d = 0$ or $M \Uparrow$, and

$$\begin{aligned}
\mathsf{NA}_d(\lambda x\, S) &:= \lambda x\, \mathsf{NA}_d(S)\\
\mathsf{NA}_d((x)\, M_1 \cdots M_n) &:= (x)\, \mathsf{NA}_{d-1}(M_1) \cdots \mathsf{NA}_{d-1}(M_n)\\
\mathsf{NA}_d((\lambda x\, S)\, M_0\, M_1 \cdots M_n) &:= \mathsf{NA}_d((S[M_0/x])\, M_1 \cdots M_n)\\
\mathsf{NA}_d(a.M) &:= a.\mathsf{NA}_d(M)\\
\mathsf{NA}_d(M + N) &:= \mathsf{NA}_d(M) + \mathsf{NA}_d(N)
\end{aligned}$$

otherwise.

First observe that d approximants are stable under parallel left reduction:

Lemma 2.9.12. *If $M \Downarrow_d$ then $\mathsf{NA}_d(M) = \mathsf{NA}_d(\mathsf{L}(M))$.*

Proof. Recall indeed that, by Lemma 2.9.9, $\mathsf{L}(M) \Downarrow_d$ so that $\mathsf{NA}_d(\mathsf{L}(M))$ is well defined. The proof is then straightforward, by induction on $M \Downarrow_d$. □

We do not prove here that d-determinable terms and the associated d-approximants are stable under arbitrary reduction: if $M \Downarrow_d$ and $M \Rightarrow_\beta M'$ then $M' \Downarrow_d$ and then $\mathsf{NA}_d(M) = \mathsf{NA}_d(M')$. We believe it is a very solid conjecture, but it would require us to develop a full standardization argument: in our non-deterministic setting, this is known to be tedious at best [23; 24]. Since we introduced hereditarily determinable terms *ad-hoc*, only to be able to define normal d-approximants, we feel that the general study of their computational behaviour is not worth the effort.

Our next step is to show that if M is in $d + 1$ determinate form, then $\tau(M)\restriction_{\mathcal{N}_d}$ depends only on $\mathsf{NA}_{d+1}(M)$.

Lemma 2.9.13. *If $M\, \mathsf{df}_{d+1}$ then, for all $s \in \mathcal{N}_d$, $\tau(M)_s = \tau(\mathsf{NA}_{d+1}(M))_s$.*

Proof. By induction on the derivation of $M\, \mathsf{df}_{d+1}$, writing $M' = \mathsf{NA}_{d+1}(M)$.

If $M \Uparrow$ then $M' = 0$ and $\tau(M)_s = 0$ for all $s \in \mathcal{N}$, hence the result holds.

If $M = \lambda x\, T$ with $T\, \mathsf{df}_{d+1}$ then $M' = \lambda x\, \mathsf{NA}_{d+1}(T)$ and we can assume $s = \lambda x\, t$: otherwise $\tau(M)_s = 0 = \tau(M')_s$. Then $t \in \mathcal{N}_d$ and by induction hypothesis $\tau(M)_s = \tau(T)_t = \tau(\mathsf{NA}_{d+1}(T))_t = \tau(M')_s$.

If $M = (x)\, N_1 \cdots N_n$ with $N_i\, \mathsf{df}_d$ for all $i \in \{1, \dots, n\}$ then $M' = (x)\, N'_1 \cdots N'_n$ with $N'_i = \mathsf{NA}_d(N_i)$ and we can assume $s = \langle x\rangle\, \bar{t}_1 \cdots \bar{t}_n$: otherwise $\tau(M)_s = 0 = \tau(M')_s$. If $d = 0$, $s \in \mathcal{N}_0$, hence $n = 0$ and then $M = x = M'$. Otherwise write $d = d' + 1$. For each $i \in \{1, \dots, n\}$, $|\bar{t}_i| \subseteq \mathcal{N}_{d'}$. By induction hypothesis we obtain $\tau(N_i)_u = \tau(N'_i)_u$ for all $u \in |\bar{t}_i|$: it follows that $\tau(N_i)^!_{\bar{t}_i} = \tau(N'_i)^!_{\bar{t}_i}$ by the definition of promotion. Then $\tau(M)_s = \prod_{i=1}^n \tau(N_i)_{\bar{t}_i} = \prod_{i=1}^n \tau(N'_i)_{\bar{t}_i} = \tau(M')_s$.

If $M = a.N$ with $N\, \mathsf{df}_{d+1}$ then $\tau(M)_s = a.\tau(N)_s = a.\tau(\mathsf{NA}_{d+1}(N))_s = \tau(M')_s$ by induction hypothesis.

Similarly, if $M = N + P$ with $N\, \mathsf{df}_{d+1}$ and $P\, \mathsf{df}_{d+1}$ then $\tau(M)_s = \tau(N)_s + \tau(P)_s = \tau(\mathsf{NA}_{d+1}(N))_s + \tau(\mathsf{NA}_{d+1}(P))_s = \tau(M')_s$ by induction hypothesis. □

We obtain our final theorem:

Theorem 2.9.14. *For all hereditarily determinable term M, the sequence $(\tau(\mathsf{NA}_d(M)))_{d\in\mathbf{N}}$ of normal vectors converges to $\mathsf{NF}(\tau(M))$ in $\mathbf{S}^{\mathcal{N}}$.*

Proof. First observe that each $\tau(\mathsf{NA}_d(M)) \in \mathbf{S}^{\mathcal{N}}$, because $\mathsf{NA}_d(M)$ is in normal form. Let $s \in \mathcal{N}$ and fix $d \geq \mathbf{d}(s) + 1$: by Lemmas 2.9.9, 2.9.10 and 2.9.12, there exists $k_0 \in \mathbf{N}$ such that $\mathsf{L}^k(M)\,\mathsf{df}_d$ and $\mathsf{NA}_d\big(\mathsf{L}^k(M)\big) = \mathsf{NA}_d(M)$ whenever $k \geq k_0$. By Lemma 2.9.13, we moreover have $\tau(\mathsf{NA}_d(M))_s = \tau\big(\mathsf{NA}_d\big(\mathsf{L}^k(M)\big)\big)_s = \tau\big(\mathsf{L}^k(M)\big)_s$. It follows that $\tau(\mathsf{NA}_d(M))_s = \mathsf{NF}(\tau(M))_s$, by Theorem 2.8.14. Since this holds for any $d \geq \mathbf{d}(s) + 1$, we have just proved that $(\tau(\mathsf{NA}_d(M))_s)_{d\in\mathbf{N}}$ converges to $\mathsf{NF}(\tau(M))_s$, for the discrete topology. □

In the case of pure λ-terms, by identifying 0 with the unsolvable Böhm tree $\perp$, it should be clear that the sequence $(\mathsf{NA}_d(M))_{d\in\mathbf{N}}$ is nothing but the increasing sequence of finite approximants of $\mathsf{BT}(M)$: Theorem 2.9.14 is thus a proper generalization of Theorem 2.8.24 of which it provides a new proof.

Chapter 3

An application of parallel cut elimination in multiplicative linear logic to the Taylor expansion of proof nets

This chapter is essentially the inclusion of the article of the same name [6], co-authored with Jules Chouquet. This much expanded version of our CSL 2018 contribution [14] was accepted for publication in Logical Methods in Computer Science.

Abstract: We examine some combinatorial properties of parallel cut elimination in multiplicative linear logic (MLL) proof nets. We show that, provided we impose a constraint on some paths, we can bound the size of all the nets satisfying this constraint and reducing to a fixed resultant net. This result gives a sufficient condition for an infinite weighted sum of nets to reduce into another sum of nets, while keeping coefficients finite. We moreover show that our constraints are stable under reduction.

Our approach is motivated by the quantitative semantics of linear logic: many models have been proposed, whose structure reflect the Taylor expansion of multiplicative exponential linear logic (MELL) proof nets into infinite sums of differential nets. In order to simulate one cut elimination step in MELL, it is necessary to reduce an arbitrary number of cuts in the differential nets of its Taylor expansion. It turns out our results apply to differential nets, because their cut elimination is essentially multiplicative. We moreover show that the set of differential nets that occur in the Taylor expansion of an MELL net automatically satisfies our constraints.

Interestingly, our nets are untyped: we only rely on the sequentiality of linear logic nets and the dynamics of cut elimination. The paths on which we impose bounds are the switching paths involved in the Danos–Regnier criterion for sequentiality. In order to accommodate multiplicative units and weakenings, our nets come equipped with jumps: each weakening node is connected to some other node. Our constraint can then be summed up as a bound on both the length of switching paths, and the number of weakenings that jump to a common node.

3.1 Introduction

3.1.1 Context: quantitative semantics and Taylor expansion

Linear logic takes its roots in the denotational semantics of λ-calculus: it is often presented, by Girard himself [Gir87], as the result of a careful investigation of the model of coherence spaces. Since its early days, linear logic has thus generated a rich ecosystem of denotational models, among which we distinguish the family of *quantitative semantics*. Indeed, the first ideas behind linear logic were exposed even before coherence spaces, in the model of normal functors [Gir88], in which Girard proposed to consider analyticity, instead of mere continuity, as the key property of the interpretation of λ-terms: in this setting, terms denote power series, representing analytic maps between modules.

This quantitative interpretation reflects precise operational properties of programs: the degree of a monomial in a power series is closely related to the number of times a function uses its argument. Following this framework, various models were considered — among which we shall include the multiset relational model as a degenerate, boolean-valued instance. These models allowed to represent and characterize quantitative properties such as the execution time [Car18a], including best and worst case analysis for non-deterministic programs [Lai+13], or the probability of reaching a value [DE11]. It is notable that this whole approach gained momentum in the early 2000's, after the introduction by Ehrhard of models [Ehr02; Ehr05] in which the notion of analytic maps interpreting λ-terms took its usual sense, while Girard's original model involved set-valued formal power series. Indeed, the keystone in the success of this line of work is an analogue of the Taylor expansion formula, that can be established both for λ-terms and for linear logic proofs.

Mimicking this denotational structure, Ehrhard and Regnier introduced the differential λ-calculus [ER03] and differential linear logic [ER06b], which allow to formulate a syntactic version of Taylor expansion: to a λ-term (resp. to a linear logic proof), we associate an infinite linear combination of approximants [ER08; Ehr18]. In particular, the dynamics (*i.e.* β-reduction or cut elimination) of those systems is dictated by the identities of quantitative semantics. In turn, Taylor expansion has become a useful device to design and study new models of linear logic, in which morphisms admit a matrix representation: the Taylor expansion formula allows to describe the interpretation of promotion — the operation by which a linear resource becomes freely duplicable — in an explicit, systematic manner. It is in fact possible to show that any model of differential linear logic without promotion gives rise to a model of full linear logic in this way [Car07]: in some sense, one can simulate cut elimination through Taylor expansion.

3.1.2 Motivation: reduction in Taylor expansion

There is a difficulty, however: Taylor expansion generates infinite sums and, *a priori*, there is no guarantee that the coefficients in these sums will remain finite under reduction. In previous works [Car07; Lai+13], coefficients were thus required to be taken in a complete semiring: all sums should converge. In order to illustrate this requirement, let us first consider the case of λ-calculus.

The linear fragment of differential λ-calculus, called *resource λ-calculus*, is the target

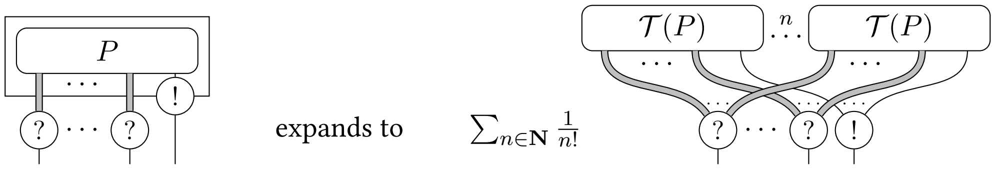

Figure 3.1 – Taylor expansion of a promotion box (thick wires denote an arbitrary number of wires)

of the syntactic Taylor expansion of λ-terms. In this calculus, the application of a term to another is replaced with a multilinear variant: $\langle s\rangle[t_1,\dots,t_n]$ denotes the n-linear symmetric application of resource term s to the multiset of resource terms $[t_1,\dots,t_n]$. Then, if $x_1,\dots,x_k$ denote the occurrences of x in s, the redex $\langle\lambda x.s\rangle[t_1,\dots,t_n]$ reduces to the sum $\sum_{f:\{1,\dots,k\}\tilde{\to}\{1,\dots,n\}} s[t_{f(1)}/x_1,\dots,t_{f(k)}/x_k]$: here f ranges over all bijections $\{1,\dots,k\}\tilde{\to}\{1,\dots,n\}$ so this sum is zero if $n\neq k$. As sums are generated by reduction, it should be noted that all the syntactic constructs are linear, both in the sense that they commute to sums, and in the sense that, in the elimination of a redex, no subterm of the argument multiset is copied nor erased. The key case of Taylor expansion is that of application:

$$\mathcal{T}(MN)=\sum_{n\in N}\frac{1}{n!}\langle\mathcal{T}(M)\rangle\mathcal{T}(N)^n \tag{3.1}$$

where $\mathcal{T}(N)^n$ is the multiset made of n copies of $\mathcal{T}(N)$ — by n-linearity, $\mathcal{T}(N)^n$ is itself an infinite linear combination of multisets of resource terms appearing in $\mathcal{T}(N)$. Admitting that $\langle M\rangle[N_1,\dots,N_n]$ represents the n-th derivative of M, computed at 0, and n-linearly applied to N_1, …, N_n, one immediately recognizes the usual Taylor expansion formula.

From (3.1), it is immediately clear that, to simulate one reduction step occurring in N, it is necessary to reduce in parallel in an unbounded number of subterms of each component of the expansion. Unrestricted parallel reduction, however, is ill defined in this setting. Consider the sum $\sum_{n\in\mathbf{N}}\langle\lambda xx\rangle[\cdots\langle\lambda xx\rangle[y]\cdots]$ where each summand consists of n successive linear applications of the identity to the variable y: then by simultaneous reduction of all redexes in each component, each summand yields y, so the result should be $\sum_{n\in\mathbf{N}} y$ which is not defined unless the semiring of coefficients is complete in some sense.

Those considerations apply to linear logic as well as to λ-calculus. We will use proof nets [Gir87] as the syntax for proofs of multiplicative exponential linear logic (MELL). The target of Taylor expansion is then in promotion-free differential nets [ER06b], which we call *resource nets* in the following, by analogy with the resource λ-calculus: these form the multilinear fragment of differential linear logic.

In linear logic, Taylor expansion consists in replacing duplicable subnets, embodied by promotion boxes, with explicit copies, as in Figure 3.1: if we take n copies of the box, the main port of the box is replaced with an n-ary !-link, while the ?-links at the border of the box collect all copies of the corresponding auxiliary ports. Again, to follow a single cut elimination step in P, it is necessary to reduce an arbitrary number of copies. And unrestricted parallel cut elimination in an infinite sum of resource nets is broken, as one can easily construct an infinite

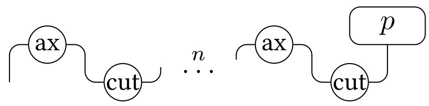

Figure 3.2 – Example of a family of nets, all reducing to a single net p, by the parallel elimination of axiom cuts.

family of nets, all reducing to the same resource net p in a single step of parallel cut elimination: see Figure 3.2.

3.1.3 Our approach: taming the combinatorial explosion of antireduction

The problem of convergence of series of linear approximants under reduction was first tackled by Ehrhard and Regnier, for the normalization of Taylor expansion of ordinary λ-terms [ER08]. Their argument relies on a uniformity property, specific to the pure λ-calculus: the support of the Taylor expansion of a λ-term forms a clique in some fixed coherence space of resource terms. This method cannot be adapted to proof nets: there is no coherence relation on differential nets such that all supports of Taylor expansions are cliques [Tas09, Section V.4.1].

An alternative method to ensure convergence without any uniformity hypothesis was first developed by Ehrhard for typed terms in a λ-calculus extended with linear combinations of terms [Ehr10]: there, the presence of sums also forbade the existence of a suitable coherence relation. This method can be generalized to strongly normalizable [12], or even weakly normalizable [13] terms. One striking feature of this approach is that it concentrates on the support (*i.e.* the set of terms having non-zero coefficients) of the Taylor expansion. In each case, one shows that, given a normal resource term t and a λ-term M, there are finitely many terms s, such that:

- the coefficient of s in $\mathcal{T}(M)$ is non zero; and
- the coefficient of t in the normal form of s is non zero.

This allows to normalize the Taylor expansion: simply normalize in each component, then compute the sum, which is component-wise finite.

The second author then remarked that the same could be done for β-reduction [13], even without any uniformity, typing or normalizability requirement. Indeed, writing $s \rightrightarrows t$ if s and t are resource terms such that t appears in the support of a parallel reduct of s, the size of s is bounded by a function of the size of t *and* the height of s. So, given that if s appears in $\mathcal{T}(M)$ then its height is bounded by that of M, it follows that, for a fixed resource term t there are finitely many terms s in the support of $\mathcal{T}(M)$ such that $s \rightrightarrows t$: in short, parallel reduction is always well-defined on the Taylor expansion of a λ-term.

Our purpose in the present paper is to develop a similar technique for MELL proof nets: we show that one can bound the size of a resource net p by a function of the size of any of its parallel reducts, and of an additional quantity on p, yet to be defined. The main challenge is indeed to circumvent the lack of inductive structure in proof nets: in such a graphical syntax, there is no structural notion of height.

We claim that a side condition on switching paths, *i.e.* paths in the sense of Danos–Regnier's correctness criterion [DR89], is an appropriate replacement. Backing this claim, there are first

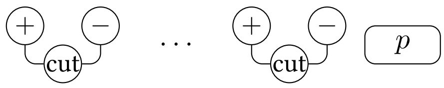

Figure 3.3 – Evanescent cuts: here each $(+)$ node can denote a tensor unit $\mathbf{1}$ or a coweakening (a nullary !-link), and then the corresponding $(-)$ node should be the dual unit $\perp$ or a weakening (a nullary $?$-link). Then the depicted net reduces to p in one parallel cut elimination step.

some intuitions:

- the main culprits for the unbounded loss of size in reduction are the chains of consecutive cuts, as in Figure 3.2;
- we want the validity of our side condition to be stable under reduction so, rather than chains of cuts, we should consider the length of switching paths;
- indeed, if p reduces to q *via* cut elimination, then the switching paths of q are somehow related with those of p;
- and the switching paths of a resource net in $\mathcal{T}(P)$ are somehow related with those of P.

In the following we will establish precise formulations of those last two points: we study the structure of switching paths through cut elimination in Section 3.4; and we describe the switching paths of the elements of $\mathcal{T}(P)$ in Section 3.7.

In presence of multiplicative units, or of weakenings (nullary $?$-links) and coweakenings (nullary !-links), we must also take special care of another kind of cuts, that we call *evanescent cuts*: when a cut between such nullary links is eliminated, it simply vanishes, leaving the rest of the net untouched, as in Figure 3.3, which is obviously an obstacle for our purpose. [1]

In order to deal with nullary links, a well known trick is to attach each weakening (or $\perp$-link) to another node in the net: switching paths can then follow such jumps, which is useful to characterize exactly those nets that come from proof trees [Gir96, Appendix A.2]. Here we will rely on this structure to control the effect of eliminating evanescent cuts on the size of a net.

In all our exposition, we adopt a particular presentation of nets: we consider n-ary exponential links rather than separate (co)dereliction and (co)contraction, as this allows to reduce the dynamics of resource nets to that of multiplicative linear logic (MLL) proof nets. [2]

3.1.4 Outline

In Section 3.2, we first introduce MLL proof nets formally, in the term-based syntax of Ehrhard [Ehr14]. We define the parallel cut elimination relation $\rightrightarrows$ in this setting, that we decompose into multiplicative reduction $\rightrightarrows_m$, axiom-cut reduction $\rightrightarrows_a$ and evanescent reduction

1. The treatment of weakenings is indeed the main novelty of the present extended version over our conference paper [14].

2. In other words, we adhere to a version of linear logic proof nets and resource nets which is sometimes called *nouvelle syntaxe*, although it dates back to Regnier's PhD thesis [Reg92]. For the linear logic *connoisseur*, this is already apparent in Figure 3.1. See also the discussion in our conclusion (Section 3.8).

$\rightrightarrows_e$. We also present the notion of switching path for this syntax, and introduce the two quantities that will be our main objects of study in the following:

- the maximum number $\mathbf{jd}(p)$ of $\perp$-links that jump to a common target;
- the maximum length $\mathbf{ln}(p)$ of any switching path in the net p.

Let us mention that typing plays absolutely no role in our approach, so we do not even consider formulas of linear logic in our exposition: we will rely on the geometrical structure of nets only.

We show in Section 3.3 that, if $p \rightrightarrows_m q$, $p \rightrightarrows_a q$ or $p \rightrightarrows_e q$ then the size of p is bounded by a function of $\mathbf{ln}(p)$, $\mathbf{jd}(p)$, and the size of q. In order to be able to iterate this combinatorial argument, we must show that, given bounds for $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$, we can infer bounds on $\mathbf{ln}(q)$ and $\mathbf{jd}(q)$: this is the subject of Sections 3.4 and 3.5.

Section 3.4 is dedicated to the proof that we can bound $\mathbf{ln}(q)$ by a function of $\mathbf{ln}(p)$: the main case is the multiplicative reduction, as this may create new switching paths in q that we must relate with those in p. In this task, we concentrate on the notion of *slipknot*: a pair of residuals of a cut of p occurring in a path of q. Slipknots are essential in understanding how switching paths are structured after cut elimination: this analysis is motivated by a technical requirement of our approach, but it can also be considered as a contribution to the theory of MLL nets *per se*.

In Section 3.5, we show that $\mathbf{jd}(q)$ is bounded by a function of $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$: the critical case here is that of chains of jumps between evanescent cuts.

We leverage all of the above results in Section 3.6, to generalize them to a reduction $p \rightrightarrows q$, or even an arbitrary sequence of reductions. In particular, if $p \rightrightarrows q$ then the size of p is bounded by a function of the size of q and of $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$. Again, this result is motivated by the study of quantitative semantics, but it is essentially a theorem about MLL.

We establish the applicability of our approach to the Taylor expansion of MELL proof nets in Section 3.7: we show that if p is a resource net of $\mathcal{T}(P)$, then $\mathbf{ln}(p)$ is bounded by a function of the size of P, and $\mathbf{jd}(p)$ is bounded by the size of P.

Finally, we discuss the scope of our results in the concluding Section 3.8.

3.2 Definitions

We provide here the minimal definitions necessary for us to work with MLL proof nets. As stated before, let us stress the fact that the choice of MLL is not decisive for the development of Sections 3.2 to 3.6. The reader can check that we rely on three ingredients only:

- the definition of switching paths;
- the fact that multiplicative reduction amounts to plug bijectively the premises of a $\otimes$-link with those of $\parr$-link (in the nullary case, evanescent cuts simply vanish);
- the definition of jumps and how they are affected by cut elimination.

The results of those sections are thus directly applicable to resource nets, thanks to our choice of generalized exponential links: this will be done in Section 3.7.

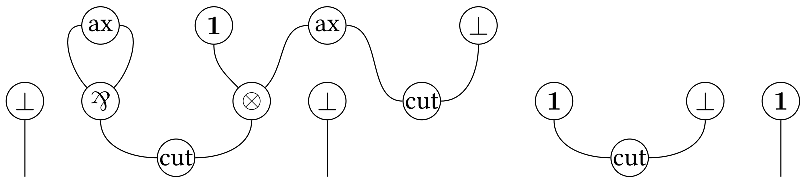

Figure 3.4 – An example of multiplicative net

3.2.1 Nets

A proof net is usually presented as a graphical object such as that of Figure 3.4. Following Ehrhard [Ehr14; Ehr18], we will rely on a term syntax for denoting such nets. This is based on a quite standard trichotomy: a proof net can be divided into a top layer of axioms, followed by trees of connectives, down to cuts between the conclusions of some trees.

We will represent the conclusions of axiom rules by variables: the duality between two conclusions of an axiom rule is given by an involution $x \mapsto \overline{x}$ over the set $\mathbf{V}$ of variables. Our nets will be finite families of trees and cuts, where trees are inductively generated from variables by the application of MLL connectives, of arbitrary arity: $\otimes(t_1, \ldots, t_n)$ and $\parr(t_1, \ldots, t_n)$. A tree thus represents a conclusion of a net, together with the nodes above it, up to axiom conclusions. A cut is then given by the pair of trees $\langle t_1 | t_2 \rangle$, whose conclusions it cuts together. In order to distinguish between various occurrences of nullary connectives $\mathbf{1} = \otimes()$ and $\bot = \parr()$, we will index them with labels taken from sets $\mathbf{U_1}$ and $\mathbf{U}_\bot$.

Formally, the set of *raw trees* (denoted by s, t, *etc.*) is generated as follows:

$$t ::= x \mid \mathbf{1}_\lambda \mid \bot_\mu \mid \otimes(t_1, \ldots, t_n) \mid \parr(t_1, \ldots, t_n)$$

where x ranges over $\mathbf{V}$, λ ranges over $\mathbf{U_1}$, μ ranges over $\mathbf{U}_\bot$ and we require $n \neq 0$ in the two last cases. We assume $\mathbf{V}$, $\mathbf{U_1}$ and $\mathbf{U}_\bot$ are pairwise disjoint and all three are denumerably infinite. We will always identify a nullary connective tree $\mathbf{1}_\lambda$ or $\bot_\mu$ with its label λ or μ, so that $\mathbf{A} = \mathbf{V} \cup \mathbf{U_1} \cup \mathbf{U}_\bot$ is just the set of *atomic trees*. We will generally use letters x, y, z for variables, μ for the elements of $\mathbf{U}_\bot$, λ for the elements of $\mathbf{U_1}$, and s, t, u, v for arbitrary raw trees.

We write $\mathbf{T}(t)$ for the set of *subtrees* of a given raw tree t, which is defined inductively in the natural way : if $t \in \mathbf{A}$, then $\mathbf{T}(t) = \{t\}$; if $t = \diamond(t_1, \ldots, t_n)$ with $\diamond \in \{\otimes, \parr\}$, then $\mathbf{T}(t) = \{t\} \cup \bigcup_{i \in \{1,\ldots,n\}} \mathbf{T}(t_i)$. We moreover write $\mathbf{V}(t)$ for $\mathbf{T}(t) \cap \mathbf{V}$, and similarly for $\mathbf{U_1}(t)$, $\mathbf{U}_\bot(t)$ and $\mathbf{A}(t)$. A *tree* is then a raw tree t such that if $\diamond(t_1, \ldots, t_n) \in \mathbf{T}(t)$ then the sets $\mathbf{A}(t_i)$ for $1 \leq i \leq n$ are pairwise disjoint: in other words, each atom occurs at most once in t. As a consequence, each subtree $u \in \mathbf{T}(t)$ occurs exactly once in a tree t.

A *cut* is an unordered pair $c = \langle t | s \rangle$ of trees such that $\mathbf{A}(t) \cap \mathbf{A}(s) = \emptyset$, and then we set $\mathbf{T}(c) = \mathbf{T}(t) \cup \mathbf{T}(s)$, and similarly for $\mathbf{V}(c)$, $\mathbf{U_1}(c)$, $\mathbf{U}_\bot(c)$ and $\mathbf{A}(c)$. Note that, in the absence of typing, we do not put any compatibility requirement on cut trees.

Given a set A, we denote by $\overrightarrow{a}$ any finite family $(a_i)_{i \in I} \in A^I$ of elements of A. In general, we abusively identify $\overrightarrow{a}$ with any enumeration $(a_1, \ldots, a_n) \in A^n$ of its elements, and we may even write $\overrightarrow{a} = a_1, \ldots, a_n$ in this case; moreover, we simply write $\overrightarrow{a}, \overrightarrow{b}$ for the concatenation of families $\overrightarrow{a}$ and $\overrightarrow{b}$ (whose index set is implicitly the sum of the index sets of $\overrightarrow{a}$ and $\overrightarrow{b}$). We may also write, e.g., $a_i \in \overrightarrow{a}$, identifying the family $\overrightarrow{a}$ with its support set. Since we only consider families of pairwise distinct elements, such abuse of notation is generally harmless: in this case, the only difference between $(a_i)_{i\in I}$ and its support set $\{a_i \mid i \in I\}$ is whether the bijection $i \mapsto a_i$ is part of the data or not. If f is a function from A to any powerset, we extend it to families in the obvious way, setting $f(\overrightarrow{a}) = \bigcup_{a\in\overrightarrow{a}} f(a)$. E.g., if $\overrightarrow{\gamma}$ is a family of trees or cuts we write $\mathbf{V}(\overrightarrow{\gamma}) = \bigcup_{\gamma\in\overrightarrow{\gamma}} \mathbf{V}(\gamma)$.

An MLL *bare proof net* is a pair $p = (\overrightarrow{c}; \overrightarrow{t})$ of a finite family $\overrightarrow{c}$ of pairwise distinct cuts and a finite family $\overrightarrow{t}$ of pairwise distinct trees such that: for all distinct cuts or trees $\gamma, \gamma' \in \overrightarrow{c} \cup \overrightarrow{t}$, $\mathbf{A}(\gamma) \cap \mathbf{A}(\gamma') = \emptyset$; and $\mathbf{V}(p) = \mathbf{V}(\overrightarrow{c}) \cup \mathbf{V}(\overrightarrow{t})$ is closed under the involution $x \mapsto \overline{x}$. We write $\mathbf{C}(p) = \overrightarrow{c}$ for the family of cuts of p. For any tree, cut or bare proof net γ, we define the *size* of γ as $\mathbf{size}(\gamma) = \#\mathbf{T}(\gamma)$: graphically, $\mathbf{size}(p)$ is nothing but the number of wires in p.

Remark 3.2.1. *In a graphical structure such as that of Figure 3.4, the interface (*i.e. *the set of extremities of dangling wires, which represent the conclusions of the net) is relevant: in particular, cut elimination preserves this interface. So, in $p = (\overrightarrow{c}; \overrightarrow{t})$, $\overrightarrow{t}$ is intrinsically a family, whose index set is precisely the interface of the structure.*

On the other hand, the rest of the net should be considered up to isomorphism: in our case, this amounts to the reindexing of cuts, and the renaming of atoms, preserving the duality involution on variables. We may call α-equivalence the corresponding equivalence relation on bare proof nets, as it has the very same status as the renaming of bound variables in the ordinary λ-calculus. In particular, $\overrightarrow{c}$ should be considered as a set, although we introduce it as a family here, just because it will be convenient to treat the concatenation $\overrightarrow{c}, \overrightarrow{t}$ as a family of cuts and trees in the following.

The reader may check that bare proof nets quotiented by α-equivalence, as introduced above, are exactly the usual (untyped) proof structures for MLL (with connectives of arbitrary arity). We keep this quotient implicit whenever possible in the remaining: in any case, α-equivalence preserves the size of nets, as well as the length of paths to be introduced later. [3]

As announced in our introduction, our nets will be equipped with jumps from $\perp$ nodes to other nodes. An MLL *proof net* will thus be the data of a bare proof net p and of a *jump function* $\jmath : \mathbf{U}_\perp(p) \to \mathbf{T}(p)$. We will often identify a proof net with its underlying bare net p, and then write $\jmath_p$ for the associated jump function. Figure 3.5 presents such a net, whose underlying graphical structure is that of Figure 3.4.

We can already introduce the first of our two key quantities: the *jump degree* $\mathbf{jd}(p)$ of a net p. We first define the jump degree of any tree $t \in \mathbf{T}(p)$, setting $\mathbf{jd}_p(t) = \#\{\mu \in \mathbf{U}_\perp(p) \mid \jmath_p(\mu) = t\}$. We will often write $\mathbf{jd}(t)$ instead of $\mathbf{jd}_p(t)$ if p is clear from the context. Then we set $\mathbf{jd}(p) = \max\{\mathbf{jd}(t) \mid t \in \mathbf{T}(p)\}$.

Remark 3.2.2. *Originally, jumps were introduced as* pis aller *for the characterization of sequentializable proof nets [Gir96, Appendix A.2]. Indeed, in presence of multiplicative units and*

3. Note that this situation differs slightly from the case of interaction nets [Laf90], where explicit axioms and cuts links are missing and there is no top-down orientation *a priori*. Term syntaxes have been proposed for those [MS08; FM99, among others] but the correspondence is less immediate: it must be restricted to deadlock-free interaction nets and, in addition to α-equivalence, one must introduce some mechanism to deal with implicit axiom-cut elimination in the application of reduction rules.

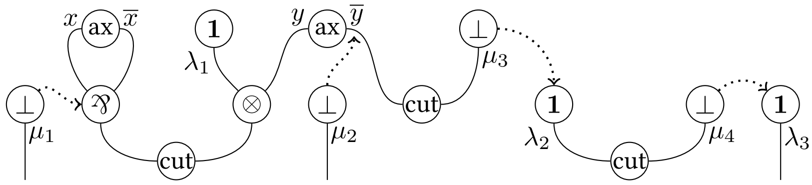

Figure 3.5 – The net $p_0 = (\langle ⅋(x, \overline{x}) | \otimes(\mathbf{1}_{\lambda_1}, y)\rangle, \langle \overline{y} | \perp_{\mu_3}\rangle, \langle \mathbf{1}_{\lambda_2} | \perp_{\mu_4}\rangle; \perp_{\mu_1}, \perp_{\mu_2}, \mathbf{1}_{\lambda_3})$ with $\jmath_{p_0} : \mu_1 \mapsto ⅋(x, \overline{x}), \mu_2 \mapsto \overline{y}, \mu_3 \mapsto \lambda_2, \mu_4 \mapsto \lambda_3$.

without jumps, Danos–Regnier's correctness criterion, requiring the connectedness and acyclicity of switching graphs, fails to accept some proof nets corresponding to proof trees using the $\perp$-rule. So, to characterize all sequentializable nets, one has to require the existence of a jump function that makes all the switching graphs connected and acyclic. This additional structure is somewhat arbitrary, and it restores a form of bureaucratic sequentiality: distinct jumping functions on the same bare net may yield equivalent sequentializations. And there is no satisfactory solution to that issue: if a notion of proof net could capture proof equivalence in MLL with units, then deciding the identity of proof nets in that setting would not be tractable, simply because proof equivalence is PSPACE-complete [HH16].

A simple, consensual alternative is to forget about jumps and drop the connectedness requirement: the acyclicity criterion characterizes exactly those nets that are sequentializable using an additional mix-rule, corresponding to the parallel juxtaposition of nets. In particular, this weaker requirement is sufficient to avoid the problematic cases of cut elimination. This is the approach we adopt: after defining switchings and paths in Subsection 3.2.3, we will restrict our attention to acyclic nets only.

We associate jump functions with nets nonetheless, but for a different purpose: bounding the jump degree in a net will allow us to control the combinatorics of the elimination of evanescent cuts, in situations such as that of Figure 3.3. Since switching paths can follow jumps, and those paths will be our main focus throughout the remaining of the paper, we chose to consider proof nets as equipped with jumps by default. Still, the reader should be aware that the main subject of interest is the underlying structure of bare proof nets.

3.2.2 Cut elimination

A *reducible cut* is a cut $\langle t|s\rangle$ such that:

- t is a variable and $\overline{t} \notin \mathbf{V}(s)$ (*axiom cut*);
- or $t \in \mathbf{U}_\mathbf{1}$ and $s \in \mathbf{U}_\perp$, and $\jmath(s) \notin \{t, s\}$ (*evanescent cut*);
- or we can write $t = \otimes(t_1, \ldots, t_n)$ and $s = ⅋(s_1, \ldots, s_n)$ (*multiplicative cut*).

The *substitution* $\gamma[t/x]$ of a tree t for a variable x in a tree (or cut, or family of trees and/or cuts) γ is defined in the usual way, with the additional assumption that $\mathbf{A}(t)$ and $\mathbf{A}(\gamma)$ are disjoint. By the definition of trees, this substitution is essentially linear: each variable x appears at most once in γ.

There are three basic cut elimination steps defined for bare proof nets, one for each kind of reducible cut:

- the elimination of a multiplicative cut yields a family of cuts: we write

$$\langle\otimes(t_1,\ldots,t_n)|\parr(s_1,\ldots,s_n)\rangle \to_m \langle t_1|s_1\rangle,\ldots,\langle t_n|s_n\rangle$$

  that we extend to nets by setting $(c, \overrightarrow{c}; \overrightarrow{t}) \to_m (\overrightarrow{c}', \overrightarrow{c}; \overrightarrow{t})$ whenever $c \to_m \overrightarrow{c}'$;
- the elimination of an axiom cut generates a substitution: we write $(\langle x|s\rangle, \overrightarrow{c}; \overrightarrow{t}) \to_a (\overrightarrow{c}; \overrightarrow{t})[s/\overline{x}]$ whenever $\overline{x} \notin \mathbf{V}(s)$;
- the elimination of an evanescent cut just deletes that cut: we write $(\langle\lambda|\mu\rangle, \overrightarrow{c}; \overrightarrow{t}) \to_e (\overrightarrow{c}; \overrightarrow{t})$ whenever $\jmath_p(\mu) \notin \{\mu, \lambda\}$.[4]

Then we write $p \to p'$ if $p \to_m p'$ or $p \to_a p'$ or $p \to_e p'$. Observe that if $p \to p'$ then $\mathbf{A}(p') \subseteq \mathbf{A}(p)$.

In order to define cut elimination between proof nets (and not bare proof nets only), we need to modify the jump function. Indeed, assume $p = (\langle t|s\rangle, \overrightarrow{c}; \overrightarrow{t})$ and p' is obtained from p by reducing the cut $\langle t|s\rangle$. Then $\mathbf{U}_\perp(p') \subseteq \mathbf{U}_\perp(p)$, but if $\mu \in \mathbf{U}_\perp(p')$ and $\jmath_p(\mu) = t$, we need to redefine $\jmath_{p'}(\mu)$, as in general $t \notin \mathbf{T}(p')$. This is done as follows:

- if $\langle t|s\rangle = \langle\otimes(t_1,\ldots,t_n)|\parr(s_1,\ldots,s_n)\rangle$ then for all $\mu \in \mathbf{U}_\perp(p) = \mathbf{U}_\perp(p')$ such that $\jmath_p(\mu) = \otimes(t_1,\ldots,t_n)$ (resp. $\parr(s_1,\ldots,s_n)$), we set $\jmath_{p'}(\mu) = t_1$ (resp. s_1);[5]
- if $\langle t|s\rangle = \langle t|x\rangle$ and p' is obtained from p by substituting t for $\overline{x}$, then for all $\mu \in \mathbf{U}_\perp(p) = \mathbf{U}_\perp(p')$ such that $\jmath_p \in \{x, \overline{x}\}$, we set $\jmath_{p'}(\mu) = t$;
- if $\langle t|s\rangle = \langle\mu|\lambda\rangle$, then for all $\mu' \in \mathbf{U}_\perp(p') = \mathbf{U}_\perp(p) \setminus \{\mu\}$ such that $\jmath_p(\mu') \in \{\mu, \lambda\}$, we set $\jmath_{p'}(\mu') = \jmath_p(\mu)$.

The result of eliminating the multiplicative cut (resp. axiom cut; evanescent cut) of the net p_0 of Figure 3.5 is depicted in Figure 3.6 (resp. Figure 3.7; Figure 3.8).

We are in fact interested in the simultaneous elimination of any number of reducible cuts, that we describe as follows. We write $p \rightrightarrows p'$ if

$$p = (c_1,\ldots,c_k, \langle x_1|t_1\rangle,\ldots,\langle x_n|t_n\rangle, \langle\mu_1|\lambda_1\rangle,\ldots,\langle\mu_l|\lambda_l\rangle, \overrightarrow{c}; \overrightarrow{t})$$

and

$$p' = (\overrightarrow{c}'_1,\ldots,\overrightarrow{c}'_k, \overrightarrow{c}; \overrightarrow{t})[t_1/\overline{x}_1]\cdots[t_n/\overline{x}_n],$$

assuming that:

- $c_i \to_m \overrightarrow{c}'_i$ for $1 \leq i \leq k$,

4. Since the cuts of a net are given as a family rather than a sequence, the order in which we write cuts in this definition is not relevant: despite our abusive notation, the reduced cut need not be the first in the enumeration, because this enumeration is not fixed.

5. We arbitrarily redirect the jumps to the first subtree to simplify the presentation, but we could equivalently have set $\jmath_{p'}(\mu)$ to be any of the immediate subtrees of $\jmath_p(\mu)$, non deterministically: in fact, this slight generalization is necessary to deal with cut elimination in resource nets.

Other strategies for choosing the destination of a jump exist in the literature: for instance, one may be tempted to systematically redirect jumps to atoms, as it is done by Tortora de Falco [Tor00, Definition 1.3.3]. But this kind of transformation is not local and it would certainly complicate our arguments.

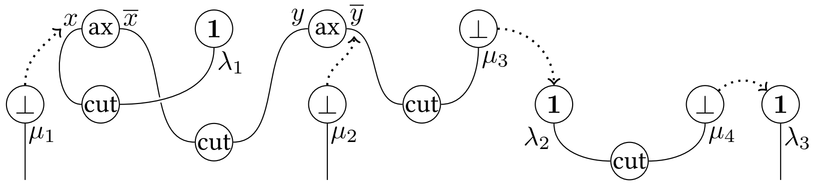

Figure 3.6 – The net $p'_m = (\langle x|\mathbf{1}_{\lambda_1}\rangle, \langle \overline{x}|y\rangle, \langle \overline{y}|\perp_{\mu_3}\rangle, \langle \mathbf{1}_{\lambda_2}|\perp_{\mu_4}\rangle; \perp_{\mu_1}, \perp_{\mu_2}, \mathbf{1}_{\lambda_3})$ with $\jmath_{p'_m} : \mu_1 \mapsto x, \mu_2 \mapsto \overline{y}, \mu_3 \mapsto \lambda_2, \mu_4 \mapsto \lambda_3$, so that $p_0 \to_m p'_m$.

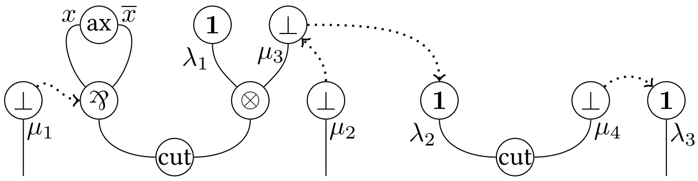

Figure 3.7 – The net $p'_a = (\langle ⅋(x,\overline{x})|\otimes(\mathbf{1}_{\lambda_1}, \perp_{\mu_3}\rangle, \langle \mathbf{1}_{\lambda_2}|\perp_{\mu_4}\rangle; \perp_{\mu_1}, \perp_{\mu_2}, \mathbf{1}_{\lambda_3})$ with $\jmath_{p'_a} : \mu_1 \mapsto ⅋(x,\overline{x}), \mu_2 \mapsto \mu_3, \mu_3 \mapsto \lambda_2, \mu_4 \mapsto \lambda_3$, so that $p_0 \to_a p'_a$.

– $\overline{x}_i \notin \{x_1, \ldots, x_n\}$ and $\overline{x}_i \notin \mathbf{V}(t_j)$ for $1 \leq i \leq j \leq n$, and
– $\jmath_p(\mu_i) \notin \{\mu_j, \lambda_j\}$ for $1 \leq i \leq j \leq l$.

It should be clear that p' is then obtained from p by successively eliminating the particular cuts we have selected, thus performing k steps of $\to_m$, n steps of $\to_a$, l steps of $\to_e$, *in no particular order*: indeed, one can check that any two elimination steps of distinct cuts commute on the nose. The resulting jump function $\jmath_{p'}$ can be described directly, by inspecting the possible cases for $\jmath_p(\mu')$ with $\mu' \in \mathbf{U}_\perp(p')$:

– if $c_i = \langle \otimes(u_1, \ldots, u_r)|⅋(v_1, \ldots, v_r)\rangle$ and, e.g., $\jmath_p(\mu') = \otimes(u_1, \ldots, u_r)$ then $\jmath_{p'}(\mu') = u_1[t_1/\overline{x}_1] \cdots [t_n/\overline{x}_n]$;
– if $\jmath_p(\mu') \in \{x_i, \overline{x}_i\}$ then $\jmath_{p'}(\mu') = t_i[t_{i+1}/\overline{x}_{i+1}] \cdots [t_n/\overline{x}_n]$;
– if $\jmath_p(\mu') \in \{\mu_i, \lambda_i\}$ then $\jmath_{p'}(\mu') = \rho(i)[t_1/\overline{x}_1] \cdots [t_n/\overline{x}_n]$, where $\rho : \{1, \ldots, l\} \to \mathbf{T}(p)$ is the redirection function inductively defined by $\rho(j) = \rho(i)$ if $\jmath_p(\mu_j) \in \{\mu_i, \lambda_i\}$ (in which case $i < j$) and $\rho(j) = \jmath_p(\mu_j)$ otherwise;
– otherwise $\jmath_{p'}(\mu') = \jmath_p(\mu')[t_1/\overline{x}_1] \cdots [t_n/\overline{x}_n]$.

The result of simultaneously eliminating all the cuts of the net p_0 of Figure 3.5 is depicted in Figure 3.9.

This general description of parallel cut elimination is obviously not very handy. In order not to get lost in notation, we will restrict our attention to the particular case in which only cuts of the same nature are simultaneously eliminated: we write $p \rightrightarrows_m p'$ if $n = l = 0$ (multiplicative cuts only), $p \rightrightarrows_a p$ if $k = l = 0$ (axiom cuts only), and $p \rightrightarrows_e p'$ if $n = k = 0$ (evanescent cuts only). Then we can decompose any parallel reduction $p \rightrightarrows p'$ into three separate steps: e.g., $p \rightrightarrows_m \cdot \rightrightarrows_a \cdot \rightrightarrows_e p'$.[6]

6. Of course, the converse does not hold: for instance the reductions $(\langle ⅋(x,\overline{x})|\otimes(y,z)\rangle; \overline{y}, \overline{z}) \rightrightarrows_m$

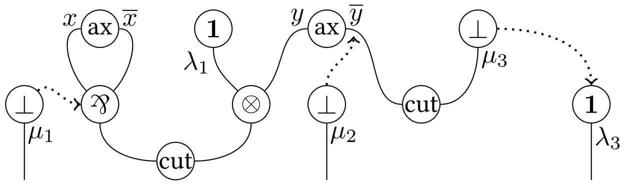

Figure 3.8 – The net $p'_e = (\langle ⅋(x,\overline{x})|\otimes(\mathbf{1}_{\lambda_1},y)\rangle, \langle\overline{y}|\bot_{\mu_3}\rangle; \bot_{\mu_1}, \bot_{\mu_2}, \mathbf{1}_{\lambda_3})$ with $\jmath_{p'_e} : \mu_1 \mapsto ⅋(x,\overline{x}), \mu_2 \mapsto \overline{y}, \mu_3 \mapsto \lambda_3$, so that $p_0 \to_e p'_e$.

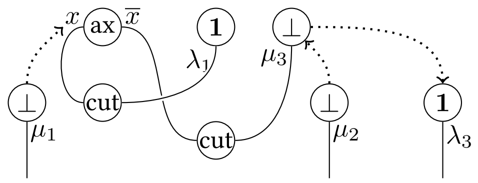

Figure 3.9 – The net $p' = (\langle x|\mathbf{1}_{\lambda_1}\rangle, \langle\overline{x}|\bot_{\mu_3}\rangle; \bot_{\mu_1}, \bot_{\mu_2}, \mathbf{1}_{\lambda_3})$ with $\jmath_{p'} : \mu_1 \mapsto x, \mu_2 \mapsto \mu_3, \mu_3 \mapsto \lambda_3$, so that $p_0 \rightrightarrows p'$.

3.2.3 Paths

In order to control the effect of parallel reduction on the size of proof nets, we rely on a side condition involving the length of switching paths, *i.e.* paths in the sense of Danos–Regnier's correctness criterion [DR89].

Let us write $\mathbf{T}_{⅋}(p)$ (resp. $\mathbf{T}_{\otimes}(p)$) for the set of the subtrees of p of the form $⅋(t_1,\dots,t_n)$ (resp. $\otimes(t_1,\dots,t_n)$). In our setting, a *switching* of a net p is a map $I : \mathbf{T}_{⅋}(p) \to \mathbf{T}(p)$ such that, for each $t = ⅋(t_1,\dots t_n) \in \mathbf{T}_{⅋}(p)$, $I(t) \in \{t_1,\dots,t_n\}$. Given a net p and a switching I of p, the associated *switching graph* is the unoriented graph with vertices in $\mathbf{T}(p)$ and edges given as follows:

- one *axiom edge* $\sim_{\{x,\overline{x}\}}$ for each axiom $\{x,\overline{x}\} \subseteq \mathbf{V}(p)$, connecting x and $\overline{x}$;
- one $\otimes$-*edge* $\sim_{t,t_i}$ for each pair (t,t_i) with $t = \otimes(t_1,\dots,t_n) \in \mathbf{T}_{\otimes}(p)$, connecting t and t_i;
- one $⅋$-*edge* $\sim_t$ for each $t \in \mathbf{T}_{⅋}(p)$, connecting t and $I(t)$;
- one *jump edge* $\sim_\mu$ for each $\mu \in \mathbf{U}_\bot(p)$, connecting μ and $\jmath_p(\mu)$;
- one *cut edge* $\sim_c$ for each cut $c = \langle t|s\rangle \in \mathbf{C}(p)$, connecting t and s.

Whenever necessary, we may write, e.g., $\sim^p_e$ or $\sim^{p,I}_e$ for the edge $\sim_e$ to make the underlying net and switching explicit. On the other hand, we will often simply write e instead of $\sim_e$ for denoting an edge. Each edge e induces a symmetric relation, involving at most two subtrees of p: we write $t \sim_e u$, and say t and u are *adjacent* whenever t and u are connected by e. *A priori*, it might be the case that distinct edges induce the same relation: for instance, if $c = \langle x|\overline{x}\rangle \in \mathbf{C}(p)$,

$(\langle x|y\rangle, \langle\overline{x}|z\rangle; \overline{y}, \overline{z}) \rightrightarrows_a (\langle y|z\rangle; \overline{y}, \overline{z})$ cannot be performed in a single step, as the cut $\langle x|y\rangle$ was newly created.

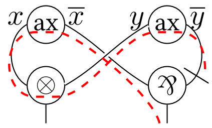

Figure 3.10 – A path in $(;\otimes(x,y),\parr(\overline{y},\overline{x}))$ for switching $I:\parr(\overline{y},\overline{x})\mapsto\overline{x}$ (we strike out the other premise).

we have $x\sim_c\overline{x}$ as well as $x\sim_{\{x,\overline{x}\}}\overline{x}$; and if $\jmath_p(\mu)=\mu'$ and $\jmath_p(\mu')=\mu$, we have $\mu\sim_\mu\mu'$ as well as $\mu\sim_{\mu'}\mu'$. Avoiding such cycles is precisely the purpose of the correctness criterion.

Given a switching I in p, an *I-path* is the data of a tree $t_0\in\mathbf{T}(p)$ and of a sequence $(e_1,\ldots,e_n)$ of pairwise distinct consecutive edges starting from t_0: in other words, we require that there exist $t_1,\ldots,t_n\in\mathbf{T}(p)$ such that, for each $i\in\{1,\ldots,n\}$, $t_{i-1}\sim_{e_i}t_i$.[7] For instance, if $p=(;\otimes(x,y),\parr(\overline{y},\overline{x}))$ and $I(\parr(\overline{y},\overline{x}))=\overline{x}$, then the chain of adjacencies $\parr(\overline{x},\overline{y})\sim_{\parr(\overline{x},\overline{y})}\overline{x}\sim_{\{x,\overline{x}\}}x\sim_{\otimes(x,y),x}\otimes(x,y)\sim_{\otimes(x,y),y}y\sim_{\{y,\overline{y}\}}\overline{y}$ defines a maximal I-path in p (see Figure 3.10).

We write $\mathbf{P}(p,I)$ for the set of all I-paths in p. We write $\chi:t_0\rightsquigarrow_{p,I}t_n$ whenever $\chi=t_0\sim_{e_1}\cdots\sim_{e_n}t_n$ is an I-path from t_0 to t_n in p: with these notations, we say χ *visits* the trees $t_0,\ldots,t_n$, and χ *crosses* the edges $e_1,\ldots,e_n$; moreover we write $\mathbf{ln}(\chi)=n$ for the length of χ. A *subpath* of χ is any I-path of the form $t_i\sim_{e_{i+1}}\cdots\sim_{e_{i+k}}t_{i+k}$. We write $\chi^\dagger$ for the reverse I-path: $\chi^\dagger=t_n\sim_{e_n}\cdots\sim_{e_0}t_0$. The *empty path* from $t\in\mathbf{T}(p)$ is the only $\epsilon_t:t\rightsquigarrow_{p,I}t$ of length 0. We say I-paths χ and χ' are *disjoint* if no edge is crossed by both χ and χ'. Observe that disjoint paths can visit common trees: in particular, if $\chi:t\rightsquigarrow_{p,I}s$ and $\chi':s\rightsquigarrow_{p,I}u$ are disjoint, we write $\chi\chi':t\rightsquigarrow_{p,I}u$ for the concatenation of χ and χ'.

We call *path* in p any I-path for I a switching of p, we write $\mathbf{P}(p)$ for the set of all paths in p, and we denote by $\mathbf{ln}(p)=\max\{\mathbf{ln}(\chi)\mid\chi\in\mathbf{P}(p)\}$ the maximal length of a path in p. We then write $\chi:t\rightsquigarrow_p s$ if $\chi\in\mathbf{P}(p)$ is a path from t to s in p, and we write $t\rightsquigarrow_p s$, or simply $t\rightsquigarrow s$, whenever such a path exists. This relation on $\mathbf{T}(p)$ is reflexive (*via* the empty path $\epsilon_t:t\rightsquigarrow t$) and symmetric (*via* reversing paths). Observe that if $\chi_1,\cdots,\chi_n\in\mathbf{P}(p)$ are pairwise disjoint, then there exists I such that $\chi_i\in\mathbf{P}(p,I)$ for $1\le i\le n$. This does not make the relation $\rightsquigarrow_p$ transitive: for instance, if $p=(;x,\parr(\overline{x},\overline{y}),y)$, we have paths $x\rightsquigarrow\parr(\overline{x},\overline{y})$ and $\parr(\overline{x},\overline{y})\rightsquigarrow y$, but both cross $\parr(\overline{x},\overline{y})$ for different switchings, and indeed, there is no path $x\rightsquigarrow y$.

We say a net p is *acyclic* if, for all $\chi\in\mathbf{P}(p)$ and $t\in\mathbf{T}(p)$, χ visits t at most once: in other words, there is no (non-empty) *cycle* $\chi:t\rightsquigarrow t$. Notice that, given any family of pairwise distinct cuts in an acyclic net p, it is always possible to satisfy the side conditions on free variables and on jumps necessary to reduce these cuts in parallel (provided each cut in the family has the shape of a multiplicative, axiom or evanescent cut). Moreover, it is a very standard result that acyclicity is preserved by cut elimination:

Lemma 3.2.3. *If p' is obtained from p by cut elimination and p is acyclic then so is p'.*

Proof. It suffices to check that if $p\to p'$ then any cycle in p' induces a cycle in p.[8] □

7. In standard terminology of graph theory, an I-path in p is a trail in the switching graph induced by p and I.
8. We do not detail the proof as it is quite standard. We will moreover generalize this technique to all paths (and

From now on, we consider acyclic nets only.

3.3 Bounding the size of antireducts: three kinds of cuts

In this section, we show that the loss of size during a parallel reduction $p \rightrightarrows_m q$, $p \rightrightarrows_a q$ or $p \rightrightarrows_e q$ is directly controlled by $\mathbf{ln}(p)$, $\mathbf{jd}(p)$ and $\mathbf{size}(q)$: more precisely, we show that the ratio $\frac{\mathbf{size}(p)}{\mathbf{size}(q)}$ is bounded by a function of $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$ in each case.

3.3.1 Elimination of multiplicative cuts

The elimination of multiplicative cuts cannot decrease the size by more than a half:

Lemma 3.3.1. *If* $p \rightrightarrows_m q$ *then* $\mathbf{size}(p) \leq 2\mathbf{size}(q)$.

Proof. Since the elimination of a multiplicative cut does not affect the rest of the (bare) net, it is sufficient to observe that if $c \rightarrow_m \overrightarrow{c}$ then $\mathbf{size}(c) = 2 + \mathbf{size}(\overrightarrow{c}) \leq 2\mathbf{size}(\overrightarrow{c})$.[9] □

So in this case, $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$ actually play no rôle.

3.3.2 Elimination of axiom cuts

Observe that:

- if $x \in \mathbf{V}(\gamma)$ then $\mathbf{size}(\gamma[t/x]) = \mathbf{size}(\gamma) + \mathbf{size}(t) - 1$;
- if $x \notin \mathbf{V}(\gamma)$ then $\mathbf{size}(\gamma[t/x]) = \mathbf{size}(\gamma)$.

It follows that, in the elimination of a single axiom cut $p \rightarrow_a q$, we have $\mathbf{size}(p) = \mathbf{size}(q) + 2$. But we cannot reproduce the proof of Lemma 3.3.1 for $\rightrightarrows_a$: as depicted in Figure 3.2, a chain of arbitrarily many axiom cuts may reduce into a single wire. We can bound the length of those chains by $\mathbf{ln}(p)$, however, and this allows us to bound the loss of size during reduction.

Lemma 3.3.2. *If* $p \rightrightarrows_a q$ *then* $\mathbf{size}(p) \leq (\mathbf{ln}(p) + 1)\mathbf{size}(q)$.

Proof. Assume $p = (\langle x_1|t_1\rangle, \dots, \langle x_n|t_n\rangle, \overrightarrow{c}; \overrightarrow{s})$ and $q = (\overrightarrow{c}; \overrightarrow{s})[t_1/\overline{x}_1] \cdots [t_n/\overline{x}_n]$ with $\overline{x}_i \notin \{x_1, \dots, x_n\}$ and $\overline{x}_i \notin \mathbf{V}(t_j)$ for $1 \leq i \leq j \leq n$. To establish the result in this case, we make the chains of eliminated axiom cuts explicit.

Due to the condition on free variables, we can partition $\langle x_1|t_1\rangle, \dots, \langle x_n|t_n\rangle$ into tuples $\overrightarrow{c}_1, \dots, \overrightarrow{c}_k$ of the shape $\overrightarrow{c}_i = (\langle x_0^i|\overline{x}_1^i\rangle, \dots, \langle x_{n_i-1}^i|\overline{x}_{n_i}^i\rangle, \langle x_{n_i}^i|t^i\rangle)$ so that:

- $x_j^i \in \{x_1, \dots, x_n\}$ for $1 \leq i \leq k$ and $0 \leq j \leq n_i$;
- $\overline{x}_j^i \in \{t_1, \dots, t_n\}$ for $1 \leq i \leq k$ and $1 \leq j \leq n_i$;
- $t^i \in \{t_1, \dots, t_n\}$ for $1 \leq i \leq k$;
- each $\overrightarrow{c}_i$ is maximal with this shape, *i.e.* $\overline{x}_0^i \notin \{t_1, \dots, t_n\}$ and $t^i \notin \{\overline{x}_1, \dots, \overline{x}_n\}$.

not only cycles) in the next section.

9. This is due to the fact that we distinguish between strict connectives and their nullary versions, that are subject to evanescent reductions.

Without loss of generality, we can moreover require that, if $i < i'$, then $\langle x^i_{n_i}|t^i\rangle$ occurs before $\langle x^{i'}_{n_{i'}}|t^{i'}\rangle$ in the tuple $(\langle x_1|t_1\rangle, \dots, \langle x_n|t_n\rangle)$. Moreover observe that, by the condition on free variables, the order of the cuts in each $\overrightarrow{c}_i$ is necessarily the same as in $(\langle x_1|t_1\rangle, \dots, \langle x_n|t_n\rangle)$.

By a standard result on substitutions, if $x \neq y$, $x \notin \mathbf{V}(v)$ and $y \notin \mathbf{V}(u)$ then $\gamma[u/x][v/y] = \gamma[v/y][u/x]$:

- if $i \neq i'$, we have $\overline{x}^i_j \neq \overline{x}^{i'}_{j'}$ for $0 \leq j \leq n_i$ and $0 \leq j' \leq n_{i'}$, so the substitutions $[\overline{x}^i_{j+1}/\overline{x}^i_j]$ and $[\overline{x}^{i'}_{j'+1}/\overline{x}^{i'}_{j'}]$ always commute for $1 \leq i < i' \leq k$, $0 \leq j < n_i$ and $0 \leq j' < n_{j'}$;
- if $i < i'$ and $\langle x^{i'}_{j'}|\overline{x}^{i'}_{j'+1}\rangle$ occurs before $\langle x^i_{n_i}|t^i\rangle$ in $(\langle x_1|t_1\rangle, \dots, \langle x_n|t_n\rangle)$, the condition on free variables imposes that $\overline{x}^{i'}_{j'} \notin \mathbf{V}(t^i)$ so the substitutions $[t^i/\overline{x}^i_{n_i}]$ and $[\overline{x}^{i'}_{j'+1}/\overline{x}^{i'}_{j'}]$ commute in this case.

By iterating those two observations, we can reorder the substitutions in q and obtain:

$$\begin{aligned}
q &= (\overrightarrow{c}; \overrightarrow{s})[t_1/\overline{x}_1] \cdots [t_n/\overline{x}_n] \\
&= (\overrightarrow{c}; \overrightarrow{s})[\overline{x}^1_1/\overline{x}^1_0] \cdots [\overline{x}^1_{n_1}/\overline{x}^1_{n_1-1}][t^1/\overline{x}^1_{n_1}] \cdots [\overline{x}^k_1/\overline{x}^k_0] \cdots [\overline{x}^k_{n_k}/\overline{x}^k_{n_k-1}][t^k/\overline{x}^k_{n_k}] \\
&= (\overrightarrow{c}; \overrightarrow{s})[t^1/\overline{x}^1_0] \cdots [t^k/\overline{x}^k_0].
\end{aligned}$$

It follows that $\mathbf{size}(q) = \mathbf{size}(\overrightarrow{c}) + \mathbf{size}(\overrightarrow{s}) + \sum_{i=1}^k \mathbf{size}(t^i) - k$. For $1 \leq i \leq k$, $\overrightarrow{c}_i$ induces a path $\overline{x}^0_i \rightsquigarrow t^i$ of length $2n_i + 2$ ($n_i + 1$ cuts and $n_i + 1$ axioms). Hence $2n_i \leq \mathbf{ln}(p) - 2$ and:

$$\begin{aligned}
\mathbf{size}(p) &= \mathbf{size}(\overrightarrow{c}) + \mathbf{size}(\overrightarrow{s}) + \sum_{i=1}^k (\mathbf{size}(t^i) + 2n_i + 1) \\
&\leq \mathbf{size}(\overrightarrow{c}) + \mathbf{size}(\overrightarrow{s}) + \sum_{i=1}^k \mathbf{size}(t^i) + k(\mathbf{ln}(p) - 1) \\
&\leq \mathbf{size}(q) + k\mathbf{ln}(p).
\end{aligned}$$

To conclude, it will be sufficient to prove that $\mathbf{size}(q) \geq k$. For $1 \leq i \leq k$, let $A_i = \{j > i \mid \overline{x}^j_0 \in \mathbf{V}(t^i)\}$, and then let $A_0 = \{i \mid \overline{x}^i_0 \in \mathbf{V}(\overrightarrow{c}, \overrightarrow{s})\}$. It follows from the construction that $\{A_0, \dots, A_{k-1}\}$ is a partition (possibly including empty sets) of $\{1, \dots, k\}$. By construction, for each $j \in A_i$, $\overline{x}^j_0$ is a strict subtree of t^i: it follows that $\mathbf{size}(t^i) > \#A_i$. Now consider $q_i = (\overrightarrow{c}; \overrightarrow{s})[t^1/\overline{x}^1_0] \cdots [t^i/\overline{x}^i_0]$ for $0 \leq i \leq k$ so that $q = q_k$. For $1 \leq i \leq k$, we obtain $\mathbf{size}(q_i) = \mathbf{size}(q_{i-1}) + \mathbf{size}(t^i) - 1 \geq \mathbf{size}(q_{i-1}) + \#A_i$. Also observe that $\mathbf{size}(q_0) = \mathbf{size}(\overrightarrow{c}; \overrightarrow{s}) \geq \#A_0$. Then we can conclude: $\mathbf{size}(q) = \mathbf{size}(q_k) \geq \sum_{i=0}^k \#A_i = k$. □

3.3.3 Elimination of evanescent cuts

We now consider the case of a reduction $p \rightrightarrows_e q$: we bound the maximal number of evanescent cuts appearing in p by a function of $\mathbf{ln}(p)$, $\mathbf{jd}(p)$ and $\mathbf{size}(q)$.

We rely on the basic fact that if $t \in \mathbf{T}(q) \subseteq \mathbf{T}(p)$, then there are at most $\mathbf{jd}(p)$ evanescent cuts of p that jump to t. The main difficulty is that an evanescent cut of p can jump to another evanescent cut of p, that is also eliminated in the step $p \rightrightarrows_e q$. See Figure 3.11 for a graphical representation of the critical case. To deal with this phenomenon, we observe that a sequence

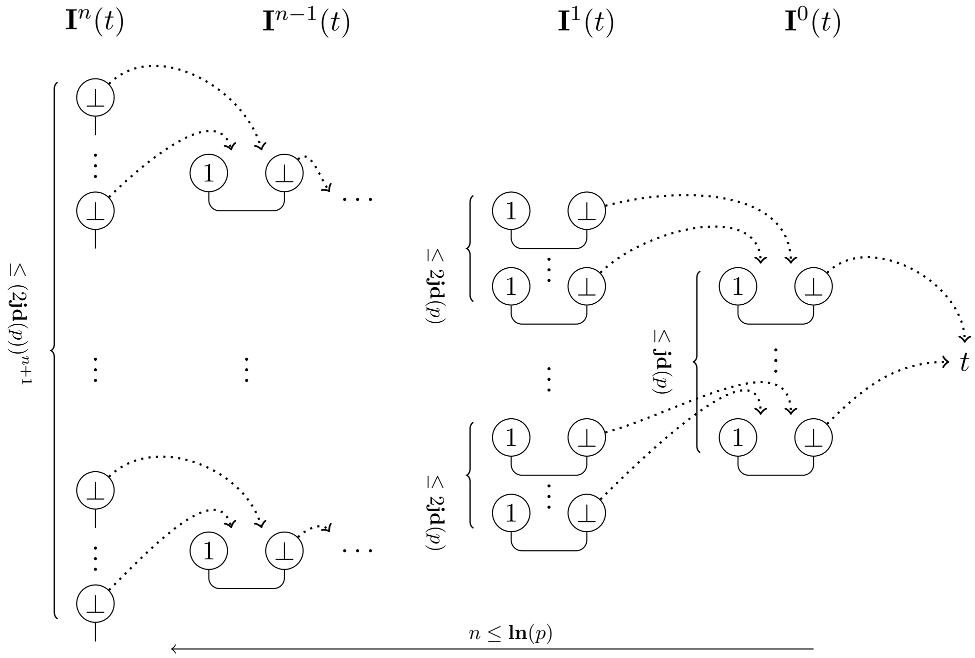

Figure 3.11 – Evanescent reductions : critical case

of cuts $\langle \mu_1 | \lambda_1 \rangle, \ldots, \langle \mu_n | \lambda_n \rangle$ with $\jmath_p(\mu_i) \in \{\lambda_{i+1}, \mu_{i+1}\}$ for all $i \in \{1, \ldots, n-1\}$, induces a path of length at least n: hence $n \leq \mathbf{ln}(p)$.

Definition 3.3.3. We define for all $n \in \mathbf{N}$ and all $t \in \mathbf{T}(p)$, $\mathbf{I}^n(t)$ as follows : $\mathbf{I}^0(t) = \jmath_p^{-1}(t) = \{\mu \in \mathbf{U}_\perp(p) \mid \jmath_p(\mu) = t\}$, and $\mathbf{I}^{m+1}(t) = \{\mu' \in \mathbf{U}_\perp(p) \mid \jmath_p(\mu') \in \{\mu, \lambda\}, \langle \mu | \lambda \rangle \in \mathbf{C}(p), \mu \in \mathbf{I}^m(t)\}$.

We can already observe that $\#\mathbf{I}^0(t) = \mathbf{jd}(t)$. This definition is parametrized by $\jmath_p$, and may we write $\mathbf{I}_p^n(t)$ to make the underlying net explicit.

Lemma 3.3.4. *Let p, q be two nets such that $p \rightrightarrows_e q$. Then:*

1. *for all $t \in \mathbf{T}(p)$, $\#\left(\bigcup_{i \in \mathbf{N}} \mathbf{I}_p^i(t)\right) \leq (2\mathbf{jd}(p))^{\mathbf{ln}(p)+1}$;*
2. *there are at most $\mathbf{size}(q) \times (2\mathbf{jd}(p))^{\mathbf{ln}(p)+1}$ evanescent cuts in $\mathbf{C}(p)$.*

Proof. We first establish that the set $\{n \in \mathbf{N} \mid \mathbf{I}^n(t) \neq \emptyset\}$ is finite for all t. Indeed, for each $\mu_n \in \mathbf{I}^n(t)$, there is a sequence of cuts $c_0, \ldots, c_{n-1}$ such that, writing $c_i = \langle \lambda_i | \mu_i \rangle$, the unique path from μ_n to t is $\chi = \chi_n \cdots \chi_1(\mu_0 \sim_{\mu_0} t)$ where, for $1 \leq i \leq n$:

– either $\jmath_p(\mu_i) = \lambda_{i-1}$ and $\chi_i = \mu_i \sim_{\mu_i} \lambda_{i-1} \sim_{c_{i-1}} \mu_{i-1}$;

– or $\jmath_p(\mu_i) = \mu_{i-1}$ and $\chi_i = \mu_i \sim_{\mu_i} \mu_{i-1}$.

We observe that $\mathbf{ln}(\chi) \geq n+1$, and then we deduce $n < \mathbf{ln}(p)$ as soon as $\mathbf{I}^n(t) \neq \emptyset$.

Now we bound the size of each $\mathbf{I}_p^n(t)$: we show that $\#\mathbf{I}_p^n(t) \leq (2\mathbf{jd}(p))^{n+1}$, by induction on n. We already have $\#\mathbf{I}_p^0(t) = \mathbf{jd}(t) \leq \mathbf{jd}(p)$. Now assume the result holds for $n \geq 0$. Then, for each $c = \langle\mu|\lambda\rangle \in \mathbf{C}(p)$ such that $\mu \in \mathbf{I}_p^n(t)$, the number of μ' such that $\jmath_p(\mu') \in c$ is at most $\mathbf{jd}(\mu) + \mathbf{jd}(\lambda) \leq 2\mathbf{jd}(p)$. We obtain: $\#\mathbf{I}_p^{n+1}(t) \leq 2\mathbf{jd}(p)\#\mathbf{I}_p^n$, which enables the induction.

We thus get $\#\left(\bigcup_{i\in\mathbf{N}} \mathbf{I}_p^i(t)\right) \leq \sum_{i=0}^{\mathbf{ln}(p)-1}(2\mathbf{jd}(p))^{i+1} \leq (2\mathbf{jd}(p))^{\mathbf{ln}(p)+1}$ which entails (1). To deduce (2) from (1), it will be sufficient to show that, for each $\mu \in \mathbf{U}_\perp(p)$, there exists $t \in \mathbf{T}(q)$ such that $\mu \in \mathbf{I}_p^k(t)$ for some $k \in \mathbf{N}$: indeed the number of evanescent cuts in p is obviously bounded by $\#\mathbf{U}_\perp(p)$.

For that purpose, write $\mu_0 = \mu$ and let $(c_i)_{i\in\{1,\ldots,k\}}$ be the longest sequence of cuts $c_i = \langle\lambda_i|\mu_i\rangle \in \mathbf{C}(p)$ such that, for all $i \in \{0,\ldots,k-1\}$, $\jmath_p(\mu_i) \in \{\mu_{i+1}, \lambda_{i+1}\}$: such a maximal sequence exists by acyclicity. Necessarily, $\jmath_p(\mu_k)$ is not part of an evanescent cut in $\mathbf{C}(p)$, so $\jmath_p(\mu_k) \in \mathbf{T}(q)$: we conclude since $\mu_0 \in \mathbf{I}_p^k(\jmath_p(\mu_k))$. □

Writing $\psi(i,j,k) = i(1+2(2j)^{k+1})$, we obtain:

Lemma 3.3.5. *If $p \rightrightarrows_e q$, then $\mathbf{size}(p) \leq \psi(\mathbf{size}(q), \mathbf{jd}(p), \mathbf{ln}(p))$.*

Proof. Writing $p = (\langle\mu_1|\lambda_1\rangle, \ldots, \langle\mu_n|\lambda_n\rangle, \overrightarrow{c}; \overrightarrow{t})$ and $q = (\overrightarrow{c}; \overrightarrow{t})$, we obtain

$$\mathbf{size}(p) = \mathbf{size}(q) + 2n \leq \mathbf{size}(q)(1 + 2(2\mathbf{jd}(p))^{\mathbf{ln}(p)+1})$$

by Lemma 3.3.4. □

3.3.4 Towards the general case

Recall that any parallel cut elimination step $p \rightrightarrows q$ can be decomposed into, e.g.: $p \rightrightarrows_e p' \rightrightarrows_m p'' \rightrightarrows_a q$. We would like to apply the previous results to this sequence of reductions, in order to bound the size of p by a function of $\mathbf{size}(q)$, $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$. Observe however that this would require us to infer a bound on $\mathbf{ln}(p'')$ from the bounds on p, in order to apply Lemma 3.3.2.

More generally, to be able to apply our results to a sequence of reductions $p \rightrightarrows \cdots \rightrightarrows q$, we need to ensure that for any reduction $p \rightrightarrows p'$, we can bound $\mathbf{ln}(p')$ and $\mathbf{jd}(p')$ by functions of $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$. This is the subject of the following two sections.

3.4 Variations of $\ln(p)$ under reduction

Here we establish that the possible increase of $\mathbf{ln}(p)$ under reduction is bounded. It should be clear that:

Lemma 3.4.1. *If $p \rightrightarrows_a q$ or $p \rightrightarrows_e q$, then $\mathbf{ln}(q) \leq \mathbf{ln}(p)$.*

Indeed axiom and evanescent reductions only shorten paths, without really changing the topology of the net.

In the case of multiplicative cuts however, cuts are duplicated and new paths are created. Consider for instance a net r, as in Figure 3.12, obtained from three nets p_1, p_2 and q, by forming the cut $\langle\otimes(t_1,t_2)|\parr(s_1,s_2)\rangle$ where $t_1 \in \mathbf{T}(p_1)$, $t_2 \in \mathbf{T}(p_2)$ and $s_1, s_2 \in \mathbf{T}(q)$. Observe that, in the reduct r' obtained by forming two cuts $\langle t_1|s_1\rangle$ and $\langle t_2|s_2\rangle$, we may very well form a path

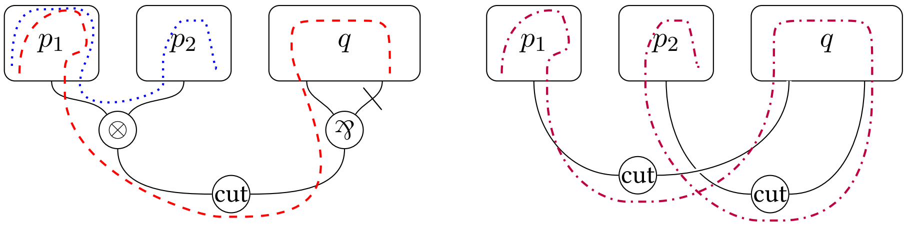

Figure 3.12 – A cut, the resulting slipknot, and examples of paths before and after reduction

that travels from p_1 to q then p_2; while in p, this is forbidden by any switching of $⅋(s_1, s_2)$. For instance, if we consider $I(⅋(s_1, s_2)) = s_1$, we may only form a path between p_1 and p_2 through $\otimes(t_1, t_2)$, or a path between q and one of the p_i's, through s_1 and the cut.

In the remainder of this section, we fix a reduction step $p \rightrightarrows_m q$, and we show that the previous example describes a general mechanism: a path χ in q that is not already in p must involve a subpath χ' between two residuals of a cut of p that was eliminated in $p \rightrightarrows_m q$. We refer to this situation as a *slipknot* in χ.

More formally, consider $c = \langle t_0 | s_0 \rangle \in \mathbf{C}(p)$ with $t_0 = \otimes(t_1, \ldots, t_n)$ and $s_0 = ⅋(s_1, \ldots, s_n)$ and assume c is eliminated in the reduction $p \rightrightarrows_m q$: then the *residuals* of c in q are the cuts $\langle t_i | s_i \rangle \in \mathbf{C}(q)$ for $1 \leq i \leq n$. For any edge e, we write (e) for any length 1 path $t \sim_e s$. If $\chi \in \mathbf{P}(q)$, a *slipknot* of χ is any subpath $(d)\xi(d')$ where d and d' are (necessarily distinct) residuals of the same cut in p. In the remaining of this section, we show that a path in q is necessarily obtained by alternating paths (essentially) in p and slipknots in q, that recursively consist of such alternations. This will allow us to bound $\mathbf{ln}(q)$ depending on $\mathbf{ln}(p)$, by reasoning inductively on these paths.

3.4.1 Preserved paths

Notice that $\mathbf{T}(q) \subseteq \mathbf{T}(p)$ and, given a switching J of q, it is always possible to extend J into a switching I of p: to determine I uniquely amounts to select a premise for each $⅋$-tree in an eliminated cut.

Let J be a switching of q and I an extension of J on p. Observe that if $t \sim_e^{p,I} t'$ and neither t nor t' is an element of an eliminated cut, then e is also an edge of q that is not a residual cut; conversely, if $t \sim_e^{q,J} t'$ and e is not a residual cut, then e is also an edge of p. We then say the edge e is *preserved* by the reduction $p \rightrightarrows_m q$. If a preserved edge e is a cut, an axiom, a $\otimes$-edge or a $⅋$-edge, then e has the same endpoints in p and in q: $t \sim_e^{p,I} t'$ iff $t \sim_e^{q,J} t'$. If $e = \mu$ is a jump, one endpoint might be changed: indeed, $\mu \sim_e^p \jmath_p(\mu)$ and $\mu \sim_e^q \jmath_q(\mu)$, and we might have $\jmath_p(\mu) \neq \jmath_q(\mu)$ when $\jmath_p(\mu)$ is part of an eliminated cut. In this case, we say e is a *redirected jump*. We say $u \in \mathbf{T}(p)$ is an *anchor* of $v \in \mathbf{T}(q)$, if either $u = v$ or u is involved in an eliminated cut $c = \langle u | u' \rangle$, and either $u = \otimes(\overrightarrow{t})$ and $v \in \overrightarrow{t}$ or $u = ⅋(\overrightarrow{s})$ and $v \in \overrightarrow{s}$.

Lemma 3.4.2. *Assume $(d)\chi(d') \in \mathbf{P}(q)$ and d and d' are residuals. Then χ is non empty, and its first and last edges are preserved.*

Proof. This is a direct consequence of the fact that if $(c)(e)$ is a path and c is a cut then e is not a cut. $\square$

Observe that even if a path $\chi \in \mathbf{P}(q)$ crosses preserved edges only, it is not sufficient to have $\chi \in \mathbf{P}(p)$, because the endpoints of redirected jumps might change. We say χ is a *preserved path* if χ crosses preserved edges only, and we can write either $\chi = \chi'$ or $\chi = (t \sim_\mu \perp_\mu)\chi'$ or $\chi'(\perp_{\mu'} \sim_{\mu'} t')$ or $\chi = (t \sim_\mu \perp_\mu)\chi'(\perp_{\mu'} \sim_{\mu'} t')$ where χ' crosses no redirected jump.

Lemma 3.4.3. *Any non empty preserved path $\chi : t \rightsquigarrow_q t'$ induces a unique preserved path $\chi^- : s \rightsquigarrow_p s'$ with the same sequence of edges: in particular, $\chi^- \in \mathbf{P}(p, I)$ as soon as $\chi \in \mathbf{P}(q, J)$ and I is an extension of J; and s (resp. s') is an anchor of t (resp. t'). Moreover, if $\chi_1\chi_2$ is a preserved path then $(\chi_1\chi_2)^- = \chi_1^-\chi_2^- \in \mathbf{P}(p)$.*

Proof. The first part is a direct consequence of the definition. If moreover $\chi_1\chi_2$ is a preserved path, and neither χ_1 nor χ_2 is an empty path, then neither the last edge of χ_1 nor the first edge of χ_2 can be a redirected jump. $\square$

By convention, if χ is empty, we set $\chi^- = \chi$. We say χ^- is the path of p *generated by* χ. In the next two subsections, we extend the generation of paths in p from paths in q, first to paths without slipknots, then to arbitrary paths.

3.4.2 Bridges and straight paths

Let $c = \langle t_0 | s_0 \rangle \in \mathbf{C}(p)$ with $t_0 = \otimes(\overrightarrow{t})$ and $s_0 = ⅋(\overrightarrow{s})$. We say an edge e is *bound to* c if either $e = c$, or $e = (t_0, t)$ with $t \in \overrightarrow{t}$, or $e = s_0$. And we say $\chi \in \mathbf{P}(p)$ is bound to c if all the edges crossed by χ are bound to c. Observe that $e \in \mathbf{P}(p)$ is a preserved edge iff it is not bound to an eliminated cut.

Lemma 3.4.4. *Let $c = \langle t_0 | s_0 \rangle$:*

- *if $\chi \in \mathbf{P}(p)$ is bound to $c = \langle t_0 | s_0 \rangle$ and $s_0 = ⅋(\overrightarrow{s})$ then there exists $s \in \overrightarrow{s}$ such that $\chi \in \mathbf{P}(p, I)$ whenever $I(s_0) = s$;*
- *if $\chi \in \mathbf{P}(p, I)$ does not cross any edge bound to c then $\chi \in \mathbf{P}(p, I')$ whenever I and I' differ only on s_0.*

Proof. It is sufficient to observe that the edge s_0 is bound to c; and the only edge $e \in \mathbf{T}_{⅋}(p)$ that may be visited by a path bound to c is s_0. $\square$

A *c-bridge* is a path χ that is bound to c and that crosses c. Observe that χ is a c-bridge iff either χ or the reverse path $\chi^\dagger$ is a subpath of some $t \sim_{(t_0,t)} t_0 \sim_c s_0 \sim_{s_0} s$ with $t \in \overrightarrow{t}$ and $s \in \overrightarrow{s}$. Moreover, given $t \in \{t_0\} \cup \overrightarrow{t}$ and $s \in \{s_0\} \cup \overrightarrow{s}$ there is a unique c-bridge $t \rightsquigarrow s$.

Lemma 3.4.5. *Assume $\chi_1\xi\chi_2 \in \mathbf{P}(q)$ and $\xi = t \sim_{\langle t|s \rangle} s$, where χ_1 and χ_2 are preserved paths, $t \in \overrightarrow{t}$ and $s \in \overrightarrow{s}$. Then there exists a c-bridge $\xi^{\simeq}$ such that $\chi_1^-\xi^{\simeq}\chi_2^- \in \mathbf{P}(p)$.*

Proof. Write $\chi_1 : v_1 \rightsquigarrow_{q,J} t$ and $\chi_2 : s \rightsquigarrow_{q,J} v_2$: by Lemma 3.4.3, we obtain $\chi_1^- : u_1 \rightsquigarrow_{p,I} t'$ and $\chi_2^- : s' \rightsquigarrow_{p,I} u_2$ where u_1, t', s' and u_2 are anchors of v_1, t, s and v_2 respectively, for any extension I of J. In particular, $t' \in \{t, t_0\}$ and $s' \in \{s, s_0\}$ and we can fix $\xi^{\simeq} : t' \rightsquigarrow s'$ to be the only c-bridge with those endpoints. Observe indeed that $\xi^{\simeq} \in \mathbf{P}(p, I)$ as soon as $I(s_0) = s$. Then by Lemmas 3.4.3 and 3.4.4, we can concatenate $\chi_1^- \xi^{\simeq} \chi_2^- : u_1 \rightsquigarrow_{p,I} u_2$. □

Despite the notation, the definition of $\xi^{\simeq}$ does depend on χ_1 and χ_2: whenever we use Lemma 3.4.5, however, the values of χ_1 and χ_2 should be clear from the context.

We say a path $\chi \in \mathbf{P}(q)$ is a *straight path* if it has no slipknot. Such a path is essentially a path of p, up to replacing residuals with bridges:

Lemma 3.4.6. *If χ is a straight path, there exists a unique sequence of pairwise distinct eliminated cuts $c_1, \ldots, c_n \in \mathbf{C}(p) \setminus \mathbf{C}(q)$, such that we can write $\chi = \chi_1(d_1) \cdots \chi_n(d_n)\chi_{n+1}$ where each d_i is a residual of c_i and χ crosses no other residual. Moreover we can form $\chi^- = \chi_1^-(d_1)^{\simeq} \cdots \chi_n^-(d_n)^{\simeq}\chi_{n+1}^- \in \mathbf{P}(p)$.*

Proof. The first part is straightforward reformulation of the absence of slipknots. The second part follows by applying Lemma 3.4.5 to each $\chi_i(d_i)\chi_{i+1}$ (or the reverse path): the concatenation of preserved paths and bridges in the definition of χ^- is allowed by Lemmas 3.4.3 and 3.4.4. □

We say two paths $\chi_1, \chi_2 \in \mathbf{P}(q)$ are *independent* if they are disjoint and there is no eliminated cut c such that both χ_1 and χ_2 cross a residual of c.

Lemma 3.4.7. *Assume $\chi_1, \ldots, \chi_n \in \mathbf{P}(q, J)$ are pairwise independent straight paths and $c_1, \ldots, c_k \in \mathbf{C}(p) \setminus \mathbf{C}(q)$ are such that no χ_i crosses a residual of c_j, for $1 \leq i \leq n$ and $1 \leq j \leq k$. Then for any $\xi_1, \ldots, \xi_k \in \mathbf{P}(p)$ such that ξ_i is bound to c for $1 \leq i \leq k$, there exists an extension I of J such that $\chi_i^- \in \mathbf{P}(p, I)$ for $1 \leq i \leq n$ and $\xi_j \in \mathbf{P}(p, I)$ for $1 \leq j \leq k$.*

Proof. It is sufficient to observe that if χ^- crosses an edge bound to an eliminated cut c then χ crosses a residual of c. Then the result is a direct consequence of the definition of χ_i^-, together with Lemmas 3.4.3 and 3.4.4. □

The generation of a path is thus compatible with the concatenation of independent straight paths:

Lemma 3.4.8. *If $\chi_1 : v \rightsquigarrow_q u$ and $\chi_2 : u \rightsquigarrow_q t$ are independent straight paths then $\chi_1\chi_2$ is a straight path and $(\chi_1\chi_2)^- = \chi_1^-\chi_2^-$.*

Proof. That $\chi_1\chi_2$ is a straight path follows directly from the hypotheses. Write $\chi_i = \chi_1^i(d_1^i) \cdots \chi_{n_i}^i(d_{n_i}^i)\chi_{n_i+1}^i$: it is then sufficient to apply the definition of χ_i^- and observe that $(\chi_{n_1+1}^1\chi_1^2)^- = (\chi_{n_1+1}^1)^-(\chi_1^2)^-$ by Lemma 3.4.3. □

3.4.3 Bounces and slipknots

Let $c = \langle t_0 | s_0 \rangle \in \mathbf{C}(p)$ with $t_0 = \otimes(\overrightarrow{t})$. A *$c$-bounce* is a path χ that is bound to c, that does not cross c and that visits t_0: χ is either the empty path ϵ_{t_0}, or $t_0 \sim_{t_0,t} t$ or $t \sim_{t_0,t} t_0$ with $t \in \overrightarrow{t}$, or $t \sim_{(t_0,t)} t_0 \sim_{(t_0,t')} t'$ with $t \neq t' \in \overrightarrow{t}$. Given $t, t' \in \{t_0\} \cup \overrightarrow{t}$, such that either $t = t_0$ or $t' = t_0$ or $t \neq t'$, there is a unique c-bounce $t \rightsquigarrow t'$.

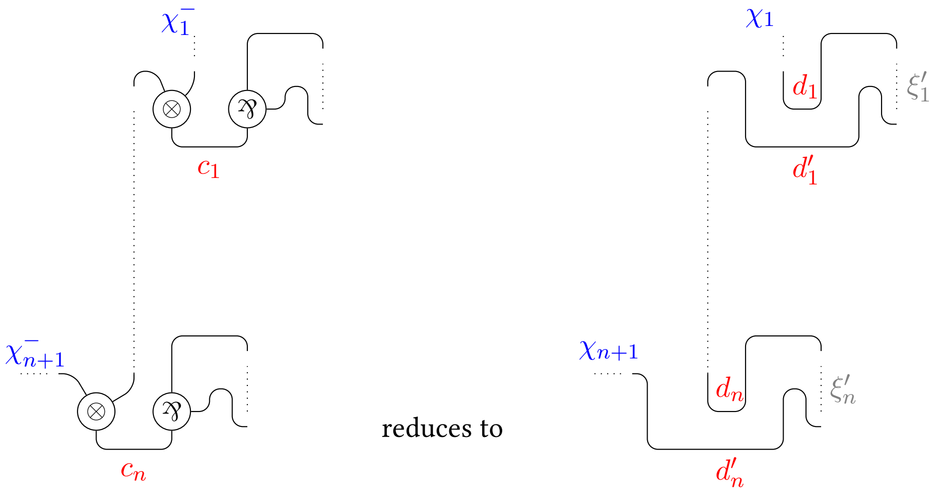

Figure 3.13 – Schematic shape of slipknots on a path (axiom and cut nodes ommited)

Lemma 3.4.9. *Assume $\chi_1\xi\chi_2 \in \mathbf{P}(q)$ and $\xi = (t_1 \sim_{\langle t_1|s_1\rangle} s_1)\xi'(s_2 \sim_{\langle t_2|s_2\rangle} t_2)$, where χ_1 and χ_2 are preserved paths, $t_1, t_2 \in \overrightarrow{t}$ and $s_1, s_2 \in \overrightarrow{s}$. Then there exists a c-bounce $\xi^\vee$ such that $\chi_1^-\xi^\vee\chi_2^- \in \mathbf{P}(p)$.*

Proof. Necessarily, $\langle t_1|s_1\rangle \neq \langle t_2|s_2\rangle$, hence $t_1 \neq t_2$. Write $\chi_1 : v_1 \rightsquigarrow_q t_1$ and $\chi_2 : t_2 \rightsquigarrow_q v_2$: we obtain $\chi_1^- : u_1 \rightsquigarrow_p t_1'$ and $\chi_2^- : t_2' \rightsquigarrow_p u_2$ where u_1, t_1', t_2' and u_2 are anchors of v_1, t_1, t_2 and v_2 respectively, for any extension I of J. In particular, $t_1' \in \{t_1, t_0\}$ and $t_2' \in \{t_2, t_0\}$ with $t_1 \neq t_2$, and we can fix $\xi^\vee : t_1' \rightsquigarrow t_2'$ to be the only the only c-bounce with those endpoints. Then we can concatenate $\chi_1^-\xi^\vee\chi_2^- : u_1 \rightsquigarrow_p u_2$ by Lemmas 3.4.3 and 3.4.4. □

Again, the definition of $\xi^\vee$ does depend on χ_1 and χ_2 but these should be clear from the context when we use Lemma 3.4.9.

We are now ready to prove that paths in q are alternations of straight paths and slipknots, and generate paths in p by replacing slipknots with bounces:

Theorem 3.4.10. *For each path $\chi \in \mathbf{P}(q)$, there exists a unique sequence of pairwise distinct eliminated cuts $c_1, \ldots, c_n \in \mathbf{C}(p) \setminus \mathbf{C}(q)$, such than we can write $\chi = \chi_1\xi_1 \cdots \chi_n\xi_n\chi_{n+1}$ where:*

- *each χ_i is a straight path that crosses no residual of c_j for $1 \leq j \leq n$;*
- *χ_i and χ_j are independent when $i \neq j$;*
- *each ξ_i is a slipknot $(d_i)\xi_i'(d_i') : t_i \rightsquigarrow t_i'$ where d_i and d_i' are residuals of c_i, and t_i and t_i' are distinct premises of the $\otimes$-tree of c_i.*

Moreover $\chi^- = \chi_1^-\xi_1^\vee \cdots \chi_n^-\xi_n^\vee\chi_{n+1}^- \in \mathbf{P}(p)$.

Figure 3.13 illustrates the relationship between χ^- and χ, in the simple case where each cut is between binary connectives, and no redirected jump is involved: each χ_i^- bounces on the $\otimes$ side of c_i and joins χ_{i+1}^- directly instead of crossing the cut.

The proof of Theorem 3.4.10 is by induction on the length of χ. We break it down into a series of intermediate results. Formally, given $\chi \in \mathbf{P}(q)$:

- we first establish Lemma 3.4.11 for all paths such that Theorem 3.4.10 holds;
- we deduce Lemma 3.4.12 for χ from Theorem 3.4.10 and Lemma 3.4.11 applied to χ;
- Lemma 3.4.13 for χ is a direct consequence of Lemma 3.4.12 applied to strict subpaths of χ;
- we prove Lemma 3.4.14 for χ by applying Theorem 3.4.10, Lemma 3.4.11 and Lemma 3.4.13 to strict subpaths of χ;
- then we prove Theorem 3.4.10 for χ by applying Lemmas 3.4.13 and 3.4.14 to χ, and Lemma 3.4.11 to strict subpaths of χ.

Lemma 3.4.11. *Assume $\chi_1, \ldots, \chi_n \in \mathbf{P}(q, J)$ are pairwise disjoint paths, and $c_1, \ldots, c_k \in \mathbf{C}(p) \setminus \mathbf{C}(q)$, are such that no χ_i crosses a residual of c_j, for $1 \leq i \leq n$ and $1 \leq j \leq k$. Then for any $\xi_1, \ldots, \xi_k \in \mathbf{P}(p)$ such that ξ_i is bound to c for $1 \leq i \leq k$, there exists an extension I of J such that $\chi_i^- \in \mathbf{P}(p, I)$ for $1 \leq i \leq n$ and $\xi_j \in \mathbf{P}(p, I)$ for $1 \leq j \leq k$.*

Proof. This is a direct consequence of the definition of χ_i^- in Theorem 3.4.10, together with Lemma 3.4.7. □

Lemma 3.4.12. *Assume $d = \langle t|s\rangle$ and $d' = \langle t'|s'\rangle$ are distinct residuals of the same cut $c = \langle t_0|s_0\rangle$ with $t_0 = \otimes(\overrightarrow{t})$, $s_0 = \invamp(\overrightarrow{s})$, $t, t' \in \overrightarrow{t}$ and $s, s' \in \overrightarrow{s}$. If $u \in d$, $u' \in d'$ and $\chi : u \rightsquigarrow_q u'$ crosses no residual of c then $u = s$ and $u' = s'$.*

Proof. Write $\chi = \chi_1\xi_1 \cdots \chi_n\xi_n\chi_{n+1}$ as in Theorem 3.4.10, and $\chi^- : v \rightsquigarrow_{p,I} v'$ where v (resp. v') is an anchor of u (resp. u'). Observe that χ is non empty because $u \neq u'$. Moreover, the first edge of χ is a preserved edge: if $\chi = (u \sim_e v)\chi'$ then e cannot be a cut, as otherwise we would have $e = d$ and χ would cross a residual of c. It follows that χ_1^- is non empty, hence χ^- is non empty. Since χ^- is non empty, there is no path $\zeta : v \rightsquigarrow v'$ bound to c: otherwise, by Lemma 3.4.11, we could form a non empty cycle $\zeta^\dagger\chi^- : v' \rightsquigarrow_p v'$.

If $u = t$ and $u' = t'$, we have $v \in \{t, t_0\}$ and $v' \in \{t', t_0\}$ with $t \neq t'$, and we obtain a c-bounce $v \rightsquigarrow v'$, hence a contradiction. If $u = t$ and $u' = s'$, we have $v \in \{t, t_0\}$ and $v' \in \{s', s_0\}$, and we obtain a c-bridge $v \rightsquigarrow v'$, hence a contradiction. We rule out the case $u = s$ and $u' = t'$ symmetrically. □

Lemma 3.4.13. *If $\chi \in \mathbf{P}(q)$ and $c = \langle\otimes(\overrightarrow{t})|\invamp(\overrightarrow{s})\rangle \in \mathbf{C}(p) \setminus \mathbf{C}(q)$, then χ crosses at most two residuals of c, and in this case we can write $\chi = \chi_1(t \sim_{\langle t|s\rangle} s)\chi_2(s' \sim_{\langle t'|s'\rangle} t')\chi_3$ with $t, t' \in \overrightarrow{t}$ and $s, s' \in \overrightarrow{s}$.*

Proof. If $\chi = \chi_1(d)\chi_2(d')\chi_3$, where $d = \langle t|s\rangle$ and $d' = \langle t'|s'\rangle$ are residuals of c with $t, t' \in \overrightarrow{t}$ and $s, s' \in \overrightarrow{s}$, and if χ_2 crosses no residual of c, then by Lemma 3.4.12 applied to χ_2, we obtain $\chi = \chi_1(t \sim_{\langle t|s\rangle} s)\chi_2(s' \sim_{\langle t'|s'\rangle} t')\chi_3$. If moreover χ_1 (resp. χ_3) crossed another residual of c, we would obtain a contradiction by applying Lemma 3.4.12 to a strict subpath of χ_1 (resp. χ_3) hence of χ. □

Lemma 3.4.14. *Slipknots are well-bracketed in the following sense: there is no path $\chi = (d_1)\chi_1(d_2)\chi_2(d'_1)\chi_3(d'_2) \in \mathbf{P}(q)$ such that, for $1 \leq i \leq 2$, d_i and d'_i are residuals of the same cut.*

Proof. Assume $\chi = (d_1)\chi_1(d_2)\chi_2(d'_1)\chi_3(d'_2) \in \mathbf{P}(q)$ such that, for $1 \leq i \leq 2$, d_i and d'_i are residuals of the same cut. We can assume w.l.o.g. that χ_1 and χ_3 are independent: otherwise there is a prefix of χ with this additional property.

For $1 \leq i \leq 2$, let $c_i = \langle t^i_0 | s^i_0 \rangle$, with $t^i_0 = \otimes(\overrightarrow{t}_i)$ and $s^i_0 = ⅋(\overrightarrow{s}_i)$, and assume $d_i = \{t_i, s_i\}$ and $d'_i = \{t'_i, s'_i\}$, with $t_i, t'_i \in \overrightarrow{t}_i$ and $s_i, s'_i \in \overrightarrow{s}_i$. By Lemma 3.4.13, we must have $\chi = (t_1 \sim_{d_1} s_1)\chi_1(t_2 \sim_{d_2} s_2)\chi_2(s'_1 \sim_{d'_1} t'_1)\chi_3(s'_2 \sim_{d'_2} t'_2)$.

By Theorem 3.4.10, we obtain $\chi_1^- : u_1 \rightsquigarrow_p v_2$ and $\chi_3^- : v_1 \rightsquigarrow_p u_2$, where $u_1 \in \{s_1, s^1_0\}$, $v_2 \in \{t_2, t^2_0\}$, $v_1 \in \{t'_1, t^1_0\}$ and $u_2 \in \{s'_2, s^2_0\}$. Write $\zeta_1 : v_1 \rightsquigarrow_p u_1$ (resp. $\zeta_2 : v_2 \rightsquigarrow_p u_2$) for the only c_1-bridge (resp. c_2-bridge) with those endpoints. Then we obtain a non empty cycle $\chi_1^- \zeta_2 (\chi_3^\dagger)^- \zeta_1 : u_1 \rightsquigarrow_p u_1$: the concatenation is allowed by Lemma 3.4.11 applied to χ_1 and $\chi_3^\dagger$. ☐

Proof of Theorem 3.4.10. By Lemma 3.4.14, we can write $\chi = \chi_1\xi_1 \cdots \chi_n\xi_n\chi_{n+1}$ where $\xi_1, \ldots, \xi_n$ are the slipknots of χ that are maximal (*i.e.* not strict subpaths of other slipknots) and this writing is unique. Let $c_1, \ldots, c_n$ be the associated eliminated cuts. By Lemma 3.4.13, the c_j's are pairwise distinct, and $\xi_j : t_j \rightsquigarrow_p t'_j$ where t_j and t'_j are distinct premises of the $\otimes$-tree of c_j. Moreover, since each slipknot of χ is a subpath of some ξ_j:

- each χ_i is a straight path and it crosses no residual of any c_j;
- the χ_i's are pairwise independent.

If $n = 0$, we can set $\chi^- = \chi_1^-$. Otherwise, we apply Lemma 3.4.11 to the χ_i's, which are strict subpaths of χ, which allows to concatenate $\chi^- = \chi_1^- \xi_1^\vee \cdots \chi_n^- \xi_n^\vee \chi_{n+1}^- \in \mathbf{P}(p)$. ☐

Again, the construction of χ^- is compatible with the concatenation of independent paths:

Lemma 3.4.15. *If $\chi_1\chi_2 \in \mathbf{P}(q)$ and χ_1 and χ_2 are independent, then $(\chi_1\chi_2)^- = \chi_1^-\chi_2^-$.*

Proof. As for Lemma 3.4.8, this is a direct consequence of the definition of $(\chi_1\chi_2)^-$, χ_1^- and χ_2^-, this time using Lemma 3.4.8 to concatenate a straight suffix of χ_1 and a straight prefix of χ_2. ☐

3.4.4 Bounding the growth of $\ln$

Now we show that we can bound $\mathbf{ln}(q)$ depending only on $\mathbf{ln}(p)$. We first need some basic properties relating the length of χ^- with that of χ.

Lemma 3.4.16. *Let $\chi, \xi \in \mathbf{P}(q)$ and $\zeta \in \mathbf{P}(p)$:*

1. *if χ is preserved then $\mathbf{ln}(\chi^-) = \mathbf{ln}(\chi)$;*
2. *if ζ is a bridge then $1 \leq \mathbf{ln}(\zeta) \leq 3$;*
3. *if ζ is a bounce then $\mathbf{ln}(\zeta) \leq 2$;*
4. *if χ is straight, then $\mathbf{ln}(\chi) \leq \mathbf{ln}(\chi^-) \leq 3\mathbf{ln}(\chi)$;*
5. *if ξ is a prefix of χ and χ is straight then ξ^- is a prefix of χ^-;*

6. *if ξ is a slipknot then* $\mathbf{ln}(\xi) \geq 3$ *and* $\mathbf{ln}(\xi^\vee) < \mathbf{ln}(\xi)$*;*

7. *in general* $\mathbf{ln}(\chi^-) \leq 3\mathbf{ln}(\chi)$.

Proof. The first three properties are direct consequences of the definitions. Item (4) follows from (1) and (2). Item (5) follows from Lemma 3.4.8. Item (6) follows from Lemma 3.4.2 and (3). And item (7) follows from (4) and (6). $\square$

Observe that in general, we do not have $\mathbf{ln}(\zeta^-) \leq \mathbf{ln}(\chi^-)$ when ζ is a prefix of χ: ζ may enter an arbitrarily long slipknot of χ that is replaced by a single bounce in χ^-. For this reason, we introduce the following notion: if $\chi \in \mathbf{P}(q)$, we define the *width* of χ (relative to the reduction $p \rightrightarrows_m q$ we consider) by $\mathbf{width}(\chi) = \max\{\mathbf{ln}(\zeta^-) \mid \zeta \text{ prefix of } \chi\}$.

Lemma 3.4.17. *For any path* $\chi \in \mathbf{P}(q)$, $\mathbf{ln}(\chi^-) \leq \mathbf{width}(\chi) \leq \mathbf{ln}(p)$ *and* $\mathbf{width}(\chi) \leq 3\mathbf{ln}(\chi)$*. Moreover, if ζ is a prefix of χ, we have* $\mathbf{width}(\zeta) \leq \mathbf{width}(\chi)$*. If moreover χ is straight,* $\mathbf{width}(\chi) = \mathbf{ln}(\chi^-) \geq \mathbf{ln}(\chi)$.

Proof. We obtain $\mathbf{ln}(\chi^-) \leq \mathbf{width}(\chi) \leq \mathbf{ln}(p)$ and $\mathbf{width}(\zeta) \leq \mathbf{width}(\chi)$ directly from the definition of width. Item (7) of Lemma 3.4.16 gives $\mathbf{width}(\chi) \leq 3\mathbf{ln}(\chi)$. If χ is straight $\mathbf{width}(\chi) = \mathbf{ln}(\chi^-) \geq \mathbf{ln}(\chi)$ follows from items (4) and (5) of Lemma 3.4.16. $\square$

Define $\varphi : \mathbf{N} \to \mathbf{N}$ inductively by $\varphi(0) = 0$ and $\varphi(n) = n + (n+1)(\varphi(n-1)+2)$ if $n > 0$. Observe that $n \leq \varphi(n) \leq \varphi(n+1)$.

Lemma 3.4.18. *If* $\chi \in \mathbf{P}(q)$ *then* $\mathbf{ln}(\chi) \leq \varphi(\mathbf{width}(\chi))$.

Proof. The proof is by induction on $\mathbf{width}(\chi)$. If χ is straight then, by Lemma 3.4.17, $\mathbf{ln}(\chi) \leq \mathbf{width}(\chi) \leq \varphi(\mathbf{width}(\chi))$.

Write $\chi = \chi_1\xi_1 \cdots \chi_n\xi_n\chi_{n+1}$ as in Theorem 3.4.10: we have $\mathbf{ln}(\chi) = \sum_{i=1}^{n+1} \mathbf{ln}(\chi_i) + \sum_{j=1}^{n} \mathbf{ln}(\xi_j)$. Since $\chi^- = \chi_1^-\xi_1^\vee \cdots \chi_n^-\xi_n^\vee\chi_{n+1}^-$, we have $\sum_{i=1}^{n+1} \mathbf{ln}(\chi_i^-) \leq \mathbf{ln}(\chi^-)$. Since each χ_i is straight, we obtain $\mathbf{ln}(\chi) \leq \mathbf{ln}(\chi^-) + \sum_{j=1}^{n} \mathbf{ln}(\xi_j)$ from the previous inequality, by applying item (4) of Lemma 3.4.16.

Moreover observe that, by Lemma 3.4.2, χ_i is non empty for $1 < i < n+1$. Hence $\mathbf{ln}(\chi^-) \geq n-1$, and we obtain $n \leq \mathbf{width}(\chi) + 1$.

It remains to bound $\mathbf{ln}(\xi_j)$ for $1 \leq j \leq n$. We can write $\xi_j = (d_j)\chi'_j(d'_j)$ where d_j and d'_j are the residuals of the cut c_j associated with ξ_j. Let ζ'_j be a prefix of χ'_j and write $\zeta_j = \chi_1\xi_1 \cdots \chi_j(d_j)\zeta'_j$ which is a prefix of χ. Observe that, by Theorem 3.4.10, c_j has no residual in ζ_j other than d_j, and $\chi_1\xi_1 \cdots \chi_j(d_j)$ and ζ'_j are independent. Hence $\chi_j(d_j)$ is straight and $\zeta_j^- = \chi_1^-\xi_1^\vee \cdots \chi_j^- d_j^{\simeq}(\zeta'_j)^-$ follows by Lemma 3.4.15. Since $\mathbf{ln}(d_j^{\simeq}) \geq 1$, we obtain $\mathbf{ln}((\zeta'_j)^-) \leq \mathbf{ln}(\zeta_j^-) - 1 \leq \mathbf{width}(\chi) - 1$.

Hence $\mathbf{width}(\chi'_j) \leq \mathbf{width}(\chi) - 1$: we apply the induction hypothesis and obtain $\mathbf{ln}(\chi'_j) \leq \varphi(\mathbf{width}(\chi'_j)) \leq \varphi(\mathbf{width}(\chi) - 1)$ because φ is monotonous. It follows that $\mathbf{ln}(\xi_j) \leq$

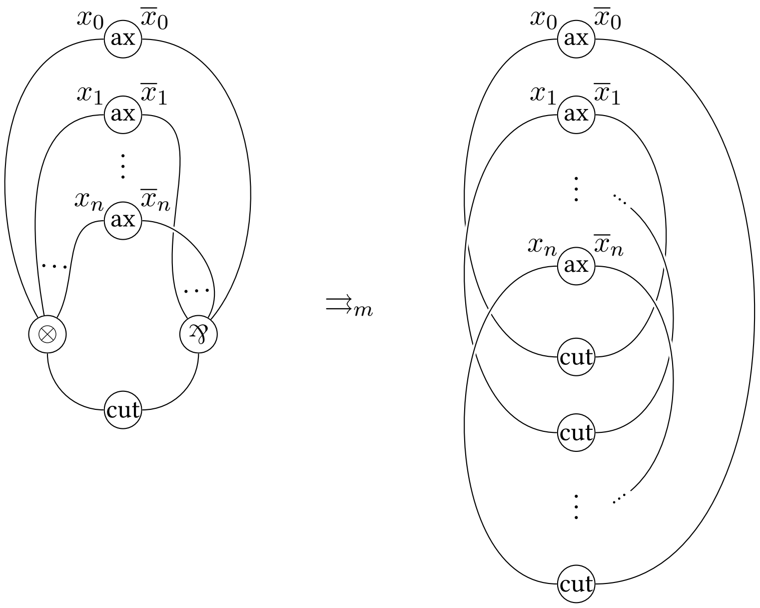

Figure 3.14 – A cyclic counterexample to Corollary 3.4.19

$\varphi(\mathbf{width}(\chi) - 1) + 2$, and we conclude:

$$\begin{aligned}
\mathbf{ln}(\chi) &\leq \mathbf{ln}(\chi^-) + \sum_{j=1}^{n} \mathbf{ln}(\xi_j) \\
&\leq \mathbf{width}(\chi) + (\mathbf{width}(\chi) + 1)(\varphi(\mathbf{width}(\chi) - 1) + 2) \\
&\leq \varphi(\mathbf{width}(\chi)).
\end{aligned}$$

□

Using Lemma 3.4.17 again, we obtain:

Corollary 3.4.19. *Let $p \rightrightarrows_m q$. Then, $\mathbf{ln}(q) \leq \varphi(\mathbf{ln}(p))$.*

Notice that the previous result can be seen as a quantitative version of the preservation of acyclicity in proof nets under reduction. In the following example, we illustrate how acyclicity is mandatory for the existence of a function φ as in Corollary 3.4.19.

Example 3.4.20. *Let $p = (\langle t|s\rangle;)$ with $t = \otimes(x_0, \ldots, x_n)$ and $s = ⅋(\overline{x}_1, \ldots, \overline{x}_n, \overline{x}_0)$: by setting $I(s) = \overline{x}_i$, we obtain a cycle $x_i \sim_{\{x_i,\overline{x}_i\}} \overline{x}_i \sim_s s \sim_{\langle t|s\rangle} t \sim_{t,x_i} x_i$. Since each path $\chi \in \mathbf{P}(p)$ can cross each of s and $\langle t|s\rangle$ at most once, it is easy to check that $\mathbf{ln}(p) = 6$.*

But $p \rightrightarrows_m q = (\langle x_0|\overline{x}_1\rangle, \ldots, \langle x_{n-1}|\overline{x}_n\rangle, \langle x_n|\overline{x}_0\rangle;)$ hence $\mathbf{ln}(q) = 2(n+1)$. The situation is illustrated in Figure 3.14.

3.4.5 Erratum

In the extended abstract of the present paper, the analogue [14, Subsection 3.2] of Subsection 3.4.4 claimed to establish similar results for another measure on paths: rather than the length $\mathbf{ln}(\xi)$ of a path ξ, we considered the number $\mathbf{cc}(\xi)$ of all the cuts $\langle t|s\rangle$ such that ξ visits t or s.

It is easy to check that Lemma 3.3.2 still holds if we replace the maximal length of a path with the maximum number of cuts in a chain of axiom cuts in the sense of Figure 3.2. Given the situation depicted in Figure 3.13, however, it is evident that a bound on the number of cuts crossed by a path cannot be preserved: the path on the left hand side crosses no cut, while the path in the reduct crosses an arbitrary number of (possibly axiom) cuts. We introduced $\mathbf{cc}(\xi)$ in our previous attempt, precisely to capture this example: if ξ is the path following the tensors of the left hand side, then $\mathbf{cc}(\xi) \geq n$. But this fix is actually not sufficient: if we replace each $c_i = \langle t_i|s_i\rangle$ in Figure 3.13, with $\langle \otimes(t_i, s_i)|\rotatebox[origin=c]{180}{\&}(x_i, \overline{x_i})\rangle$, a path in the obtained net can visit at most two of these new cuts, but it reduces to the left hand side, with ξ such that $\mathbf{cc}(\xi) \geq n$.

It might be possible to adapt our method for dealing with a relaxed definition of visited cut: for instance, we might consider the number of cuts c such that ξ visits a tree $t \in \mathbf{T}(c)$ (instead of $t \in c$). But this notion is no longer local, and would introduce further technicalities: for that reason, we decided to focus on the length of paths instead, which is a more intuitive and standard notion, without any *ad hoc* reference to cuts.

3.5 Variations of $\mathbf{jd}(p)$ under reduction

For establishing that $\mathbf{jd}(q)$ is bounded as a function of $\mathbf{jd}(p)$ and $\mathbf{ln}(p)$ we examine the reductions separately.

Lemma 3.5.1. *Let p, q two proof nets. If $p \rightrightarrows_m q$, then $\mathbf{jd}(q) \leq 2\mathbf{jd}(p)$.*

Proof. For all $t \in \mathbf{T}(q)$ the only case in which $\jmath_q^{-1}(t) \neq \jmath_p^{-1}(t)$ is that of redirected jumps: there must be $\mu \in \mathbf{U}_\perp(q)$ such that t is part of a residual $\langle t|s\rangle$ of an eliminated cut $\langle t_0|s_0\rangle \in \mathbf{C}(p)$, with $\jmath_p(\mu) = t_0$. In this case we have $\mathbf{jd}_q(t) = \mathbf{jd}_p(t) + \mathbf{jd}_p(t_0) \leq 2\mathbf{jd}(p)$. □

Lemma 3.5.2. *Let p, q two proof nets. If $p \rightrightarrows_e q$, then $\mathbf{jd}(q) \leq (2\mathbf{jd}(p))^{\mathbf{ln}(p)+1}$.*

Proof. Fix $t \in \mathbf{T}(q)$. For any $\mu \in \mathbf{U}_\perp(q)$, if $\jmath_q(\mu) = t$, then $\mu \in \mathbf{I}_p^n(t)$ for some $n \in \mathbf{N}$: this is precisely the purpose of the definition of $\mathbf{I}_p^n$. We obtain $\mathbf{jd}_q(t) \leq \#\left(\bigcup_{n\in\mathbf{N}} \mathbf{I}_p^n(t)\right) \leq (2\mathbf{jd}(p))^{\mathbf{ln}(p)+1}$ by Lemma 3.3.4 (1). □

Lemma 3.5.3. *Let p, q two proof nets. If $p \rightrightarrows_a q$, then $\mathbf{jd}(q) \leq (\mathbf{ln}(p) + 1)\mathbf{jd}(p)$.*

Proof. As in the proof of Lemma 3.3.2, we can write $p = (\overrightarrow{c_1}, \dots, \overrightarrow{c_k}, \overrightarrow{c}; \overrightarrow{s})$ and $q = (\overrightarrow{c}; \overrightarrow{s})[t^1/\overline{x}_0^1] \cdots [t^k/\overline{x}_0^k]$ where $\overrightarrow{c}_i = (\langle x_0^i|\overline{x}_1^i\rangle, \dots, \langle x_{n_i-1}^i|\overline{x}_{n_i}^i\rangle, \langle x_{n_i}^i|t^i\rangle)$.

By the definition of cut elimination, for all $\mu \in \mathbf{U}_\perp(q)$ we have:

$$\jmath_q(\mu) = \begin{cases} t^i[t^{i+1}/\overline{x}_0^{i+1}] \cdots [t^k/\overline{x}_0^k] & \text{if } \jmath_p(\mu) \in \{x_0^i, \overline{x}_0^i, x_1^i, \dots, \overline{x}_{n_i}^i, t^i\} \\ \jmath_q(\mu)[t^1/\overline{x}_0^1] \cdots [t^k/\overline{x}_0^k] & \text{otherwise} \end{cases}.$$

It follows that:

- $\mathbf{jd}_q(t^i[t^{i+1}/\overline{x}_0^{i+1}]\cdots[t^k/\overline{x}_0^k]) = \mathbf{jd}_p(t^i) + \sum_{j=0}^{n_i}(\mathbf{jd}_p(x_j^i) + \mathbf{jd}_p(\overline{x}_j^i)) \leq (2n_i+3)\mathbf{jd}(p)$;
- if $t \in \mathbf{T}(p) \setminus \bigcup_{i=1}^k \{x_0^i, \overline{x}_0^i, x_1^i, \ldots, \overline{x}_{n_i}^i, t^i\}$, then $\mathbf{jd}_q(t[t^1/\overline{x}_0^1]\cdots[t^k/\overline{x}_0^k]) = \mathbf{jd}_p(t)$.

To conclude, it is sufficient to observe that $2n_i + 2 \leq \mathbf{ln}(p)$: indeed each $\overrightarrow{c}_i$ induces a path alternating between $n_i + 1$ axioms an $n_i + 1$ cuts. □

3.6 Bounding the size of antireducts: general and iterated case

The previous results now allow us to treat the general case of a reduction $p \rightrightarrows q$.

Theorem 3.6.1. *If $p \rightrightarrows q$ then $\mathbf{size}(p) \leq \psi\left(2(\mathbf{ln}(p)+1)\mathbf{size}(q), \mathbf{ln}(p), \mathbf{jd}(p)\right)$.*

Proof. Consider q', q'' such that $p \rightrightarrows_e q' \rightrightarrows_a q'' \rightrightarrows_m q$. We have:

$$\begin{aligned}
\mathbf{size}(p) &\leq \psi(\mathbf{size}(q'), \mathbf{jd}(p), \mathbf{ln}(p)) && \text{(by Lemma 3.3.5)}\\
&\leq \psi((\mathbf{ln}(q')+1)\mathbf{size}(q''), \mathbf{jd}(p), \mathbf{ln}(p)) && \text{(by Lemma 3.3.2)}\\
&\leq \psi((\mathbf{ln}(p)+1)\mathbf{size}(q''), \mathbf{jd}(p), \mathbf{ln}(p)) && \text{(by Lemma 3.4.1)}\\
&\leq \psi(2(\mathbf{ln}(p)+1)\mathbf{size}(q), \mathbf{jd}(p), \mathbf{ln}(p)) && \text{(by Lemma 3.3.1)} \quad .
\end{aligned}$$

□

Corollary 3.6.2. *If q is an MLL net and $n, m \in \mathbf{N}$, then*

$$\{p \mid p \rightrightarrows q,\ \mathbf{jd}(p) \leq m \text{ and } \mathbf{ln}(p) \leq n\}$$

is finite.

Of course, that result holds only up to the α-equivalence mentioned in Remark 3.2.1: then it is easy to check that the cardinality of $\{p \mid \mathbf{size}(p) \leq k\}$ is bounded by a function of $k \in \mathbf{N}$. Also recall from Remark 3.2.2 that we are actually interested in bare nets rather than nets with jumps, so Corollary 3.6.2 should be read as follows: given a bare net q and $n, m \in \mathbf{N}$ there are finitely many bare nets p such that $p \rightrightarrows q$ and that can be equipped with a jump function $\jmath_p$ satisfying $\mathbf{jd}(p) \leq m$ and $\mathbf{ln}(p) \leq n$. More precisely, Theorem 3.6.1 entails that the number of such bare nets p can be bounded by a function of m, n and $\mathbf{size}(q)$.

It follows that, given an infinite linear combination $\sum_{i\in I} a_i.p_i$, assuming that we can equip each p_i with a jump function $\jmath_{p_i}$ so that $\{\mathbf{ln}(p_i) \mid i \in I\} \cup \{\mathbf{jd}(p_i) \mid i \in I\}$ is finite, we can always consider an arbitrary family of reductions $p_i \rightrightarrows q_i$ for $i \in I$ and form the sum $\sum_{i\in I} a_i.q_i$: this is always well defined. But if we want to iterate this process and perform a reduction from $\sum_{i\in I} a_i.q_i$ to $\sum_{i\in I} a_i.r_i$, when $q_i \rightrightarrows r_i$ for $i \in I$, we need to ensure that a similar side condition holds for the q_i's. Again, this is a consequence of our previous results, which we sum up in the following two theorems.

Theorem 3.6.3. *Let $p \rightrightarrows q$. Then $\mathbf{ln}(q) \leq \varphi(\mathbf{ln}(p))$.*

Proof. Consider q', q'' such that $p \rightrightarrows_m q' \rightrightarrows_e q'' \rightrightarrows_a q$. We have:

$$\begin{aligned}
\mathbf{ln}(q) &\leq \mathbf{ln}(q'') && \text{(by Lemma 3.4.1)} \\
&\leq \mathbf{ln}(q') && \text{(by Lemma 3.4.1)} \\
&\leq \varphi(\mathbf{ln}(p)) && \text{(by Corollary 3.4.19)} \quad .
\end{aligned}$$

□

Theorem 3.6.4. *There exists a function $\theta : \mathbf{N} \to \mathbf{N}$ such that $\mathbf{jd}(q) \leq \theta(\mathbf{ln}(p), \mathbf{jd}(p))$ whenever $p \rightrightarrows q$.*

Proof. Consider q', q'' such that $p \rightrightarrows_a q' \rightrightarrows_e q'' \rightrightarrows_m q$. We have

$$\begin{aligned}
\mathbf{jd}(q) &\leq 2\mathbf{jd}(q'') && \text{(by Lemma 3.5.1)} \\
&\leq 2(2\mathbf{jd}(q'))^{\mathbf{ln}(q')+1} && \text{(by Lemma 3.5.2)} \\
&\leq 2(2(\mathbf{ln}(p)+1)\mathbf{jd}(p))^{\mathbf{ln}(q')+1} && \text{(by Lemma 3.5.3)} \\
&\leq 2(2(\mathbf{ln}(p)+1)\mathbf{jd}(p))^{\mathbf{ln}(p)+1} && \text{(by Lemma 3.4.1)} \quad .
\end{aligned}$$

□

By the previous results, we can iterate Corollary 3.6.2 and obtain:

Corollary 3.6.5. *If q is an MLL net and $k, n, m \in \mathbf{N}$, then*

$$\{p \mid p \rightrightarrows^k q,\ \mathbf{jd}(p) \leq m \text{ and } \mathbf{ln}(p) \leq n\}$$

is finite.

3.7 Taylor expansion

We now show how the previous results apply to Taylor expansion. For that purpose, we must extend our syntax to MELL proof nets. Our presentation departs from Ehrhard's [Ehr18] in our treatment of promotion boxes: instead of introducing boxes as tree constructors labelled by nets, with auxiliary ports as inputs, we consider box ports as 0-ary trees, that are related with each other in a *box context*, associating each box with its contents. This is in accordance with the usual presentation of promotion as a black box, and has two motivations:

- in Ehrhard's syntax, the promotion is not a net but an open tree, for which the trees associated with auxiliary ports must be mentioned explicitly: this would complicate the expression of Taylor expansion;
- since we consider a single class of $?$-links instead of having a separate dereliction, we must impose constraints on auxiliary ports, that are easier to express when these ports are directly represented in the syntax.

Then we show that if p is a resource net in the support of the Taylor expansion of an MELL proof net P, then $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$ are bounded by functions of P.

Observe that we need only to consider the support of Taylor expansion, so we do not formalize the expansion of MELL nets into infinite linear combinations of resource nets: rather, we introduce $\mathcal{T}(P)$ as a set of approximants.

3.7.1 MELL nets

In addition to the set of variables, we fix a denumerable set $\mathbf{B}$ of *box ports*: we assume given an enumeration $\mathbf{B} = \{a_i^b \mid i, b \in \mathbf{N}\}$. We call *principal ports* the ports a_0^b and *auxiliary ports* the other ports. Instead of separate contractions and derelictions, we consider a unified $?$-link of arbitrary arity; auxiliary ports of boxes must be premises of such links (or of auxiliary ports of outer boxes, that must satisfy this constraint inductively).

The weakenings (and coweakenings, in the resource nets yet to be introduced) are not essentially different from the multiplicative units in our untyped nets. Indeed, we will see that the geometrical and combinatorial behaviour of the $?$-link (resp. the $!$-link) is identical to that of the $⅋$ (respectively, of the $\otimes$). This will be reflected in our use of labels: in addition to $\mathbf{U_1}$ and $\mathbf{U}_\perp$, we will use labels from denumerable sets $\mathbf{U}_!$ and $\mathbf{U}_?$ (now assuming $\mathbf{V}$, $\mathbf{B}$, $\mathbf{U_1}$, $\mathbf{U}_\perp$, $\mathbf{U}_!$ and $\mathbf{U}_?$ are pairwise disjoint), and write $\mathbf{U}_+ = \mathbf{U_1} \cup \mathbf{U}_!$ and $\mathbf{U}_- = \mathbf{U}_\perp \cup \mathbf{U}_?$.

We introduce the corresponding term syntax. *Raw pre-trees* (S°, T°, *etc.*) and *raw trees* (S, T, *etc.*) are defined by mutual induction as follows:

$$
\begin{aligned}
T &::= x \mid \mathbf{1}_\lambda \mid \perp_\mu \mid \otimes(T_1, \ldots, T_n) \mid ⅋(T_1, \ldots, T_n) \mid a_0^b \mid ?_{\mu'}() \mid ?(T_1^\circ, \ldots, T_n^\circ) \\
T^\circ &::= T \mid a_{i+1}^b
\end{aligned}
$$

where x ranges over $\mathbf{V}$, λ ranges over $\mathbf{U_1}$, μ ranges over $\mathbf{U}_\perp$, μ' ranges over $\mathbf{U}_?$, b and i range over $\mathbf{N}$ and we require $n \neq 0$ in each of $\otimes(T_1, \ldots, T_n)$, $⅋(T_1, \ldots, T_n)$ and $?(T_1^\circ, \ldots, T_n^\circ)$. The set $\mathbf{T}^\circ(S)$ of the sub-pre-trees of S is defined in the natural way, as well as the set $\mathbf{T}(S)$ of sub-trees of S, from which we derive the definitions of $\mathbf{V}(S)$, $\mathbf{B}(S)$, $\mathbf{U}_\perp(S)$, *etc.* The set $\mathbf{A}(S)$ of atoms of S is then $\mathbf{V}(S) \cup \mathbf{B}(S) \cup \mathbf{U_1}(S) \cup \mathbf{U}_-(S)$.

A *tree* (resp. a *pre-tree*) is a raw tree (resp. raw pre-tree) in which each atom occurs at most once. A *cut* is an unordered pair of trees $C = \langle T | S \rangle$ with disjoints sets of atoms. Pre-trees and cuts only describe the surface level of MELL nets: we also have to introduce promotion boxes.

We now define *box contexts* and *pre-nets* by mutual induction as follows. A box context Θ is the data of a finite set $B_\Theta \subset \mathbf{N}$, and of a pre-net of the form $\Theta(b) = (\Theta_b; \overrightarrow{C}_b; T_b, \overrightarrow{S}_b^\circ; \jmath_b)$, for each $b \in B_\Theta$. We then write $\mathbf{ar}_\Theta(b)$, or simply $\mathbf{ar}(b)$ for the length of the family $\overrightarrow{S}_b^\circ$, which we call the *arity* of the box b. A pre-net is a tuple $P^\circ = (\Theta; \overrightarrow{C}; \overrightarrow{S}^\circ; \jmath)$ where:

- Θ is a box context;
- the *jump function* $\jmath$ is a function $\mathbf{U}_-(\overrightarrow{C}, \overrightarrow{S}^\circ) \to \mathbf{T}(\overrightarrow{C}, \overrightarrow{S}^\circ)$;
- each atom occurs at most once in $\overrightarrow{C}, \overrightarrow{S}^\circ$;
- $a_i^b \in \mathbf{B}(\overrightarrow{C}; \overrightarrow{S}^\circ)$ iff $b \in B_\Theta$ and $0 \leq i \leq \mathbf{ar}(b)$;
- $\mathbf{V}(\overrightarrow{C}, \overrightarrow{S}^\circ)$ is closed under the involution $x \mapsto \overline{x}$.

Then a *net* is a pre-net of the form $P = (\Theta; \overrightarrow{C}; \overrightarrow{S}; \jmath)$, *i.e.* without auxiliary ports as conclusions. In the following, we may write, e.g., Θ_P for Θ in this case. An example is illustrated in Figure 3.15.

Remark 3.7.1. *To be formal, in the definition of a box context Θ, we should also fix an enumeration of the family $\overrightarrow{S}_b^\circ$ in $\Theta(b)$. Indeed, when we write $a_0^b, a_1^b, \ldots, a_{\mathbf{ar}(b)}^b$ for the ports of a box b, and*

$\Theta(b) = (\Theta_b; \overrightarrow{C}_b; T_b, S^\circ_{b,1}, \ldots, S^\circ_{b,\mathbf{ar}(b)}; \jmath_b)$ for the contents of the box, we implicitly assume a bijection which maps each pre-tree $S^\circ_{b,i}$ to the auxiliary port a^b_i of which it is a premise (which leaves T to be mapped to a^b_0). We prefer to keep this information implicit in the following, as the notations should allow to recover it, whenever necessary.

On the other hand, an analogue of Remark 3.2.1 applies in this new setting, as pre-nets and nets should be considered up to some notion of isomorphism preserving the interface, which amounts to:

- *reindexing cuts, so that $\overrightarrow{C}$ is considered as a set;*
- *reordering premises of ?-links, which accounts for the associativity and commutativity of the underlying binary contraction;*
- *renaming atoms and boxes and, simultaneously, changing the enumeration of the family $\overrightarrow{S}^\circ_b$ in each box, all this preserving the duality involution on variables, the partition $\{a^b_i \mid i \in \mathbf{N}\}_{b\in\mathbf{N}}$ of $\mathbf{B}$, the jump functions, and the association of each $S^\circ_{b,i+1}$ to a^b_{i+1}.*

We still consider this as a form of α-equivalence as it only involves particular renamings of atoms or indices, preserving the rest of the structure. Again, we keep this quotient implicit whenever possible in the remaining.

Also, as already mentioned in Remark 3.2.2, we rely on jumps to control the combinatorics of the elimination of evanescent cuts: we need nets to be equipped with jumps only to ensure that the resource nets in the Taylor expansion can also be equipped with jumps, that moreover enjoy uniform bounds. More precisely, we will show that if an MELL net P can be equipped with $\jmath_P$ that satisfies the acyclicity criterion, then each $p \in \mathcal{T}(P)$ can be equipped with $\jmath_p$, satisfying uniform bounds on $\mathbf{ln}(p)$ and $\mathbf{jd}(p)$.

The existence of such a jump function $\jmath_P$ should be understood as side condition only: we keep it in the definition of nets by default because we rely on it everywhere in the following, but in the end we are actually interested in the compatibility of Taylor expansion with cut elimination for nets without jumps. And the reader may check that, without jumps, our pre-nets (up to α-equivalence) are essentially the same as, e.g., the in-PS's (up to the names of internal ports) defined by de Carvalho [Car16] for his proof of the injectivity of Taylor expansion.

Given a pre-net $P^\circ = (\Theta; \overrightarrow{C}; \overrightarrow{S}^\circ; \jmath)$, we write $\mathbf{V}(P^\circ) = \mathbf{V}(\overrightarrow{C}; \overrightarrow{S}^\circ)$, $\mathbf{T}(P^\circ) = \mathbf{T}(\overrightarrow{C}; \overrightarrow{S}^\circ)$, *etc.* We define the *toplevel size* of MELL pre-nets by $\mathbf{size}_0(P^\circ) = \#\mathbf{T}^\circ(P^\circ)$. We write $\mathbf{depth}(P^\circ)$ for the maximum level of nesting of boxes in P°, *i.e.* the inductive depth in the above definition of pre-nets. The *size* of MELL pre-nets includes that of their boxes: we set $\mathbf{size}(P^\circ) = \mathbf{size}_0(P^\circ) + \sum_{b\in B_\Theta} \mathbf{size}(\Theta(b))$ — this definition is of course by induction on $\mathbf{depth}(P^\circ)$.

Notice that, by the above definition, for all $\mu \in \mathbf{U}_-(\overrightarrow{C}, \overrightarrow{S}^\circ)$, $\jmath_{P^\circ}(\mu)$ must be at the same depth as μ, and cannot be an auxiliary port.

We extend the switching functions of MLL to ?-links: for each $T = ?(T^\circ_1, \ldots, T^\circ_n) \in \mathbf{T}(P^\circ)$, $I(T) \in \{T^\circ_1, \ldots, T^\circ_n\}$, which induces a *?-edge* $T \sim^{P^\circ,I}_T I(T)$. We also consider *box edges* $a^b_0 \sim^{P^\circ}_{b,i} a^b_i$ for $b \in B_\Theta$ and $1 \leq i \leq ar(b)$: w.r.t. paths, a box b behaves like $\mathbf{ar}(b)$ axiom links having the principal port of the box as a common vertice, and the content is not considered. Finally, *jump edges* also include the case of weakenings: $\mu \sim^{P^\circ}_\mu \jmath(\mu)$ for $\mu \in \mathbf{U}_-(P^\circ)$.

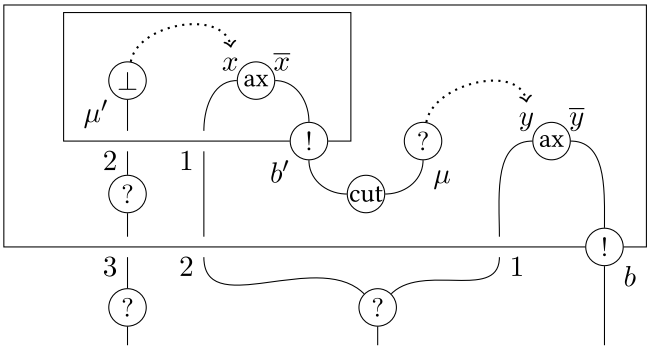

Figure 3.15 – Representation of the net $(\Theta;\,;?(a_3^b), ?(a_2^b, a_1^b), a_0^b; \jmath)$ where $B_\Theta = \{b\}$, $\mathbf{ar}(b) = 3$ and $\Theta(b) = (\Theta'; \langle a_0^{b'} | \mu\rangle; \overline{y}, y, a_1^{b'}, ?(a_2^{b'}); \jmath')$ where $B_{\Theta'} = \{b'\}$, $\mathbf{ar}(b') = 2$, $\jmath'(\mu) = y$ and $\Theta'(b') = (\Theta'';\,; \overline{x}, x, \mu'; \jmath'')$ where $B_{\Theta''} = \emptyset$ and $\jmath''(\mu') = x$.

We write $\mathbf{P}(P^\circ, I)$ (resp. $\mathbf{P}(P^\circ)$) for the set of I-paths (resp. paths) in P°. We say a pre-net P° is *acyclic* if there is no cycle in $\mathbf{P}(P^\circ)$ and, inductively, each $\Theta(b)$ is acyclic. From now on, we consider acyclic pre-nets only.

3.7.2 Resource nets and Taylor expansion

The Taylor expansion of a net P will be a set of *resource nets*: these are the same as the multiplicative nets introduced before, with the addition of term constructors for ! and ?. Raw trees are given as follows:

$$t ::= x \mid \mathbf{1}_\lambda \mid \bot_\mu \mid \otimes(t_1, \ldots, t_n) \mid ⅋(t_1, \ldots, t_n) \mid !_{\lambda'}() \mid ?_{\mu'}() \mid !(t_1, \ldots, t_n) \mid ?(t_1, \ldots, t_n).$$

where x ranges in $\mathbf{V}$, λ ranges over $\mathbf{U_1}$, μ ranges over $\mathbf{U}_\bot$, λ' ranges over $\mathbf{U}_!$, μ' ranges over $\mathbf{U}_?$, and we require $n \neq 0$ in each case. In resource nets, we extend switchings to ?-links and jumps from weakenings as in MELL nets, associated with ?-edges and jump edges. Moreover, for each $t = !(t_1, \ldots, t_n)$, we have *!-edges* $t \sim_{t,t_i} t_i$ for $1 \leq i \leq n$. Observe that, except for the notation of the root constructor, the trees $!_\lambda()$, $?_\mu()$, $!(t_1, \ldots, t_n)$ and $?(t_1, \ldots, t_n)$, are exactly the same as $\mathbf{1}_\lambda$, $\bot_\mu$, $\otimes(t_1, \ldots, t_n)$ and $⅋(t_1, \ldots, t_n)$ respectively: in particular they induce the same geometry for paths.

During Taylor expansion, we need to replace a box in a pre-net with an arbitrary number of approximants of this box. Let us call *box replacement of arity* n the data $r = (\overrightarrow{s}_0, \ldots, \overrightarrow{s}_n)$ of $n+1$ families of pairwise distinct resource trees $\overrightarrow{s}_0, \ldots, \overrightarrow{s}_n$, such that $\mathbf{A}(s) \cap \mathbf{A}(s') = \emptyset$ whenever $s \in \overrightarrow{s}_i$ and $s' \in \overrightarrow{s}_{i'}$, and $i \neq i'$ or $s \neq s'$. A family $\overrightarrow{r} = (r_b)_{b \in B}$ of box replacements such that $\mathbf{A}(r_b) \cap \mathbf{A}(r_{b'}) = \emptyset$ for $b \neq b' \in B$ is *applicable to the pre-term* T° if $\mathbf{A}(T^\circ) \cap \mathbf{A}(\overrightarrow{r}) = \emptyset$ and, for each $a_i^b \in \mathbf{B}(T^\circ)$, $b \in B$ and r_b is of arity at least i.

Definition 3.7.2. Let $\overrightarrow{r}$ be a B-indexed family of box replacements, and write $r_b = (\overrightarrow{s}_0^b, \ldots, \overrightarrow{s}_{n_b}^b)$ for each $b \in B$. Assuming that $\overrightarrow{r}$ is applicable to the tree S (resp. the pre-tree S°), the *substitution of $\overrightarrow{r}$ for the boxes of* S (resp. of S°) is the tree $S[\overrightarrow{r}]$ (resp. the family of pre-trees $S^\circ\{\overrightarrow{r}\}$)

defined by mutual induction on pre-trees and trees as follows:

$$x[\vec{r}] = x \quad \lambda[\vec{r}] = \lambda \quad \mu[\vec{r}] = \mu \quad a_0^b[\vec{r}] = \begin{cases} \lambda_b & \text{if } \vec{s}_0^{\,b} \text{ is empty} \\ !(\vec{s}_0^{\,b}) & \text{otherwise} \end{cases}$$

$$\otimes(T_1, \dots, T_n)[\vec{r}] = \otimes(T_1[\vec{r}], \dots, T_n[\vec{r}])$$
$$\parr(T_1, \dots, T_n)[\vec{r}] = \parr(T_1[\vec{r}], \dots, T_n[\vec{r}])$$
$$?(T_1^\circ, \dots, T_n^\circ)[\vec{r}] = \begin{cases} \mu_{?(T_1^\circ,\dots,T_n^\circ)} & \text{if } T_1^\circ\{\vec{r}\}, \dots, T_n^\circ\{\vec{r}\} \text{ is empty} \\ ?(T_1^\circ\{\vec{r}\}, \dots, T_n^\circ\{\vec{r}\}) & \text{otherwise} \end{cases}$$
$$T\{\vec{r}\} = T[\vec{r}] \qquad a_{i+1}^b\{\vec{r}\} = \vec{s}_{i+1}^{\,b}$$

where each $\lambda_b \in \mathbf{U}_!$ and each $\mu_{?(T_1^\circ,\dots,T_n^\circ)} \in \mathbf{U}_?$ is chosen fresh (not in $\mathbf{A}(S)$ nor $\mathbf{A}(S^\circ)$ nor $\mathbf{A}(\vec{r})$) and unique. [10]

We are now ready to introduce the expansion of MELL nets depicted in Figure 3.1. [11] During the construction, we need to track the conclusions of copies of boxes, in order to collect copies of auxiliary ports in the external ?-links: this is the role of the intermediate notion of pre-Taylor expansion.

First, recall that we write $T_b, S_{b,1}^\circ, \dots, S_{b,\mathbf{ar}(b)}^\circ$ for the trees of $\Theta(b)$ that are respectively mapped to $a_0^b, a_1^b, \dots, a_{\mathbf{ar}(b)}^b$. Also, in this case, let us write $\vec{S}_b^\circ = (S_{b,1}^\circ, \dots, S_{b,\mathbf{ar}(b)}^\circ)$.

Definition 3.7.3. Given a closed pre-net $P^\circ = (\Theta; \vec{C}; \vec{S}^\circ; \jmath)$, a *pre-Taylor expansion* of P° is any pair (p, f) of a resource net $p = (\vec{c}; \vec{t}; \jmath_p)$, together with a function $f : \vec{t} \to \vec{S}^\circ$ such that $f^{-1}(T)$ is a singleton whenever $T \in \vec{S}^\circ$ is a tree, obtained as follows:

- for each $b \in B_\Theta$, fix a number $k_b \geq 0$ of copies;
- for $1 \leq j \leq k_b$, fix inductively a pre-Taylor expansion (p_j^b, f_j^b) of $\Theta(b)$, renaming the atoms so that the sets $\mathbf{A}(p_j^b)$ are pairwise disjoint, and also disjoint from $\mathbf{A}(\vec{C}) \cup \mathbf{A}(\vec{S}^\circ)$;
- write $p_j^b = (\vec{c}_j^{\,b}; t_j^b, \vec{s}_j^{\,b}; \jmath_j^b)$ so that $f_j^b(t_j^b) = T_b$;
- write $\vec{r} = (r_b)_{b \in B_\Theta}$ for the family of box replacements $r_b = (\vec{u}_0^{\,b}, \dots, \vec{u}_{\mathbf{ar}(b)}^{\,b})$, where $\vec{u}_0^{\,b} = (t_1^b, \dots, t_{k_b}^b)$ and each $\vec{u}_i^{\,b}$ is an enumeration of $\bigcup_{j=1}^{k_b} (f_j^b)^{-1}(S_{b,i}^\circ)$ for $1 \leq i \leq \mathbf{ar}(b)$;
- set $\vec{t} = \vec{S}^\circ\{\vec{r}\}$ and $\vec{c} = \vec{C}[\vec{r}], \vec{c}\,'$ where $\vec{c}\,'$ is the concatenation of the families $\vec{c}_j^{\,b}$ for $b \in B_\Theta$ and $1 \leq j \leq k_b$

10. So, formally, this construction should be parametrized by suitable injections $\{a_0^b \in \mathbf{T}(S^\circ)\} \to \mathbf{U}_!$ and $\{?(T_1^\circ, \dots, T_n^\circ) \in \mathbf{T}(S^\circ)\} \to \mathbf{U}_?$ to ensure this linearity constraint. We keep this implicit in the following, but will rely on the fact that, given $t \in \mathbf{T}(S^\circ\{\vec{r}\})$, one can recover unambiguously one of the following: either $T \in \mathbf{T}(S^\circ)$ such that $t = T[\vec{r}]$; or b and j such that $t \in \mathbf{T}(\vec{s}_j^{\,b})$.

11. More extensive presentations of the Taylor expansion of MELL nets exist in the literature, in various styles [PT09; GPT16; Car16, among others]. Our only purpose here is to introduce sufficient notations to present our analysis of the jump degree and the length of paths in $\mathcal{T}(P)$ w.r.t. the size of P.

– for $t \in \overrightarrow{t}$, set $f(t) = a_i^b$ if $t \in \overrightarrow{u}_i^b$ with $1 \leq i \leq \mathbf{ar}(b)$, otherwise let $f(t)$ be the tree $T \in \overrightarrow{S}^\circ$ such that $t = T[\overrightarrow{r}]$;

– for each $\mu \in \mathbf{U}_-(p)$, $\jmath_p(\mu)$ is defined as follows:

  – if $\mu \in \mathbf{U}_-(p_j^b)$ then we set $\jmath_p(\mu) = \jmath_{p_j^b}(\mu)$.

  – if $\mu = \mu_{?(T_1^\circ,\ldots,T_n^\circ)}$ then each $T_i^\circ\{\overrightarrow{r}\}$ is empty; then we select any $i \in \{1,\ldots,n\}$ and set $\jmath_p(\mu) = a_0^b[\overrightarrow{r}]$ where $b \in B_\Theta$ is the box such that $T_i^\circ = a_j^b$ for some $1 \leq j \leq \mathbf{ar}(b)$;

  – otherwise $\mu \in \mathbf{U}_-(\overrightarrow{C}; \overrightarrow{S}^\circ)$, and then we set $\jmath_p(\mu) = \jmath(\mu)[\overrightarrow{r}]$ (note that $\jmath(\mu)$ is a tree so this is a valid application of Definition 3.7.2).

The *Taylor expansion* of a net P is then $\mathcal{T}(P) = \{p \mid (p, f) \text{ is a pre-Taylor expansion of } P\}$.

Example 3.7.4. *Given the net $P = (\Theta; ; ?(a_3^b), ?(a_2^b, a_1^b), a_0^b; \jmath)$ of Figure 3.15, we construct an element p of $\mathcal{T}(P)$ as follows. First, we take two copies of box b, fixing $k_b = 2$. Recall that $\Theta(b) = (\Theta'; \langle a_0^{b'}|\mu\rangle; \overline{y}, y, a_1^{b'}, ?(a_2^{b'}); \jmath')$ where $B_{\Theta'} = \{b'\}$. Hence, to construct (p_j^b, f_j^b) we must first fix a number $k_{b',j}$ of copies of the box b': we set $k_{b',1} = 0$ and $k_{b',2} = 1$, and it remains to select a single pre-Taylor expansion (p', f') of $\Theta'(b')$ for the only copy of b' in (p_2^b, f_2^b). Since $\Theta'(b') = (\Theta''; ; \overline{x}, x, \mu'; \jmath'')$ contains no box, we must have $p' = (; \overline{x}, x, \mu'; \jmath'')$ with $f'(\overline{x}) = \overline{x}$, $f'(x) = x$ and $f'(\mu') = \mu'$.*

Since $k_{b',1} = 0$, we construct $p_1^b = \left(\langle a_0^{b'}|\mu\rangle\{r_{b',1}\}; (\overline{y}, y, a_1^{b'}, ?(a_2^{b'}))\{r_{b',1}\}; \jmath_1^b\right)$ where $r_{b',1}$ is the empty replacement: we obtain $p_1^b = (\langle\lambda|\mu\rangle; \overline{y}, y, \mu'')$ where $\lambda \in \mathbf{U}_!$ and $\mu'' \in \mathbf{U}_?$ are fresh, and we set $f_1^b(\overline{y}) = \overline{y}$, $f_1^b(y) = y$, $f_1^b(\mu'') = ?(a_2^{b'})$, and also $\jmath_1^b(\mu) = \jmath'(\mu)[r_{b',1}] = y$ and $\jmath_1^b(\mu'') = a_0^{b'}[r_{b',1}] = \lambda$.

Having defined (p', f') as above, we must set $r_{b',2} = ((\overline{x}), (x), (\mu'))$ and we define $p_2^b = \left(\langle a_0^{b'}|\mu\rangle\{r_{b',2}\}; (\overline{y}, y, a_1^{b'}, ?(a_2^{b'}))\{r_{b',2}\}; \jmath_2^b\right)$: we thus obtain $p_2^b = (\langle !(\overline{x})|\mu\rangle; \overline{y}, y, x, ?(\mu'))$, with $f_2^b(\overline{y}) = \overline{y}$, $f_2^b(y) = y$, $f_2^b(x) = a_1^{b'}$ and $f_2^b(?(\mu')) = ?(a_2^{b'})$, and we set $\jmath_2^b(\mu) = \jmath'(\mu)[r_{b',2}] = y$ and $\jmath_2^b(\mu') = \jmath''(\mu') = x$.

We rename the atoms in both pre-Taylor expansions as follows: $p_1^b = (\langle\lambda_1|\mu_1\rangle; \overline{y}_1, y_1, \mu_1'')$ and $p_2^b = (\langle !(\overline{x}_2)|\mu_2\rangle; \overline{y}_2, y_2, x_2, ?(\mu_2'))$, also redefining f_1^b, $\jmath_1^b$, f_2^b and $\jmath_2^b$ accordingly.

Finally, we set $\overrightarrow{c} = \langle\lambda_1|\mu_1\rangle, \langle !(\overline{x}_2)|\mu_2\rangle$ and $\overrightarrow{t} = (?(a_3^b), ?(a_2^b, a_1^b), a_0^b)\{r_b\}$ where $r_b = ((\overline{y}_1, \overline{y}_2), (y_1, y_2), (x_2), (\mu_1'', ?(\mu_2')))$. We obtain:

$$p = (\langle\lambda_1|\mu_1\rangle, \langle !(\overline{x}_2)|\mu_2\rangle; ?(\mu_1'', ?(\mu_2')), ?(x_2, y_1, y_2), !(\overline{y}_1, \overline{y}_2); \jmath_p)$$

with $\jmath_p(\mu_1) = y_1$, $\jmath_p(\mu_2) = y_2$, $\jmath_p(\mu_1'') = \lambda_1$ and $\jmath_p(\mu_2') = x_2$, which is depicted in Figure 3.16.

3.7.3 Paths in Taylor expansion

In the following, we fix a pre-Taylor expansion (p, f) of $P^\circ = (\Theta; \overrightarrow{C}; \overrightarrow{S}^\circ; \jmath)$ and we describe the structure of paths in p. We show that the critical case depicted in Figure 3.17 is maximal, so that a path of p passes through at most two copies of each box of P°.

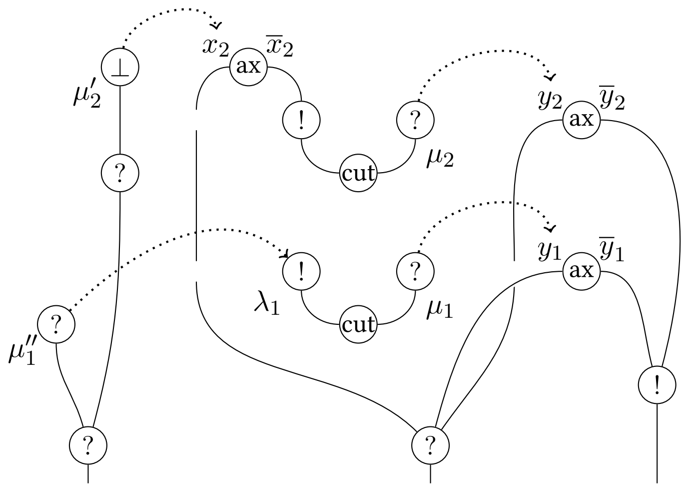

Figure 3.16 – Representation of the resource net $p \in \mathcal{T}(P)$ where P is the net of Figure 3.15, $k_b = 2$, $k_{b',1} = 0$ and $k_{b',2} = 1$.

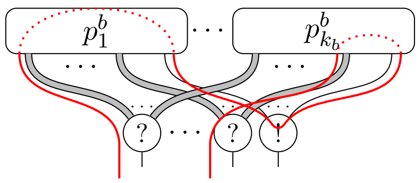

Figure 3.17 – Box paths in Taylor expansion of P°: critical case

Observe that

$$\mathbf{T}(p) = \{T[\overrightarrow{r}] \mid T \in \mathbf{T}(\overrightarrow{C}, \overrightarrow{S}^\circ)\} \cup \bigcup_{b \in B_\Theta} \bigcup_{j=1}^{k_b} \mathbf{T}(p_j^b)$$

(using the notations of Definition 3.7.3). It follows that, for each $t \in \mathbf{T}(p)$:

- either t is in a copy of a box, *i.e.* (up to α-equivalence) $t \in \mathbf{T}(p_j^b)$ for some $b \in B_\Theta$ and $1 \leq j \leq k_b$, and then we say t is *inner* and write $\beta(t) = b$ and $\iota(t) = (b, j)$;
- or there exists a unique $T \in \mathbf{T}(P^\circ)$ such that $t = T[\overrightarrow{r}]$, and then we say t is *outer*, and write $t^* = T$.

We further distinguish the *cocontractions* of p, *i.e.* the outer trees $!(t_1^b, \ldots, t_{k_b}^b)$ for $b \in B_\Theta$, which we denote by $!_b$, so that $!_b^* = a_0^b$.

We say an edge $t \sim_e^{p,I} s$ of p is an *inner edge* (resp. an *outer edge*) if t and s are both inner (resp. outer) trees. We say a path $\xi \in \mathbf{P}(p)$ is an *inner path* (resp. an *outer path*) if it crosses inner edges (resp. outer edges) only.

If $t \sim_e^{p,I} s$ is an inner edge then $\iota(t) = \iota(s)$ and we also have $t \sim_e^{p_j^b, I_j^b} s$ where $(b,j) = \iota(t)$ and I_j^b is the restriction of I to $\mathbf{U}_-(p_j^b)$. In this case, we also set $\beta(e) = b$ and $\iota(e) = (b,j)$. If ξ is an inner path, we set $\beta(\xi)$ (resp. $\iota(\xi)$) for the common value of β (resp. ι) on the edges crossed by ξ, and we obtain:

Lemma 3.7.5. *If ξ is an inner path then $\xi \in \mathbf{P}(p_j^b, I_j^b)$ where $(b,j) = \iota(\xi)$.*

The classification of the outer edges of p is more delicate. First, we associate a switching I^* of P° with each switching I of p as follows:

- if $I(\parr(T_1[\overrightarrow{r}], \dots, T_n[\overrightarrow{r}])) = T_i[\overrightarrow{r}]$, we set $I^*(\parr(T_1, \dots, T_n)) = T_i$;
- if $I(?(T_1^\circ\{\overrightarrow{r}\}, \dots, T_n^\circ\{\overrightarrow{r}\})) \in T_i^\circ\{\overrightarrow{r}\}$, we set $I^*(?(T_1^\circ, \dots, T_n^\circ)) = T_i^\circ$;
- if $?(T_1^\circ, \dots, T_n^\circ)[\overrightarrow{r}] = \mu$, $T_i^\circ = a_j^b$ and $\jmath_p(\mu) = !_b$, we set $I^*(?(T_1^\circ, \dots, T_n^\circ)) = T_i^\circ$. [12]

If $t \sim_e^{p,I} s$ is an outer edge then, in each of the following cases, we can define an I^*-edge e^* of P° such that $e = e^*[\overrightarrow{r}]$ and $t^* \sim_{e^*}^{P^\circ, I^*} s^*$:

- e is an axiom edge, and we set $e^* = e$;
- e is a $\otimes$-edge, e.g. $t = \otimes(t_1, \dots, t_n)$ and $s = t_i$, and we set $e^* = (t^*, t_i^*)$;
- e is a $\parr$-edge or a $?$-edge, e.g. $t = \parr(t_1, \dots, t_n)$, $s = t_i$ and $I(t) = s$, and we set $e^* = t^*$;
- $e = \langle t | s \rangle$ is a cut edge, and we set $e^* = \langle t^* | s^* \rangle$;
- $e = \mu \in \mathbf{U}_-(P^\circ) \subset \mathbf{U}_-(p)$ and we set $e^* = e$ (observe that in this case we have $\jmath_{P^\circ}(\mu) = \jmath_p(\mu)^*$).

If any of the above cases holds, we say the outer edge e is *superficial*.

If e is an outer edge that is not superficial then e must be a *created jump*: $e = \mu \in \mathbf{U}_?(p) \setminus \left(\mathbf{U}_?(P^\circ) \cup \bigcup_{b \in B_\Theta} \bigcup_{j=1}^{k_b} \mathbf{U}_?(p_j^b)\right)$. If, e.g., $t = \mu$, then we can write $t^* = ?(T_1^\circ, \dots, T_n^\circ)$ and $s = \jmath_p(t) = !_b$ where b is such that $I^*(t^*) = a_i^b$ with $1 \leq i \leq \mathbf{ar}(b)$. In this case, we obtain a path $\hat{e} = t^* \sim_{t^*}^{P^\circ, I^*} a_i^b \sim_{b,i}^{P^\circ, I^*} a_0^b = s^*$.

Lemma 3.7.6. *If ξ is an outer I-path in p, then there exists an I^*-path ξ^* in P° with $\mathbf{ln}(\xi^*) \geq \mathbf{ln}(\xi)$.*

Proof. It is sufficient to replace each outer edge e crossed by ξ with:

- either e^* if e is superficial,
- or the path $\hat{e}$ or $\hat{e}^\dagger$ if e is a created jump.

Observe indeed that if $t \sim_e^{p,I} s$ and $t' \sim_{e'}^{p,I} s'$ are outer paths of length 1 with $e \neq e'$ then the paths $(t \sim_e^{p,I} s)^* : t^* \rightsquigarrow_{P^\circ, I^*} s^*$ and $(t' \sim_{e'}^{p,I} s')^* : t'^* \rightsquigarrow_{P^\circ, I^*} s'^*$ thus defined are disjoint, and of length at least 1. ☐

Some edges are neither inner nor outer: a *boundary edge* is an edge $t \sim_e^{p,I} s$ such that t is outer and s is inner, in which case we set $\iota(e) = \iota(s)$. There are two kinds of boundary edges:

12. Observe that there might be several possible choices for T_i° so I^* is not uniquely defined in this manner: our following constructions thus depend on the choices we make for I^*.

- the *principal boundary* of the box copy (b, j) is the !-edge $(!_b, t_j^b)$;
- an *auxiliary boundary* e of the box copy (b, j) is any ?-edge $t \sim_t^{p,I} s$ where $I(t) = s \in \overrightarrow{s}_j^b$ is such that $f_j^b(s) = S_{b,i}^\circ$ with $1 \leq i \leq \mathbf{ar}(b)$, in which case we must have $I^*(t^*) = a_i^b$, and then we write $\lceil e \rceil = i$ for the index of the corresponding auxiliary port.

We call *box path* any path of the form $\chi = (e)\xi(e')$ where e and e' are boundaries and ξ is an inner path: in this case, we write $\beta(\chi) = \beta(\xi)$ and $\iota(\chi) = \iota(\xi)$. Obviously, any path ξ with outer endpoints is obtained as an alternation of outer paths and box paths: we can write uniquely $\xi = \xi_0\chi_1\xi_1 \cdots \chi_n\xi_n$ where each ξ_i is an outer path, and each χ_i is a box path.

Let $\chi = (e)\xi(e') : t \rightsquigarrow_p s$ be a box path with $\iota(\chi) = (b, j)$. Since $e \neq e'$, then at most one of e and e' is the principal boundary, and if both e and e' are auxiliary boundaries, then we must have $\lceil e \rceil \neq \lceil e' \rceil$: indeed e is the only ?-edge whose premises include $a_{\lceil e \rceil}^b\{\overrightarrow{r}\}$. We can thus define $\chi^* : t^* \rightsquigarrow_{P^\circ} s^*$ as follows:

- if e and e' are auxiliary boundaries then $\chi^* = t^* \sim_{t^*} a_{\lceil e \rceil}^b \sim_{b,\lceil e \rceil} a_0^b \sim_{b,\lceil e' \rceil} a_{\lceil e' \rceil}^b \sim_{s^*} s^*$
- otherwise, e.g. e' is principal and we set $\chi^* = t^* \sim_{t^*} a_{\lceil e \rceil}^b \sim_{b,\lceil e \rceil} a_0^b = s^*$.

Lemma 3.7.7. *Assume $\xi = \xi_0\chi_1\xi_1 \cdots \chi_n\xi_n : t \rightsquigarrow_{p,I} s$ where each ξ_i is an outer path, and each χ_i is a box path. Then, setting $\xi^* = \xi_0^*\chi_1^*\xi_1^* \cdots \chi_n^*\xi_n^*$ we obtain $\xi^* : t^* \rightsquigarrow_{P^\circ,I^*} s^*$. Moreover, if $\beta(\chi_i) = \beta(\chi_j) = b$ and $i < j$, then $j = i + 1$, $\xi_i = \epsilon_{!_b}$, and $\iota(\chi_i) \neq \iota(\chi_j)$.*

Proof. We have already observed in the proof of Lemma 3.7.6 that if ξ and ξ' are disjoint outer paths then ξ^* and ξ'^* are also disjoint. Similarly, if ξ is outer and χ is a box path, it follows directly from the definitions that ξ^* and χ^* are disjoint. And if χ and χ' are box paths with disjoint boundaries, again χ^* and χ'^* are disjoint paths by construction. It follows that, if $\xi = \xi_0\chi_1\xi_1 \cdots \chi_n\xi_n : t \rightsquigarrow_{p,I} s$, each ξ_i is an outer path, and each χ_i is a box path, then the concatenation $\xi^* = \xi_0^*\chi_1^*\xi_1^* \cdots \chi_n^*\xi_n^*$ is well defined.

Write $\chi_i = (e_i)\xi_i'(e_i') : t_i \rightsquigarrow s_i$ for $1 \leq i \leq n$. Assume $\beta(\chi_i) = \beta(\chi_j) = b$, and moreover $\beta(\chi_k) \neq b$ for $i < k < j$. We obtain a path $\xi' = (\xi_i\chi_{i+1} \cdots \xi_{j-1})^* : s_i^* \rightsquigarrow_{P^\circ} t_j^*$: by construction, ξ' does not cross any box edge (b, l) for $1 \leq l \leq \mathbf{ar}(b)$. If (e_i') and (e_j) were both auxiliary, we could form a cycle $\xi'(t_j^* \sim_{t_j^*} a_{\lceil e_j \rceil}^b \sim_{b,\lceil e_j \rceil} a_0^b \sim_{b,\lceil e_i' \rceil} a_{\lceil e_i' \rceil}^b \sim_{s_i^*} s_i^*)$, since ξ' would cross neither t_j^* nor s_i^*. If, e.g., (e_i') was principal and (e_j) was auxiliary, we could form a cycle $\xi'(t_j^* \sim_{t_j^*} a_{\lceil e_j \rceil}^b \sim_{b,\lceil e_j \rceil} a_0^b)$, as ξ' would not cross t_j^*. So both must be principal and we have $s_i^* = t_j^* = a_0^b$: since P° has no non empty cycle, we must have $\xi' = \epsilon_{a_0^b}$ hence $\xi_i\chi_{i+1} \cdots \xi_{j-1} = \epsilon_{!_b}$ and then $j = i + 1$ and $\xi_i = \epsilon_{!_b}$. Since $e_i' \neq e_j$, we moreover obtain $\iota(\chi_i) \neq \iota(\chi_j)$.

It remains only to prove that, in general, we never have $\beta(\chi_i) = \beta(\chi_j)$ with $j > i + 1$: otherwise, by iterating our previous argument, we would obtain $\beta(\chi_k) = \beta(\chi_i)$ whenever $i \leq k \leq j$, and both e_k and e_k' would both be principal boundaries whenever $i < k < j$. □

It follows that p is acyclic as soon as P° is. Indeed, if ξ is a cycle in p:

- either ξ contains an outer tree, and we can apply Lemma 3.7.7 to obtain a cycle in P°;
- or ξ is an inner path, and we proceed inductively in $\Theta(\beta(\xi))$.

Our next result is a quantitative version of this property: not only there is no cycle in p but the length of paths in p is bounded by a function of P° (whereas the size of p is obviously not bounded in general).

Theorem 3.7.8. *If $p \in \mathcal{T}(P^\circ)$ and $\xi \in \mathbf{P}(p)$ then $\mathbf{ln}(\xi) \leq 2^{\mathbf{depth}(P^\circ)}\mathbf{size}(P^\circ)$.*

Proof. The proof is by induction on $\mathbf{depth}(P^\circ)$.

First assume that $\xi = \xi_0(e_1)\chi_1(e'_1)\xi_1 \cdots (e_n)\chi_n(e'_n)\xi_n$ where each ξ_i is an outer path, and each $(e_i)\chi_i(e'_i)$ is a box path. Write $(b_i, j_i) = \iota(\chi_i)$: by applying the induction hypothesis to $\chi_i \in \mathbf{P}(p^{b_i}_{j_i})$, we obtain $\mathbf{ln}(\chi_i) \leq 2^{\mathbf{depth}(\Theta(b_i))}\mathbf{size}(\Theta(b_i))$. Moreover observe that $2n + \sum_{i=0}^{n} \mathbf{ln}(\xi^*_i) \leq \mathbf{ln}(\xi^*) \leq \mathbf{size}_0(P^\circ)$. By Lemma 3.7.6, it follows that $2n + \sum_{i=0}^{n} \mathbf{ln}(\xi_i) \leq \mathbf{size}_0(P^\circ)$. We obtain:

$$\mathbf{ln}(\xi) = 2n + \sum_{i=0}^{n} \mathbf{ln}(\xi_i) + \sum_{i=1}^{n} \mathbf{ln}(\chi_i) \leq \mathbf{size}_0(P^\circ) + \sum_{i=1}^{n} 2^{\mathbf{depth}(\Theta(b_i))}\mathbf{size}(\Theta(b_i)).$$

By Lemma 3.7.7, each $b \in B_\Theta$ occurs at most twice in the sequence $(b_1, \dots, b_n)$, hence we obtain:

$$\mathbf{ln}(\xi) \leq \mathbf{size}_0(P^\circ) + 2 \sum_{b \in B_\Theta} 2^{\mathbf{depth}(\Theta(b))}\mathbf{size}(\Theta(b)).$$

hence

$$\mathbf{ln}(\xi) \leq 2^{\mathbf{depth}(P^\circ)}\big(\mathbf{size}_0(P^\circ) + \sum_{b \in B_\Theta} \mathbf{size}(\Theta(b))\big).$$

since $\mathbf{depth}(\Theta(b)) < \mathbf{depth}(P^\circ)$ for each $b \in B_\Theta$. We conclude recalling that $\mathbf{size}(P^\circ) = \mathbf{size}_0(P^\circ) + \sum_{b \in B_\Theta} \mathbf{size}(\Theta(b))$.

The other possible cases are those of paths $\chi_0(e'_0)\xi$, $\xi(e_{n+1})\chi_n$ or $\chi_0(e'_0)\xi(e_{n+1})\chi_n$ where ξ is as above e'_0 and e_{n+1} are boundaries and χ_0 and χ_n are inner paths. Reasonning as in the proof of Lemma 3.7.7, we also obtain that each $b \in B_\Theta$ occurs at most twice in the sequence, e.g., $(b_0, \dots, b_{n+1})$, and then the proof follows similarly. □

In particular, we obtain $\mathbf{ln}(p) \leq 2^{\mathbf{depth}(P^\circ)}\mathbf{size}(P^\circ)$, In the following lemma, we show that our measure on jumps in the Taylor expansion of P° is also entirely determined by P°.

Lemma 3.7.9. *If $p \in \mathcal{T}(P^\circ)$ then $\mathbf{jd}(p) \leq \mathbf{size}(P^\circ)$.*

Proof. We show that if $t \in \mathbf{T}(p)$ then $\mathbf{jd}(t) \leq \mathbf{size}(P^\circ)$. The proof is, again, by induction on $\mathbf{depth}(P^\circ)$. If t is inner with $\iota(t) = (b, j)$, then we conclude directly by applying the induction hypothesis to p^b_j and $\Theta(b)$: indeed in this case, $\jmath_p^{-1}(t) = \jmath_{p^b_j}^{-1}(t)$, and $\mathbf{size}(\Theta(b)) \leq \mathbf{size}(P^\circ)$.

So we can assume that t is outer. In this case, observe from Definition 3.7.3 that if $\jmath_p(\mu) = t$ then $\mu = T[\overrightarrow{r}]$ for some $T \in \mathbf{T}(P^\circ)$. It follows that $\#\jmath_p^{-1}(t) \leq \#\mathbf{T}(P^\circ) \leq \mathbf{size}(P^\circ)$.

□

3.7.4 Cut elimination and Taylor expansion

In resource nets [ER06b], the elimination of the cut

$$\langle ?(t_1,\ldots,t_n)|!(s_1,\ldots,s_m)\rangle$$

yields the finite sum

$$\sum_{\sigma:\{1,\ldots,n\}\xrightarrow{\sim}\{1,\ldots,m\}} \langle t_1|s_{\sigma(1)}\rangle,\ldots,\langle t_n|s_{\sigma(n)}\rangle.$$

It turns out that the results of Sections 3.3 to 3.6 apply directly to resource nets: setting

$$\langle ?(t_1,\ldots,t_n)|!(s_1,\ldots,s_n)\rangle \to \langle t_1|s_{\sigma(1)}\rangle,\ldots,\langle t_n|s_{\sigma(n)}\rangle$$

for each permutation σ, we obtain an instance of multiplicative reduction, as the order of premises is irrelevant from a combinatorial point of view — this is all the more obvious because no typing constraint was involved in our argument. In other words, Corollary 3.6.5 also applies to the parallel reduction of resource nets. With Theorem 3.7.8 and Lemma 3.7.9 we obtain:

Corollary 3.7.10. *If q is a resource net and P is an MELL net and $k \in \mathbf{N}$, $\{p \in \mathcal{T}(P) \mid p \rightrightarrows^k q\}$ is finite.*

As for Corollaries 3.6.2 and 3.6.5, this holds only up to α-equivalence. And, again, it should be read keeping in mind that jumps are only an additional control structure on top of the underlying net. Indeed, if P is a bare MELL net (*i.e.* an MELL net without a jump function) then we can define $\mathcal{T}(P)$ as a set of bare resource nets. Then, given $k \in \mathbf{N}$, a bare resource net q, a bare MELL net P, and a jump function $\jmath$ such that $(P,\jmath)$ acyclic, there are finitely many bare resource nets $p \in \mathcal{T}(P)$ such that $p \rightrightarrows^k q$: it suffices to construct $\jmath_p$ from $\jmath$.

Beware that Corollary 3.7.10 depends on the acyclicity of the original MELL net. The following example shows how a cyclic net can induce infinite sets of antireducts.

Example 3.7.11. *Let $P = (\Theta; \langle ?(a_1^b)|a_0^b\rangle; ; \jmath)$ with $\Theta(b) = (\Theta'; ; x, \overline{x}; ; \jmath)$ where the domain of Θ', $\jmath$ and $\jmath'$ is empty. Then, by definition,*

$$\mathcal{T}(P) = \{p = (\langle\lambda|\mu\rangle;)\} \cup \{p_n = (\langle ?(x_0,\ldots,x_n)|!(\overline{x}_0,\ldots,\overline{x}_n)\rangle;) \mid n \in \mathbf{N}\}$$

where $\jmath_p(\lambda) = \mu$. Then, for each $n \in \mathbf{N}$, we have

$$p_n \to (\langle x_0|\overline{x}_{\sigma(0)}\rangle,\ldots,\langle x_n|\overline{x}_{\sigma(n)}\rangle;)$$

for each permutation σ of $\{0,\ldots,n\}$. In particular, if we set $\sigma(i) = i+1 \mod (n+1)$, then we obtain $p_n \to q_n = (\langle x_0|\overline{x}_1\rangle,\ldots,\langle x_n|\overline{x}_0\rangle;) \rightrightarrows_a (\langle x_0|\overline{x}_0\rangle;)$, and it follows that $\{q \in \mathcal{T}(P) \mid q \rightrightarrows^2 (\langle x_0|\overline{x}_0\rangle;)\}$ is infinite. This situation is illustrated in Figure 3.18.

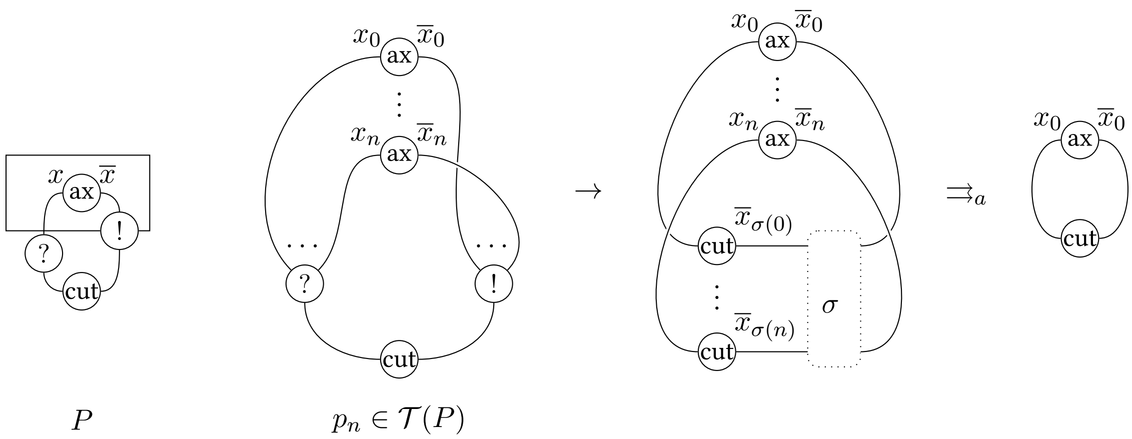

Figure 3.18 – Resource nets p_n of $\mathcal{T}(P)$ reducing to a single net

3.8 Conclusion

Recall that our original motivation was the definition of a reduction relation on infinite linear combinations of resource nets, simulating cut elimination in MELL through Taylor expansion. We claim that a suitable notion is as follows:

Definition 3.8.1. Write $\sum_{i \in I} a_i p_i \Rightarrow \sum_{i \in I} a_i q_i$ as soon as:

- each p_i is a resource net and each q_i is a finite sum of resource nets such that $p_i \rightrightarrows q_i$;
- for any resource net p, $\{i \in I \mid p_i = p\}$ is finite;
- for any resource net q, $\{i \in I \mid q \text{ is a summand of } q_i\}$ is finite.

In particular, if $\sum_{i \in I} a_i p_i$ is a Taylor expansion, then Corollary 3.7.10 ensures that the last condition of the definition of $\Rightarrow$ is automatically valid. The details of the simulation in a quantitative setting remain to be worked out, but the main stumbling block is now over: the necessary equations on coefficients are well established, as they have been extensively studied in the various denotational models; it only remained to be able to form the associated sums directly in the syntax.

Another incentive to publish our results is the *normalization-by-evaluation* programme that we develop with Guerrieri, Pellissier and Tortora de Falco [17]. This approach is restricted to connected MELL proof nets, *i.e.* MELL proof nets without weakening, and whose switching graphs are not only acyclic but also connected: [13]

- in this setting, a net P is entirely determined by the point of order 2 of its Taylor expansion, *i.e.* the unique resource net $p \in \mathcal{T}(P)$ with binary cocontractions only [GPT16];
- moreover, given two cut-free η-expanded nets Q and R, both the size of the normal form of a cut between Q and R and the number of cut elimination steps necessary to reach it can be bounded by a function of the relational semantics of Q and R [CPT11];

13. These are sufficiently expressive to simulate the λI-calculus, which is Turing-complete.

— from this data, we obtain a bound on the size of the point p_0 of order 2 of the normal form of the cut, as well as a bound on the number of parallel cut elimination steps necessary to obtain p_0 from its antecedent p in the Taylor expansion of the cut.

Our results in the present paper then provide a bound on the size of p: to find p it is then sufficient to compute the relational semantics of all the elements of the Taylor expansion of the cut whose size does not exceed this bound, and to check which one gives a semantics of order 2; then we can compute p_0 as the normal form of p, and this is sufficient to determine the normal form of the cut.

The restriction to connected nets is necessary to apply the injectivity result of Guerrieri, Pellissier and Tortora de Falco [GPT16], based on a fixed order of Taylor expansion. The injectivity of Taylor expansion and thus of the relational semantics of full MELL has been proved by de Carvalho [Car18b]: to determine P from $\mathcal{T}(P)$, this result relies on a k-heterogeneous expansion of P, *i.e.* an expansion of P for which the number of copies of each box is a power of k, and those degrees of expansion are chosen pairwise distinct. For the result to apply, the value of the parameter k must be sufficiently large: such a k may be computed from the linear expansion of P, obtained by taking exactly one copy of each box; but the degrees of expansion of boxes cannot be bounded in advance, and it is thus not clear if the above normalization-by-evaluation procedure could be adapted in this setting.

Let us conclude with a remark about a possible adaptation of our results to a (maybe) more standard representation of nets, including separate derelictions and coderelictions, with a finer grained cut elimination procedure. This introduces additional complexity in the formalism but it essentially requires no new concept or technique: the difficulty in parallel reduction is to control the chains of cuts to be simultaneously eliminated, and decomposing cut elimination into finer reduction steps can only decrease the length of such chains. On the other hand, in that setting, it is well known that cut elimination alone is not enough to capture the β-reduction of λ-calculus, and it must be extended with additional rewriting rules accounting for structural identities (e.g., associativity and commutativity of contraction). The details of the Taylor expansion analysis of cut elimination up to these identities are worked out in the PhD thesis of the first author [25, Chapter 2], [14] including the treatment of coefficients as mentioned above.

14. In French.

Chapter 4

On the Taylor expansion of λ-terms and the groupoid structure of their rigid approximants

This chapter is essentially the inclusion of the article of the same name [7], co-authored with Federico Olimpieri, and accepted for publication in Logical Methods in Computer Science. *It is a follow-up on preliminary work presented at the TLLA workshop in 2018 [18].*

Abstract: We show that the normal form of the Taylor expansion of a λ-term is isomorphic to its Böhm tree, improving Ehrhard and Regnier's original proof along three independent directions.

First, we simplify the final step of the proof by following the left reduction strategy directly in the resource calculus, avoiding to introduce an abstract machine *ad hoc*.

We also introduce a groupoid of permutations of copies of arguments in a rigid variant of the resource calculus, and relate the coefficients of Taylor expansion with this structure, while Ehrhard and Regnier worked with groups of permutations of occurrences of variables.

Finally, we extend all the results to a nondeterministic setting: by contrast with previous attempts, we show that the uniformity property that was crucial in Ehrhard and Regnier's approach can be preserved in this setting.

4.1 Introduction

4.1.1 Quantitative semantics

The field of quantitative semantics, in the sense originally introduced by Girard [Gir88], is currently very lively within the linear logic community and beyond. The basic idea is to interpret λ-terms as generalized power series, associated with analytic maps — instead of continuous maps, *à la* Scott. The concept predates linear logic, and in fact it provided the foundations for it, *via* its simpler, qualitative counterpart: coherence spaces [Gir87]. It was later revisited, e.g. by

Lamarche [Lam92] and Hasegawa [Has02], to provide a denotational interpretation of linear logic proofs as matrices; but the current momentum originates in the more recent introduction by Ehrhard [Ehr05] of models of linear logic, based on a particular class of topological vector spaces, and thus accommodating differentiation.

In that setting, the analytic maps associated with λ-terms are also smooth maps, *i.e.* they are infinitely differentiable. This led to the differential extensions of λ-calculus [ER03] and linear logic [ER06b] by Ehrhard and Regnier. The keystone of this line of work is an analogue of the Taylor expansion formula, which allows to translate terms (or proofs) into infinite linear combinations of finite approximants [ER08]: in the case of λ-calculus, those approximants are the terms of a resource calculus, in which the copies of arguments of a function must be provided explicitly, and then consumed linearly, instead of duplicated or discarded during reduction.

This renewed approach to quantitative semantics served as the basis of a considerable amount of recent work: either as a framework for denotational models accommodating linear combinations of maps [Lai+13; Lai16; TAO17; Ong17, *etc.*], possibly in contexts where sums are constrained to a particular form, such as the probabilistic setting [DE11; TAO18, *etc.*]; or as a tool for characterizing computational properties of programs *via* those of their approximants [MP11; 12; LL19; BM20, *etc.*].

Indeed, by contrast with denotational semantics, resource approximants retain a dynamics, albeit very simple and finitary: the size of terms is strictly decreasing under reduction. The seminal result relating the reduction of λ-terms with that of their approximants is the commutation between Taylor expansion and normalization: Ehrhard and Regnier have shown that the Taylor expansion M^* of a λ-term M can always be normalized, and that its normal form is nothing but the Taylor expansion of the Böhm tree $BT(M)$ of M [ER08; ER06a]. In particular, the normal form of Taylor expansion defines a proper denotational semantics.

4.1.2 Contributions

Ehrhard and Regnier's proof of the identity $BT(M)^* = NF(M^*)$ can be summed up as follows:

Step 1: The non-zero coefficients of resource terms in M^* do not depend on M. More precisely, we can write $M^* = \sum_{s\in T(M)} \frac{1}{m(s)} s$, where $T(M)$ is the support set of Taylor expansion and $m(s)$ is an integer coefficient depending only on the resource term s.

Step 2: The set $T(M)$ is a clique for the coherence relation obtained by setting $s \coh s'$ iff s and s' differ only by the multiplicity of arguments in applications.

Step 3: The respective supports of $NF(s)$ and $NF(s')$ are disjoint whenever $s \coh s'$ and $s \neq s'$. Then one can set $NF(M^*) = \sum_{s\in T(M)} \frac{1}{m(s)} NF(s)$, the summands being pairwise disjoint.

Step 4: If s is uniform, *i.e.* $s \coh s$, and t is in the support of $NF(s)$ (the normal form of s, which is a finite sum of resource terms) then $m(t)$ divides $m(s)$ and the coefficient of t in $NF(s)$ is $\frac{m(s)}{m(t)}$.

Step 5: By Step 1, $BT(M)^* = \sum_{t\in T(BT(M))} \frac{1}{m(t)} t$. To deduce the identity $BT(M)^* = NF(M^*)$ from the previous results, it is then sufficient to prove that $t \in T(BT(M))$ iff there exists $s \in T(M)$ such that t is in the support of $NF(s)$.

The first two steps are easy consequences of the definitions. For Step 3, it is sufficient to follow a well chosen normalization strategy, and check that it preserves coherence and that if two coherent terms share a reduct then they are equal [ER08, Section 3]. Step 4 relies on a careful investigation of the combinatorics of substitution in the resource calculus: this involves an elaborate argument about the structure of particular subgroups of the group of permutations of variable occurrences [ER08, Section 4]. Finally, Ehrhard and Regnier establish Step 5 by relating Taylor expansion with execution in an abstract machine [ER06a].

In the present work, we propose to revisit this seminal result, along three directions.

(i) We largely simplify Step 5, relying on a technique introduced by the second author [13]. We consider the hereditary head reduction strategy (a slight variant of leftmost reduction, underlying the construction of Böhm trees) and show that it can be simulated directly in the resource calculus, through Taylor expansion. We thus avoid the intricacies of an abstract machine with resource state.

(ii) We extend all the results to a model of nondeterminism, introduced as a formal binary choice operator in the calculus. By contrast with previous proposals to nondeterminism from Ehrhard [Ehr10], or Pagani, Tasson and Vaux Auclair [12; 13], we show that uniformity can still be relied upon, provided one keeps track of choices in the resource calculus: the coherence associated with nondeterministic choice is then that of the *with* connective ($\&$) of linear logic.

(iii) We analyse coefficients in the Taylor expansion by introducing a groupoid whose objects are rigid resource terms, *i.e.* resource terms in which multisets of arguments are replaced with lists, and whose isomorphisms are permutation terms, *i.e.* terms equipped with permutations that act on lists of arguments. This is more in accordance with the intuition that $m(s)$ is the number of permutations of arguments that leave s (or rather, any rigid representation of s) invariant: Ehrhard and Regnier rather worked on permutations of variable occurrences, which allowed them to consider groups rather than a groupoid.

Although we implement all three contributions together, they are essentially independent of each other. Indeed, the simplification of Step 5 brought by our contribution (i) only concerns the compatibility of Taylor expansion with normalization at the level of support sets, which does not involve coefficients; and it does not rely on uniformity, so its extension to nondeterministic superpositions is straightforward.

Moreover, while our contribution (ii) enables us to enforce the uniformity condition of Steps 2 to 4 in presence of a choice operator, it also ensures that distinct branches of a choice have disjoint supports in the Taylor expansion. This treatment of nondeterminism makes it completely transparent in the computation of coefficients. In particular, one could straightforwardly extend all steps of Ehrhard and Regnier's proof in this setting, *ceterit paribus*.

Our contribution (iii) is thus not needed for that endeavour: it only offers an alternative viewpoint on the combinatorics of substitution and normalization in the resource calculus, in a uniform setting. Nonetheless, we consider it to be the main contribution of the paper, precisely because of the new light it sheds on this dynamics, which in turn reveals possible connections with other approaches.

4.1.3 Scope and related works

Our contribution (i) establishes that, although it is interesting in itself, Ehrhard and Regnier's study of the relationship between elements in the Taylor expansion of a term and its execution in an abstract machine is essentially superfluous for proving the commutation theorem.

Barbarossa and Manzonetto have independently proposed another technique which amounts to show that any reduction from an element of $T(M)$ can be completed into a sequence of reductions simulating a β-reduction step [BM20, Section 4.1]. The strength of our own proposal is that, rather than a mere simulation result, we establish a commutation on the nose: hereditary head reduction commutes with Taylor expansion, at the level of supports. Moreover, the Böhm tree of a λ-term is the limit of its hereditary head reducts, which ensures that this commutation extends to normalization (Step 5). The same path was followed by Dal Lago and Leventis [LL19] for the probabilistic case. Let us mention that the commutation with hereditary head reduction actually holds not only at the level of supports, but also taking coefficients into account [13], in the more general setting of the algebraic λ-calculus [2] and without any additional condition: then, whenever the convergence of the sum defining the normal form of Taylor expansion is assured, the main commutation theorem ensues directly. This offers an alternative to the method of Ehrhard and Regnier that is the focus of the present paper.

As stated before, our proposal (ii) to restore uniformity in a nondeterministic setting is valid only because the resource calculus keeps a syntactic track of choices. The corresponding constructors are exactly those used by Tsukada, Asada and Ong [TAO17] who were interested in identifying equivalent execution paths of nondeterministic programs, but those authors do not mention, nor rely upon any coherence property: this forbids Steps 1 to 4 and, instead, they depend on infinite sums of arbitrary coefficients to be well defined. By contrast, Dal Lago and Leventis have independently proposed nearly the same solution as ours [LL19, Section 2.2], with only a minor technical difference in the case of sums.

The previous two proposals (i) and (ii) may be considered as purely technical improvements of the state of the art in the study of Taylor expansion. What we deem to be the most meaningful contribution of the present paper is our study of the groupoid of rigid resource terms. This provides us with a new understanding of the coefficients in the Taylor expansion of a term, in which we can recast the proof of the commutation theorem, especially Step 4: apart from this change of focus, the general architecture of our approach does not depart much from that of Ehrhard and Regnier, but we believe the obtained combinatorial results are closer to the original intuition behind the definition of m. In fact, a notable intermediate result (Lemma 4.5.11, p.140) is that the function that maps each permutation term to the permutation it induces on the occurrences of a fixed variable is functorial: one might understand Ehrhard and Regnier's proof of Step 4 as the image of ours through that functor. Moreover, our study suggests interesting connections with otherwise independent approaches to denotational semantics based on generalized species of structures [Fio+08; TAO17] and rigid intersection type systems [MPV18].

It is indeed most natural to compare our proposals to the line of work of Tsukada, Asada and Ong [TAO17; TAO18]. On the one hand, Tsukada *et al.* thrive to develop an abstract understanding of reduction paths in a nondeterministic λ-calculus. They are led to consider a polyadic calculus *à la* Mazza [Maz12; MPV18] with syntactic markers for nondeterministic choice, moreover obeying linearity, typing and η-expansion constraints: in particular, in that polyadic setting, λ-abstractions bind lists of variables, each bound variable occurring exactly once. Then, to each simple type, they associate a groupoid of intersection types: an isomorphism in this groupoid acts on polyadic rigid terms by permuting variables bound in abstractions and lists of arguments in applications, in such a way that terms in its source intersection type yield terms in its target. They show that the obtained collection of groupoids form a bicategorical model of the simply typed $\lambda\mathbf{Y}$-calculus, the interpretation being given by a polyadic rigid variant of Taylor expansion. This interpretation is moreover isomorphic to the one obtained in generalized species of structures [Fio+08].

On the other hand, our contribution (ii) shows that Ehrhard and Regnier's technique can already be adapted to the same kind of nondeterminism as the one considered by Tsukada *et al.*, without introducing any new concept. Also, besides having markers for nondeterministic choice, the only difference between our rigid terms and the ordinary resource terms is that arguments are linearly ordered: we do not consider a polyadic version. In fact, the same rigid terms were already used by Tsukada *et al.* as *intermediate representations* of resource terms, in order to recover Ehrhard and Regnier's commutation theorem as a by-product of their construction [TAO17, Section VI]. Moreover, our permutation terms are similar to their typed isomorphisms and this suggests directions for further investigations.

A natural follow-up to the present work would thus be to explore possible variations on our groupoid of permutation terms, and in particular adapt it to a polyadic setting, also taking free variables into account. We expect this study to yield a bicategorical model of the pure, untyped λ-calculus, similarly induced by rigid Taylor expansion *à la* Tsukada–Asada–Ong. Then potential connexions between the obtained model and the construction of various reflexive objects in the bicategory of generalized species of structures [Fio+08, Section 6.2] should be investigated.

Another possible route to the untyped setting, actively developed by the first author, is to construct a category satisfying a domain-like equation in the model of generalized species [Oli21]. The objects in this category are very much like intersection types, except that the usual identities between types (commutativity and, possibly, idempotency) are made explicit as morphisms, which allows to develop a bicategorical treatment of intersection type systems.

4.1.4 Structure of the paper

In the very brief Section 4.2, we review some results from group theory that will be useful later.

In Section 4.3 we extend the ordinary untyped λ-calculus with a generic nondeterministic choice operator, and present its operational semantics, inspired from that of the algebraic λ-calculus, as well as the corresponding notion of (non extensional) Böhm trees.

Section 4.4 recalls and adapts the definitions of the resource calculus and Taylor expansion. We obtain Step 2 as a straightforward consequence of the definitions and Step 5 by showing that the support of Taylor expansion is compatible with hereditary head reduction — this is our contribution (i). We moreover complete Step 1, making prominent the rôle played by permutations acting on lists of resource terms.

Section 4.5 is the core of the paper, developing our main contribution (iii): we introduce the rigid version of resource terms, and the isomorphisms between them, given by permutation terms; then we explore the relationship between the groupoid thus formed and the combinatorics of Taylor expansion. We first show that the coefficient $m(s)$ is nothing but the cardinality of the group of automorphisms of any rigid version of s. Then we study the structure of permutation terms between substitutions, first in the general case, then in the uniform case — which is allowed in our nondeterministic setting thanks to our contribution (ii). We leverage the obtained results to determine the coefficient of any resource term in the symmetric multilinear substitution associated with a reduction step issued from a uniform redex.

The final Section 4.6 builds on the study of rigid resource terms and permutation terms to achieve Steps 3 and 4. We conclude the paper with the commutation theorem.

4.2 Some basic facts on groups and group actions

Let $\mathcal{G}$ be a group, X be a set, and write $(g, a) \in \mathcal{G} \times X \mapsto [g]a \in X$ for a left action of $\mathcal{G}$ on X. If $a \in X$, then the *stabilizer* of a under this action is $St(a) := \{g \in \mathcal{G} \mid [g]a = a\}$, which is a subgroup of $\mathcal{G}$ (also called the isotropy group of a); and the *orbit* of a is the set $[\mathcal{G}]a := \{[g]a \mid g \in \mathcal{G}\} \subseteq X$. If $\mathcal{H}, \mathcal{K} \subseteq \mathcal{G}$, we write $\mathcal{H}\mathcal{K} := \{hk \mid h \in \mathcal{H},\ k \in \mathcal{K}\}$. If $f : X \to Y$, $X' \subseteq X$ and $Y' \subseteq Y$ we write $f(X') := \{f(x) \mid x \in X'\}$ and $f^{-1}(Y') := \{x \mid f(x) \in Y'\}$.

Assuming that $\mathcal{G}$ is finite, the following three facts are standard results of group theory.

Fact 4.2.1. For any $a \in X$,

$$Card([\mathcal{G}]a) = \frac{Card(\mathcal{G})}{Card(St(a))} \quad .$$

Proof. [Lan02, Proposition 5.1]. □

Fact 4.2.2. Let $\mathcal{H}$ and $\mathcal{K}$ be any subgroups of $\mathcal{G}$. Then

$$Card(\mathcal{H}\mathcal{K}) = \frac{Card(\mathcal{H})\,Card(\mathcal{K})}{Card(\mathcal{H} \cap \mathcal{K})} \quad .$$

Proof. [Suz82, §(3.11)]. □

Fact 4.2.3. Let $f : \mathcal{G} \to \mathcal{H}$ be a group homomorphism and $\mathcal{K}$ be a subgroup of $\mathcal{H}$. Then

$$\frac{Card(\mathcal{G})}{Card(f^{-1}(\mathcal{K}))} = \frac{Card(f(\mathcal{G}))}{Card(f(\mathcal{G}) \cap \mathcal{K})} \quad .$$

Proof. By the theorem of correspondence under homomorphisms [Suz82, Theorem 5.5 (1)], observing that $f(\mathcal{G}) \cap \mathcal{K} = f(f^{-1}(\mathcal{K}))$. □

4.3 A generic nondeterministic λ calculus

4.3.1 $\lambda_\oplus$-terms

We consider a nondeterministic version of the λ-calculus in a pure, untyped setting. The terms are those of the pure λ-calculus, augmented with a binary operator $\oplus$ denoting a form of nondeterministic superposition: [1]

$$\Lambda_\oplus \ni M, N, P, Q ::= x \mid \lambda x.M \mid MN \mid M \oplus N.$$

As usual $\lambda_\oplus$-terms are considered up to renaming bound variables, and we write $M[N/x]$ for the capture avoiding substitution of N for x in M. We give precedence to application over abstraction, and to abstraction over $\oplus$, and moreover associate applications on the left, so that we may write $\lambda x.MNP \oplus Q$ for $(\lambda x.((MN)P)) \oplus Q$. We write $\lambda \vec{x}.M$ for a term of the form $\lambda x_1. \cdots \lambda x_n.M$ (possibly with $n = 0$).

Rather than specifying the computational effect of $\oplus$ explicitly, by reducing $M \oplus M'$ to either M or M', we consider two reductions rules

$$(M \oplus N)P \to MP \oplus NP \qquad \text{and} \qquad \lambda x.(M \oplus N) \to \lambda x.M \oplus \lambda x.N$$

in addition to the β-reduction rule. This is in accordance with most of the literature associated with the Taylor expansion of λ-terms [ER03; Ehr10; 12; 13] and quantitative denotational semantics [Ehr05], where nondeterministic choice is modelled by the sum of denotations: rather than the current state of a nondeterministic computation, a term represents a superposition of possible results. [2] In particular, this approach allows to keep standard rewriting notions and techniques such as confluence, standardization, etc. Formally, $\to$ is defined inductively by the inference rules of Figure 4.1: we simply extend the three base cases contextually.

Observe that neither the definition of terms nor that of reduction make the choice operator commutative, associative nor idempotent: e.g., $x \oplus y$ and $y \oplus x$ are two distinct normal forms. It is possible to extend the reduction relation to validate the structural properties associated with various kinds of superpositions (plain nondeterministic choice, probabilistic choice or a more general quantitative superposition) while retaining good rewriting properties: we refer the reader to the work of Leventis [Lev19] for an extensive study of this approach.

By contrast, for our purposes, it is essential to keep $\oplus$ as a free binary operator: following Tsukada, Asada and Ong [TAO17], we keep track of the branching structure of choices along the reduction. This information will be reflected in the Taylor expansion to be introduced in Section 4.4: this is the key to recover the uniformity property while allowing for nondeterministic superpositions of terms.

1. Throughout the paper, we use a self explanatory if not standard variant of BNF notation for introducing syntactic objects: here we define the set $\Lambda_\oplus$ as that inductively generated by variables, λ-abstraction, application and sum, and we will denote terms using letters among M, N, P, Q, possibly with sub- and superscripts.

2. In fact, only the rule $(M \oplus N)P \to MP \oplus NP$ is really necessary in order to enable the potential redexes that can occur if M or N is an abstraction: in the setting of quantitative semantics, term application is left-linear. The other reduction rule can be derived in case one admits extensionality in the models or the η-rule in the calculus (here we don't, though): having it in the calculus means that we follow a *call-by-name* interpretation of nondeterministic evaluation, which amounts to λ-abstraction being linear [Ass+14]. The results of the paper could be developed similarly without it. We chose to keep it nonetheless, because it simplifies the underlying theory of Böhm trees and allows us to obtain Ehrhard and Regnier's results [ER08; ER06a] as a particular case of our own.

$$\frac{}{(\lambda x.M)N \to M[N/x]} \qquad \frac{}{(M \oplus N)P \to MP \oplus NP} \qquad \frac{}{\lambda x.(M \oplus N) \to \lambda x.M \oplus \lambda x.N}$$

$$\frac{M \to M'}{\lambda x.M \to \lambda x.M'} \quad \frac{M \to M'}{MN \to M'N} \quad \frac{M \to M'}{NM \to NM'} \quad \frac{M \to M'}{M \oplus N \to M' \oplus N} \quad \frac{M \to M'}{N \oplus M \to N \oplus M'}$$

Figure 4.1 – Reduction rules of the $\lambda_\oplus$-calculus

In fact we will not really consider the reduction relation $\to$ in the present paper, and rather focus on the *hereditary head reduction strategy* obtained by defining the function $L : \Lambda_\oplus \to \Lambda_\oplus$ inductively as follows:

$$\begin{aligned}
L(M \oplus N) &:= L(M) \oplus L(N) \\
L(\lambda \vec{x}.\lambda y.(M \oplus N)) &:= \lambda \vec{x}.(\lambda y.M \oplus \lambda y.N) \\
L(\lambda \vec{x}.(M \oplus N)PQ_1 \cdots Q_k)) &:= \lambda \vec{x}.(MP \oplus NP)Q_1 \cdots Q_k \\
L(\lambda \vec{x}.yQ_1 \cdots Q_k) &:= \lambda \vec{x}.yL(Q_1) \cdots L(Q_k) \\
L(\lambda \vec{x}.(\lambda y.M)NQ_1 \cdots Q_k)) &:= \lambda \vec{x}.M[N/y]Q_1 \cdots Q_k \quad .
\end{aligned}$$

Observe that this definition is exhaustive because any term in $\Lambda_\oplus$ is either of the form $M \oplus N$ or of the form $\lambda \vec{x}.\lambda y.(M \oplus N)$ or of the form $\lambda \vec{x}.RQ_1 \cdots Q_k$ with $R = (\lambda y.M)N$ or $R = (M \oplus N)P$ or $R = y$.

It should be clear that $M \to^* L(M)$ and that $L(M) = M$ whenever M is normal [3] but the converse does not necessarily hold. It can moreover be shown that any normalizable term M reaches its normal form by repeatedly applying the function L, for instance by adapting the standardization techniques of Leventis [24; Lev19], but this is not the focus of the present paper. Indeed, we are only interested in the construction of Böhm trees, and we rely on the fact that the Böhm tree of a term M can be understood as the limit of the sequence $(L^n(M))_{n\in\mathbb{N}}$, in a sense that we detail below. In particular, M and $L(M)$ have the same Böhm tree (Lemma 4.3.3).

4.3.2 Böhm trees

We first define the set $\Lambda_\oplus^\perp$ of *term approximants* as follows:

$$\Lambda_\oplus^\perp \ni M, N, P, Q ::= \perp \mid x \mid \lambda x.M \mid MN \mid M \oplus N$$

then we consider the least partial order $\leq \subseteq \Lambda_\oplus^\perp \times \Lambda_\oplus^\perp$ that is compatible with syntactic constructs and such that $\perp \leq M$ for each $M \in \Lambda_\oplus^\perp$. Formally, $\leq$ is defined inductively by the rules of Figure 4.2.

The set $\mathcal{N} \subset \Lambda_\oplus^\perp$ of *elementary Böhm trees* is the least set of approximants such that:

– $\perp \in \mathcal{N}$;

3. If one considers $\oplus$ as a nondeterministic choice operator, normalizability is meant in its *must* flavour here. Indeed, we do not perform the choice within the reduction relation itself, so $M \oplus N$ is normal iff M and N both are.

$$\frac{}{\bot \leq M} \qquad \frac{}{M \leq M} \qquad \frac{M \leq N \qquad N \leq P}{M \leq P}$$

$$\frac{M \leq M'}{\lambda x.M \leq \lambda x.M'} \qquad \frac{M \leq M' \qquad N \leq N'}{MN \leq M'N'} \qquad \frac{M \leq M' \qquad N \leq N'}{M \oplus N \leq M' \oplus N'}$$

Figure 4.2 – The approximation order on $\Lambda_{\oplus}^{\bot}$.

- $\lambda \vec{x}.xN_1 \cdots N_n \in \mathcal{N}$ as soon as $N_1, \ldots, N_n \in \mathcal{N}$;[4] and
- $N_1 \oplus N_2 \in \mathcal{N}$ as soon as $N_1, N_2 \in \mathcal{N}$.

For each $\lambda_{\oplus}$-term M, we construct an elementary Böhm tree $\mathcal{N}(M)$ as follows:

$$\begin{aligned} \mathcal{N}(M \oplus N) &\coloneqq \mathcal{N}(M) \oplus \mathcal{N}(N) \\ \mathcal{N}(\lambda \vec{x}.xQ_1 \cdots Q_k) &\coloneqq \lambda \vec{x}.x\mathcal{N}(Q_1) \cdots \mathcal{N}(Q_k) \\ \mathcal{N}(M) &\coloneqq \bot \qquad \text{in all other cases.} \end{aligned}$$

Lemma 4.3.1. *For any $M \in \Lambda_{\oplus}$, $\mathcal{N}(M) \leq \mathcal{N}(L(M))$.*

Proof. By induction on M. If $M = M_1 \oplus M_2$ then $\mathcal{N}(M) = \mathcal{N}(M_1) \oplus \mathcal{N}(M_2)$ and $L(M) = L(M_1) \oplus L(M_2)$, hence $\mathcal{N}(L(M)) = \mathcal{N}(L(M_1)) \oplus \mathcal{N}(L(M_2))$ and we conclude by induction hypothesis. The case $M = \lambda \vec{x}.xQ_1 \cdots Q_k$ is similar. Otherwise, $\mathcal{N}(M) = \bot \leq \mathcal{N}(L(M))$. □

Hence for a fixed $\lambda_{\oplus}$-term M, the sequence $(\mathcal{N}(L^n(M)))_{n \in \mathbb{N}}$ is increasing, and we call its downwards closure in $\mathcal{N}$ *the Böhm tree of* M, which we denote by $BT(M)$: *i.e.* we set $BT(M) \coloneqq \{N \in \mathcal{N} \mid \exists n \in \mathbb{N},\ N \leq \mathcal{N}(L^n(M))\}$.

Example 4.3.2. *Let $M = \Theta\lambda y.(y \oplus x)$ where Θ is Turing's fixpoint combinator, so that $L^3(M) = M \oplus x$. We can think of $BT(M)$ as the infinite tree $((\cdots \oplus x) \oplus x) \oplus x$: formally, $BT(M) = \{\bot \oplus nx \mid n \in \mathbb{N}\}$ where we define inductively $M \oplus 0N = M$ and $M \oplus (n+1)N = (M \oplus nN) \oplus N$.*

It could be shown that Böhm trees define a denotational semantics: if $M \to M'$ then $BT(M) = BT(M')$.[5] Here we just observe that Böhm trees are invariant under hereditary head reduction, which follows directly from the definition:

Lemma 4.3.3. *Let $M \in \Lambda_{\oplus}$. Then $BT(M) = BT(L(M))$.*

It will be sufficient to follow this strategy in order to establish Step 5, *i.e.* the qualitative version of the commutation between normalization and the Taylor expansion of $\lambda_{\oplus}$-terms, to be defined in the next section.

---

4. Here the sequence $\lambda \vec{x}$ of abstractions can be empty, and we can have $n = 0$, in which case the body of the term is just the head variable.

5. Again, this would require the adaptation of standardization techniques to $\lambda_{\oplus}$, similar to those developed by Leventis for the probabilistic λ-calculus [Lev19].

4.4 Taylor expansion in a uniform nondeterministic setting

In order to define Taylor expansion, we need to introduce an auxiliary language: the resource calculus.

4.4.1 Resource terms

We call *resource expressions* the elements of $\Delta_\oplus \cup \Delta_\oplus^!$, where the set $\Delta_\oplus$ of *resource terms* and the set $\Delta_\oplus^!$ of *resource monomials* are defined by mutual induction as follows: [6]

$$\Delta_\oplus \ni s, t, u, v ::= x \mid \lambda x.s \mid \langle s \rangle \bar{t} \mid s \oplus \bullet \mid \bullet \oplus s \qquad \Delta_\oplus^! \ni \bar{s}, \bar{t}, \bar{u}, \bar{v} ::= [s_1, \ldots, s_n]$$

and, in addition to α-equivalence, we consider resource expressions up to permutations of terms in monomials, so that $[s_1, \ldots, s_n]$ denotes a multiset of terms. We give precedence to application and abstraction over $- \oplus \bullet$ and $\bullet \oplus -$, and we write $\langle s \rangle \bar{t}_1 \cdots \bar{t}_n$ for $\langle \cdots \langle s \rangle \bar{t}_1 \cdots \rangle \bar{t}_n$, so that we may write $\lambda x.\langle s \rangle \bar{t}\,\bar{u} \oplus \bullet$ for $(\lambda x.(\langle \langle s \rangle \bar{t} \rangle \bar{u})) \oplus \bullet$. We write $\lambda \vec{x}.s$ for a term of the form $\lambda x_1. \cdots \lambda x_n.s$. We moreover write $\bar{s} \cdot \bar{t}$ for the multiset union of $\bar{s}$ and $\bar{t}$, and if $\bar{s} = [s_1, \ldots, s_n]$ then we write $|\bar{s}| := n$ for the size of $\bar{s}$; in particular $|\bar{s}| = 0$ iff $\bar{s}$ is the empty multiset $[]$, which is neutral for multiset union.

If X is a set, we write $\mathbb{N}[X]$ for the freely generated commutative monoid over X: formally, this is the same as the set of finite multisets of elements of X but we choose to consider its elements as finite linear combinations of elements of X with coefficients in $\mathbb{N}$. In the following, we write $\Delta_\oplus^{(!)}$ for either $\Delta_\oplus$ or $\Delta_\oplus^!$, so that $\mathbb{N}[\Delta_\oplus^{(!)}]$ is either $\mathbb{N}[\Delta_\oplus]$ or $\mathbb{N}[\Delta_\oplus^!]$: when we consider a sum E of resource expressions, we always require E to be a sum of terms or a sum of monomials, *i.e.* $E \in \mathbb{N}[\Delta_\oplus^{(!)}]$. Then we write $\mathit{supp}(E) \subseteq \Delta_\oplus^{(!)}$ for the *support set* of E, which is finite. We extend the syntactic constructs of the resource calculus to finite sums of resource expressions by linearity, so that:

- if $S = \sum_{i=1}^n s_i$ then $\lambda x.S = \sum_{i=1}^n \lambda x.s_i$, $\bullet \oplus S = \sum_{i=1}^n \bullet \oplus s_i$ and $S \oplus \bullet = \sum_{i=1}^n s_i \oplus \bullet$;
- if moreover $\bar{T} = \sum_{j=1}^m \bar{t}_j$ then $\langle S \rangle \bar{T} = \sum_{i=1}^n \sum_{j=1}^m \langle s_i \rangle \bar{t}_j$ and $[S] \cdot \bar{T} = \sum_{i=1}^n \sum_{j=1}^m [s_i] \cdot \bar{t}_j$.

For any resource expression $e \in \Delta_\oplus^{(!)}$, we write $n_x(e)$ for the number of occurrences of variable x in e. If moreover $\bar{u} = [u_1, \ldots, u_n] \in \Delta_\oplus^!$, we introduce the *symmetric n-linear substitution* $\partial_x e \cdot \bar{u} \in \mathbb{N}[\Delta_\oplus^{(!)}]$ of $\bar{u}$ for the variable x in e, which is informally defined as follows:

$$\partial_x e \cdot \bar{u} := \begin{cases} \sum\limits_{\sigma \in \mathfrak{S}_n} e[u_{\sigma(1)}/x_1, \ldots, u_{\sigma(n)}/x_n] & \text{if } n_x(e) = n \\ 0 & \text{otherwise} \end{cases}$$

where $x_1, \ldots, x_{n_x(e)}$ enumerate the occurrences of x in e. [7]

6. Recall that the cartesian product of vector spaces is given by the disjoint union of bases: this is the intuition behind the operators $- \oplus \bullet$ and $\bullet \oplus -$, which will serve in the Taylor expansion of the operator $\oplus$ of $\Lambda_\oplus$. Again, we leave the exact computational behavior of $\oplus$ unspecified, and we treat it generically as a pairing operator (without projections): in this we follow Tsukada *et al.* [TAO17].

7. Enumerating the occurrences of x in e only makes sense if we fix an ordering of each monomial in e: the rigid resource calculus to be introduced later in the paper will allow us to give a more formal account of this intuitive presentation. For now we stick to the alternative definition given in the next paragraph.

$$\frac{}{\langle \lambda x.s\rangle \bar{t} \to_\partial \partial_x s \cdot \bar{t}} \qquad \frac{}{\langle s \oplus \bullet\rangle \bar{t} \to_\partial \langle s\rangle \bar{t} \oplus \bullet} \qquad \frac{}{\langle \bullet \oplus s\rangle \bar{t} \to_\partial \bullet \oplus \langle s\rangle \bar{t}}$$

$$\frac{}{\lambda x.(s \oplus \bullet) \to_\partial \lambda x.s \oplus \bullet} \qquad \frac{}{\lambda x.(\bullet \oplus s) \to_\partial \bullet \oplus \lambda x.s}$$

$$\frac{s \to_\partial S'}{\lambda x.s \to_\partial \lambda x.S'} \qquad \frac{s \to_\partial S'}{\langle s\rangle \bar{t} \to_\partial \langle S'\rangle \bar{t}} \qquad \frac{\bar{s} \to_\partial \bar{S}'}{\langle t\rangle \bar{s} \to_\partial \langle t\rangle \bar{S}'}$$

$$\frac{s \to_\partial S'}{s \oplus \bullet \to_\partial S' \oplus \bullet} \qquad \frac{s \to_\partial S'}{\bullet \oplus s \to_\partial \bullet \oplus S'} \qquad \frac{s \to_\partial S'}{[s] \cdot \bar{t} \to_\partial [S'] \cdot \bar{t}}$$

Figure 4.3 – Reduction rules of the resource calculus with sums

Formally, $\partial_x e \cdot \bar{u}$ is defined by induction on e, setting:

$$\begin{aligned}
\partial_x y \cdot \bar{u} &:= \begin{cases} y & \text{if } y \neq x \text{ and } n = 0 \\ u_1 & \text{if } y = x \text{ and } n = 1 \\ 0 & \text{otherwise} \end{cases} \\
\partial_x \lambda y.s \cdot \bar{u} &:= \lambda y.(\partial_x s \cdot \bar{u}) \\
\partial_x (s \oplus \bullet) \cdot \bar{u} &:= \partial_x s \cdot \bar{u} \oplus \bullet \\
\partial_x (\bullet \oplus s) \cdot \bar{u} &:= \bullet \oplus \partial_x s \cdot \bar{u} \\
\partial_x \langle s\rangle \bar{t} \cdot \bar{u} &:= \sum_{(I_1, I_2) \in \mathcal{Q}_2(n)} \langle \partial_x s \cdot \bar{u}_{I_1}\rangle \partial_x \bar{t} \cdot \bar{u}_{I_2} \\
\partial_x [t_1, \ldots, t_k] \cdot \bar{u} &:= \sum_{(I_1, \ldots, I_k) \in \mathcal{Q}_k(n)} [\partial_x t_1 \cdot \bar{u}_{I_1}, \ldots, \partial_x t_n \cdot \bar{u}_{I_k}]
\end{aligned}$$

where $\mathcal{Q}_k(n)$ denotes the set of k-tuples $(I_1, \ldots, I_k)$ of (possibly empty) pairwise disjoint subsets of $\{1, \ldots, n\}$ such that $\bigcup_{j=1}^k I_j = \{1, \ldots, n\}$,[8] and we write $\bar{u}_{\{i_1,\ldots,i_j\}} := [u_{i_1}, \ldots, u_{i_j}]$ whenever $1 \leq i_1 < \ldots < i_j \leq n$. It is easy to check that $\partial_x e \cdot \bar{t} \neq 0$ iff $n_x(e) = |\bar{t}|$.

The reduction of the resource calculus is the relation from resource expressions to finite formal sums of resource expressions induced by the rules of Figure 4.3: the first rule is the counterpart of β-reduction in the resource calculus; the next four rules implement the commutation of $\oplus$ with abstraction and application to a monomial; the final six rules ensure the contextuality of the resulting relation. It is extended to a binary relation on $\mathbb{N}[\Delta_\oplus^{(!)}]$ by setting $e + F \to_\partial E' + F$ whenever $e \to_\partial E'$. As in the case of the original resource calculus [ER08], the reduction relation $\to_\partial$ is confluent and strongly normalizing. Confluence may be proved following the same technique as for the original resource calculus [13, Section 3.4]: we do not provide any detail, because we will soon focus on a reduction strategy, which is functional. For strong normalization, slightly more care is needed, because the size of expressions does not necessarily decrease under reduction:

8. Note that this data is equivalent to a function $\{1, \ldots, n\} \to \{1, \ldots, k\}$.

Lemma 4.4.1. *The reduction $\to_\partial \subseteq \mathbb{N}[\Delta_\oplus^{(!)}] \times \mathbb{N}[\Delta_\oplus^{(!)}]$ is strictly normalizing.*

Proof. If $e \in \Delta_\oplus^{(!)}$, we write $n_\lambda(e) \in \mathbb{N}$ (resp. $n_\oplus(e) \in \mathbb{N}$) for the number of abstractions (resp. of $\oplus$) occurring in e. Let $\#_\oplus(e)$ denote the multiset of natural numbers containing a value $n_\oplus(s)$ for each occurrence of a subterm $\lambda x.s$ or $\langle s\rangle\bar{t}$ in e. Formally:

$$\#_\oplus(x) \coloneqq [] \quad \#_\oplus(\lambda x.s) \coloneqq [n_\oplus(s)] \cdot \#_\oplus(s) \quad \#_\oplus(s \oplus \bullet) \coloneqq \#_\oplus(s) \quad \#_\oplus(\bullet \oplus s) \coloneqq \#_\oplus(s)$$
$$\#_\oplus(\langle s\rangle\bar{t}) \coloneqq [n_\oplus(s)] \cdot \#_\oplus(s) \cdot \#_\oplus(\bar{t}) \qquad \#_\oplus([s_1,\dots,s_n]) \coloneqq \#_\oplus(s_1) \cdot \dots \cdot \#_\oplus(s_n)$$

where we use the same notations for multisets as for monomials.

We first establish that, for all $e \in \Delta_\oplus^{(!)}$, $\bar{t} \in \Delta_\oplus^!$ and $e' \in \mathit{supp}(\partial_x e \cdot \bar{t})$, we have $n_\lambda(e') = n_\lambda(e) - 1$: the proof is by a straightforward induction on e.

Then, whenever $e \to_\partial E'$ and $e' \in \mathit{supp}(E')$, we have $n_\oplus(e') = n_\oplus(e)$, and:

1. either $n_\lambda(e') = n_\lambda(e) - 1$;
2. or $n_\lambda(e') = n_\lambda(e)$, and we can write $\#_\oplus(e) = \bar{n} \cdot [n+1]$ and $\#_\oplus(e') = \bar{n} \cdot [n]$.

The proof is by induction on the derivation of $e \to_\partial E'$: the β-redex case holds using the previous result on multilinear substitution to obtain (1); the other four base cases yield (2); and each other case follows straightforwardly by the induction hypothesis.

Now, for each $E = e_1 + \dots + e_n \in \mathbb{N}[\Delta_\oplus^{(!)}]$, we write $\#_\lambda(E) = [n_\lambda(e_1), \dots, n_\lambda(e_n)]$ and $\#_\oplus(E) = \#_\oplus(e_1) \cdot \dots \cdot \#_\oplus(e_n)$. By the previous result and the definition of $\to_\partial$ on sums of resource expressions: if $E \to_\partial E'$ then either $\#_\lambda(E) < \#_\lambda(E)$, or $\#_\lambda(E) = \#_\lambda(E)$ and $\#_\oplus(E) < \#_\oplus(E)$, considering the multiset order. We conclude since the latter is well-founded. $\square$

We write $NF(E)$ for the unique normal form of $E \in \mathbb{N}[\Delta_\oplus^{(!)}]$, which is a linear operator: $NF(\sum_{i=1}^k e_i) = \sum_{i=1}^k NF(e_i)$. As stated before, we do not focus on the reduction relation itself, and we rather consider the *hereditary head reduction* strategy obtained by defining the function $L : \Delta_\oplus^{(!)} \to \mathbb{N}[\Delta_\oplus^{(!)}]$ inductively as follows:

$$L(s \oplus \bullet) \coloneqq L(s) \oplus \bullet \qquad L(\bullet \oplus s) \coloneqq \bullet \oplus L(s)$$
$$L(\lambda\vec{x}.\lambda y.(s \oplus \bullet)) \coloneqq \lambda\vec{x}.(\lambda y.s \oplus \bullet) \qquad L(\lambda\vec{x}.\lambda y.(\bullet \oplus s)) \coloneqq \lambda\vec{x}.(\bullet \oplus \lambda y.s)$$
$$L(\lambda\vec{x}.\langle\langle s \oplus \bullet\rangle\bar{t}\rangle\bar{u}_1 \cdots \bar{u}_k) \coloneqq \lambda\vec{x}.\langle\langle s\rangle\bar{t} \oplus \bullet\rangle\bar{u}_1 \cdots \bar{u}_k$$
$$L(\lambda\vec{x}.\langle\langle \bullet \oplus s\rangle\bar{t}\rangle\bar{u}_1 \cdots \bar{u}_k) \coloneqq \lambda\vec{x}.\langle \bullet \oplus \langle s\rangle\bar{t}\rangle\bar{u}_1 \cdots \bar{u}_k$$
$$L(\lambda\vec{x}.\langle y\rangle\bar{s}_1 \cdots \bar{s}_k) \coloneqq \lambda\vec{x}.\langle y\rangle L(\bar{s}_1) \cdots L(\bar{s}_k)$$
$$L([s_1,\dots,s_k]) \coloneqq [L(s_1),\dots,L(s_k)]$$
$$L(\lambda\vec{x}.\langle\lambda y.s\rangle\bar{t}\,\bar{u}_1 \cdots \bar{u}_k) \coloneqq \lambda\vec{x}.\langle\partial_y s \cdot \bar{t}\rangle\bar{u}_1 \cdots \bar{u}_k$$

extended to sums of resource expressions by linearity, setting $L(\sum_{i=1}^k e_i) \coloneqq \sum_{i=1}^k L(e_i)$.

Again, it should be clear that $e \to_\partial^* L(e)$: if e contains a redex (*i.e.* the left-hand side of any of the first five rules of Figure 4.3) in head position, then $L(e)$ is obtained by firing this redex; otherwise each term in a monomial argument of the head variable is reduced, following the same strategy inductively. Moreover, $e = L(e)$ iff e is normal: we obtain an equivalence because $\to_\partial$ is strongly normalizing on sums of resource expressions and, if e is not normal, $L(e)$ is obtained by firing at least one redex in e.

Due to the definition of $\to_\partial$ on sums and the linearity of L, these properties extend directly: $E \to_\partial^* L(E)$ and $E = L(E)$ iff E is normal (*i.e.* it is a sum of normal expressions). It moreover follows that L is normalizing: for all $E \in \mathbb{N}[\Delta_\oplus^{(!)}]$, there is n such that $L^n(E) = NF(E)$.

4.4.2 Taylor expansion of $\lambda_\oplus$-terms

The Taylor expansion of a $\lambda_\oplus$-term will be an infinite linear combination of resource terms: to introduce it, we first need some preliminary notations and results.

If X is a set, we write $\mathbb{Q}^+\langle X\rangle$ for the set of possibly infinite linear combinations of elements of X with non negative rational coefficients (in fact we could use any commutative semifield): equivalently, $\mathbb{Q}^+\langle X\rangle$ is the set of functions from X to the set of non negative rational numbers. We write $A = \sum_{a\in X} A_a\, a \in \mathbb{Q}^+\langle X\rangle$ and then the *support set* of A is $supp(A) = \{a \in X \mid A_a \neq 0\}$.

All the syntactic constructs of resource expressions are extended to infinite linear combinations, componentwise:

– if $S \in \mathbb{Q}^+\langle \Delta_\oplus\rangle$ then

$$\lambda x.S \coloneqq \sum_{s\in\Delta_\oplus} S_s(\lambda x.s) \quad , \quad S\oplus\bullet \coloneqq \sum_{s\in\Delta_\oplus} S_s(s\oplus\bullet) \quad \text{and} \quad \bullet\oplus S \coloneqq \sum_{s\in\Delta_\oplus} S_s(\bullet\oplus s) \quad ;$$

– if moreover $\bar{T} \in \mathbb{Q}^+\langle \Delta_\oplus^!\rangle$ then

$$\langle S\rangle\bar{T} \coloneqq \sum_{s\in\Delta_\oplus}\sum_{\bar{t}\in\Delta_\oplus^!} S_s\bar{T}_{\bar{t}}(\langle s\rangle\bar{t}) \quad ;$$

– and if $S_1, \ldots, S_n \in \mathbb{Q}^+\langle\Delta_\oplus\rangle$ then

$$[S_1, \ldots, S_n] \coloneqq \sum_{(s_1,\ldots,s_n)\in\Delta_\oplus^n} \Big(\prod_{i=1}^n S_{i s_i}\Big)[s_1, \ldots, s_n] \quad .$$

Observe indeed that each of these infinite sums is finite in each component: e.g., for each $\bar{s} \in \Delta_\oplus^!$, there are finitely many tuples $(s_1, \ldots, s_n) \in \Delta_\oplus^n$ such that $\bar{s} = [s_1, \ldots, s_n]$.

Similarly we extend syntactic constructs to sets of resource expressions:

– if $S \subseteq \Delta_\oplus$ then

$$\lambda x.S \coloneqq \{\lambda x.s \mid s \in S\} \quad , \quad S\oplus\bullet \coloneqq \{s\oplus\bullet \mid s \in S\} \quad \text{and} \quad \bullet\oplus S \coloneqq \{\bullet\oplus s \mid s \in S\} \quad ;$$

– if moreover $\bar{T} \subseteq \Delta_\oplus^!$ then

$$\langle S\rangle\bar{T} \coloneqq \{\langle s\rangle\bar{t} \mid s \in S,\ \bar{t} \in \bar{T}\} \quad ;$$

– and if $S_1, \ldots, S_n \subseteq \Delta_\oplus$ then

$$[S_1, \ldots, S_n] \coloneqq \{[s_1, \ldots, s_n] \mid s_i \in S_i \text{ for } 1 \leq i \leq n\} \quad .$$

Considering subsets of $\Delta_{\oplus}^{(!)}$ as infinite linear combinations of resource expressions with boolean coefficients, this is just a variant of the previous construction (which can be carried out in any commutative semifield). Moreover, syntactic constructs commute with the support function: e.g., $\lambda x.supp(S) = supp(\lambda x.S)$.

Let $S \in \mathbb{Q}^+\langle\Delta_\oplus\rangle$. We define $S^n \in \mathbb{Q}^+\langle\Delta^!_\oplus\rangle$ by induction on n: $S^0 = []$ and $S^{n+1} = [S] \cdot S^n$. Then we define the *promotion* of S as the series $S^! = \sum_{n=0}^{\infty} \frac{1}{n!} S^n$: because the supports of S^n and S^p are disjoint when $n \neq p$, this sum is componentwise finite. If $S \subseteq \Delta_\oplus$ is a set of terms, we may also write $S^! = \{[s_1, \ldots, s_n] \mid s_1, \ldots, s_n \in S\}$ for the set of monomials of terms in S, so that $supp(S^!) = supp(S)^!$ for any $S \in \mathbb{Q}^+\langle\Delta_\oplus\rangle$.

We define the *Taylor expansion* $M^* \in \mathbb{Q}^+\langle\Delta_\oplus\rangle$ of $M \in \Lambda_\oplus$ inductively as follows:

$$\begin{aligned} x^* &:= x \\ (\lambda x.N)^* &:= \lambda x.N^* \\ (PQ)^* &:= \langle P^* \rangle (Q^*)^! \\ (P \oplus Q)^* &:= (P^* \oplus \bullet) + (\bullet \oplus Q^*) \quad . \end{aligned}$$

Note that this definition follows the one for the ordinary λ-calculus given by Ehrhard and Regnier [ER08], in the form described in their Lemma 18. We extend it to $\oplus$ by encoding the pair of vectors (P^*, Q^*) as the sum vector $(P^* \oplus \bullet) + (\bullet \oplus Q^*)$.[9]

Example 4.4.2. *We have* $(x \oplus x)^* = (x \oplus \bullet) + (\bullet \oplus x)$ *hence*

$$(\lambda x.(x \oplus x))^* = \lambda x.((x \oplus \bullet) + (\bullet \oplus x)) = (\lambda x.(x \oplus \bullet)) + (\lambda x.(x \oplus \bullet))$$

and

$$(y(x \oplus x))^* = \sum_{n \in \mathbb{N}} \frac{1}{n!} \langle y \rangle [(x \oplus \bullet) + (\bullet \oplus x)]^n = \sum_{n \in \mathbb{N}} \sum_{i=0}^{n} \frac{1}{i!(n-i)!} \langle y \rangle [x \oplus \bullet]^i \cdot [\bullet \oplus x]^{n-i} \quad .$$

Writing $T(M) := supp(M^*)$ for the support of Taylor expansion, we obtain:

$$\begin{aligned} T(x) &= \{x\} \\ T(\lambda x.N) &= \lambda x.T(N) = \{\lambda x.t \mid t \in T(N)\} \\ T(PQ) &= \langle T(P) \rangle T(Q)^! = \{\langle s \rangle [t_1, \ldots, t_n] \mid s \in T(P) \text{ and } t_1, \ldots, t_n \in T(Q)\} \\ T(P \oplus Q) &= (T(P) \oplus \bullet) \cup (\bullet \oplus T(Q)) = \{s \oplus \bullet \mid s \in T(P)\} \cup \{\bullet \oplus t \mid t \in T(Q)\} \end{aligned}$$

so that $M^* = \sum_{s \in T(M)} M^*_s s$.

We can immediately check that Step 2 still holds for our extension of Taylor expansion to $\lambda_\oplus$-terms: we prove that $T(M)$ is always a clique for the coherence relation $\frown \subseteq \Delta_{\oplus}^{(!)} \times \Delta_{\oplus}^{(!)}$ inductively defined by the rules of Figure 4.4. The first four rules are exactly those for the

9. Note that the original notion of Taylor expansion for nondeterministic λ-terms (considered as algebraic λ-terms without coefficients) interprets nondeterministic choice directly as a sum, setting $(M \oplus N)^* = M^* + N^*$ [Ehr10; 12; 13]. Following Tsukada, Asada and Ong [TAO17], we can recover this notion, by erasing the markers $- \oplus \bullet$ and $\bullet \oplus -$, with one *caveat*: in general, this might yield infinite sums of coefficients, because a single resource term without markers may be obtained from infinitely many terms with markers. Define for instance $x \oplus n\bullet$ by analogy with Example 4.3.2: $x \oplus 0\bullet = x$ and $x \oplus (n+1)\bullet = (x \oplus n\bullet) \oplus \bullet$. Then forgetting markers in the sum $\sum_{i=0}^{\infty} x \oplus n\bullet$ yields $\sum_{i=0}^{\infty} x$. And it turns out that normalizing the Taylor expansion of nondeterministic terms does yield such sums: see Example 4.4.6.

$$\frac{}{x \frown x} \qquad \frac{s \frown s'}{\lambda x.s \frown \lambda x.s'} \qquad \frac{s \frown s' \quad \bar{t} \frown \bar{t}'}{\langle s \rangle \bar{t} \frown \langle s' \rangle \bar{t}'} \qquad \frac{t_i \frown t_j \text{ for } 1 \leq i, j \leq n+m}{[t_1, \ldots, t_n] \frown [t_{n+1}, \ldots, t_{n+m}]}$$

$$\frac{s \frown s'}{s \oplus \bullet \frown s' \oplus \bullet} \qquad \frac{s \frown s'}{\bullet \oplus s \frown \bullet \oplus s'} \qquad \frac{}{s \oplus \bullet \frown \bullet \oplus s'}$$

Figure 4.4 – Rules for the coherence relation on $\Delta_{\oplus}^{(!)}$.

ordinary resource calculus [ER08, Section 3], while the last three rules are reminiscent of the definition of the cartesian product of coherence spaces [Gir87, Definition 5]. Again, this is consistent with the fact that we treat $\oplus$ as a pairing construct, denoting an unspecified superposition operation.

Observe that the relation $\frown$ is automatically symmetric, but not reflexive: e.g., $[s, t] \not\frown [s, t]$ when $s \not\frown t$. We say a resource expression e is *uniform* if $e \frown e$, so that uniform expressions form a coherence space in the usual sense. [10] We call *clique* any set E of resource expressions such that $e \frown e'$ for all $e, e' \in E$. In particular, the elements of a clique are necessarily uniform.

We obtain the expected result by a straightforward induction on $\lambda_{\oplus}$-terms:

Theorem 4.4.3 (Step 2)**.** *The Taylor support $T(M)$ is a clique.*

4.4.3 Multiplicity coefficients

We now generalize Step 1 in our generic nondeterministic setting: we can define a multiplicity coefficient $m(s) \in \mathbb{N}$ for each $s \in \Delta_{\oplus}$ so that $M_s^* = \frac{1}{m(s)}$ whenever $s \in T(M)$.

Given any set X and $n \in \mathbb{N}$, we consider the left action of the group $\mathfrak{S}_n$ of all permutations of $\{1, \ldots, n\}$ on the set X^n of n-tuples, defined as follows: if $\vec{a} = (a_1, \ldots, a_n)$ and $\sigma \in \mathfrak{S}_n$ then $[\sigma]\vec{a} = (a_{\sigma^{-1}(1)}, \ldots, a_{\sigma^{-1}(n)})$. Writing $[\sigma]\vec{a} = (a'_1, \ldots, a'_n)$, we obtain $a'_{\sigma(i)} = a_i$. Let us recall that if $\vec{a} \in X^n$, then the stabilizer of $\vec{a}$ is $\mathit{St}(\vec{a}) = \{\sigma \in \mathfrak{S}_n \mid [\sigma]\vec{a} = \vec{a}\}$.

If $\vec{s} = (s_1, \ldots, s_n) \in \Delta_{\oplus}^n$ and $S \in \mathbb{Q}^+\langle \Delta_{\oplus} \rangle$, we write $S^{\vec{s}} = \prod_{i=1}^n S_{s_i}$: observe that this does not depend on the ordering of the s_i's, so if $\bar{s} = [s_1, \ldots, s_n] \in \Delta_{\oplus}^!$, we may as well write $S^{\bar{s}} = S^{(s_1, \ldots, s_n)}$. We obtain:

Lemma 4.4.4. *Let $S \in \mathbb{Q}^+\langle \Delta_{\oplus} \rangle$ and $\bar{s} \in \mathit{supp}(S^!)$. If $\vec{s} = (s_1, \ldots, s_n)$ is an enumeration of $\bar{s}$,* i.e. *$[s_1, \ldots, s_n] = \bar{s}$, then $(S^!)_{\bar{s}} = \dfrac{S^{\bar{s}}}{\mathit{Card}(\mathit{St}(\vec{s}))}$.*

10. Note that, by contrast with the coherence relation considered by Dal Lago and Leventis for the Taylor expansion of probabilistic λ-terms [LL19], $e \frown e'$ does not imply the uniformity of e nor e': we have $s \oplus \bullet \frown \bullet \oplus s'$ without any condition on s and s'. We could adapt our main results with a finer coherence, similar to theirs, requiring $s \frown s$ and $s' \frown s'$ for $s \oplus \bullet \frown \bullet \oplus s'$ to hold: uniform expressions and cliques are the same for both relations. Nonetheless, we find it interesting that this additional hypothesis is not needed for Step 3.

Proof. By the definition of promotion, and by linearity, we obtain

$$S^! = \sum_{n=0}^{\infty} \frac{1}{n!} \sum_{(s_1,\ldots,s_n)\in\Delta_\oplus^n} S^{(s_1,\ldots,s_n)}[s_1,\ldots,s_n] \quad .$$

If $|\bar{s}| = n$, we thus obtain:

$$(S^!)_{\bar{s}} = \mathit{Card}(\{(s_1,\ldots,s_n) \mid [s_1,\ldots,s_n] = \bar{s}\})\frac{S^{\bar{s}}}{n!} \quad .$$

Observing that $\{(s_1,\ldots,s_n) \mid [s_1,\ldots,s_n] = \bar{s}\}$ is the orbit of any enumeration of $\bar{s}$ under the action of $\mathfrak{S}_n$, and that $\mathit{Card}(\mathfrak{S}_n) = n!$, we conclude by Fact 4.2.1. □

Let $s \in \Delta_\oplus$. We inductively define $m(s)$, the *multiplicity coefficient* of s, as follows:

$$\begin{aligned} m(x) &:= 1 \\ \left.\begin{array}{r} m(\lambda x.s) \\ m(s \oplus \bullet) \\ m(\bullet \oplus s) \end{array}\right\} &:= m(s) \\ m(\langle s\rangle \bar{t}) &:= m(s)m(\bar{t}) \\ m([t_1]^{n_1} \cdot \cdots \cdot [t_n]^{n_n}) &:= \prod_{i=1}^{n} n_i!\, m(t_i)^{n_i} \end{aligned}$$

assuming the t_i's are pairwise distinct in the case of a monomial. Again, this definition extends straightforwardly the one given by Ehrhard and Regnier for their resource calculus [ER08, Section 2.2.1], given that $- \oplus \bullet$ and $\bullet \oplus -$ are both linear. Observe that, considering the function m as a vector $m \in \mathbb{Q}^+\langle \Delta_\oplus^{(!)}\rangle$, if $\vec{s}$ is an enumeration of $\bar{s}$ then $m(\bar{s}) = m^{\bar{s}}\,\mathit{Card}(\mathit{St}(\vec{s}))$.

Theorem 4.4.5 (Step 1)**.** *Let $s \in T(M)$. Then $M^*_s = \dfrac{1}{m(s)}$.*

Proof. The only interesting case is that of an application: $M = PQ$. Assume $s \in T(M)$; then $s = \langle u\rangle \bar{v}$ with $u \in T(P)$ and $\bar{v} = [v_1,\ldots,v_n] \in T(Q)^!$. By definition, $M^*_s = (\langle P^*\rangle (Q^*)^!)_{\langle u\rangle \bar{v}} = P^*_u (Q^*)^!_{\bar{v}}$. Setting $\vec{v} = (v_1,\ldots,v_n)$, we obtain $M^*_s = P^*_u (Q^*)^{\bar{v}} / \mathit{Card}(\mathit{St}(\vec{v}))$ by Lemma 4.4.4. By the induction hypothesis applied to P and Q, we obtain $1/P^*_u = m(u)$ and $1/Q^*_{v_i} = m(v_i)$ hence $1/M^*_s = m(u)m^{\bar{v}}\,\mathit{Card}(\mathit{St}(\vec{v})) = m(u)m(\bar{v}) = m(s)$. □

We can as well obtain Step 4 following Ehrhard and Regnier's study of permutations of variables occurrences, but here we choose to depart from their approach. At this point, indeed, we hope the reader will share our opinion that the combinatorics of Taylor expansion is more intimately connected with the action of permutations on the enumerations of monomials occurring in resource expressions.

In the upcoming Section 4.5, we propose to flesh out this viewpoint, and to recast resource expressions as equivalence classes of their rigid (*i.e.* non-commutative) representatives, up to the isomorphisms of a groupoid of permutation terms inductively defined on the syntactic structure.

The other remaining Steps 3 and 5 are purely qualitative properties of the Taylor support. We choose to treat also Step 3 in the rigid setting, to be introduced later, because it is essentially a property of rigid reduction. On the other hand, the commutation of Step 5 can be established directly.

4.4.4 Taylor expansion of Böhm trees

The Taylor expansion of a Böhm tree is obtained as follows. First we extend the definition of Taylor expansion from $\Lambda_\oplus$ to $\Lambda_\oplus^\perp$ by adding the inductive case $\perp^* \coloneqq 0$, hence $T(\perp) = \emptyset$. Then we set $T(BT(M)) \coloneqq \bigcup_{B \in BT(M)} T(B)$.

We can already observe that if $B \in \mathcal{N}$ and $s \in T(B)$ then s is normal: indeed, the absence of redexes is preserved by the inductive definition of Taylor expansion. It follows that any $s \in T(BT(M))$ is normal. Moreover, it is clear that Theorem 4.4.5 extends to term approximants, hence $B_s^* = \frac{1}{m(s)}$ whenever $s \in T(B)$. Thus, it makes sense to define the Taylor expansion of a Böhm tree as: $BT(M)^* \coloneqq \sum_{s \in T(BT(M))} \frac{1}{m(s)} s$.

Example 4.4.6. *Recall from Example 4.3.2 that if we set* $M = \Theta\lambda y.(y \oplus x)$ *then* $BT(M) = \{\perp \oplus nx \mid n \in \mathbb{N}\}$. *Observe that* $T(\perp \oplus nx) = \{(\bullet \oplus x) \oplus i\bullet \mid 0 \leq i < n\}$ *so that* $T(BT(M)) = \{(\bullet \oplus x) \oplus n\bullet \mid n \in \mathbb{N}\}$ *and* $BT(M)^* = \sum_{i=0}^{\infty} (\bullet \oplus x) \oplus n\bullet$, *because* $m((\bullet \oplus x) \oplus n\bullet) = 1$ *for each* $n \in \mathbb{N}$.

We shall achieve Step 5 by showing that the parallel left strategy in $\Lambda_\oplus$ can be simulated in the support of Taylor expansion, and that $T(BT(M))$ is formed by accumulating the normal forms reached from $T(M)$ by this strategy.

First, we extend the operations $\partial_x - \cdot -$, $L(-)$ and $NF(-)$ to sets of resource expressions in the following way:

$$\partial_x E \cdot \bar{T} \coloneqq \bigcup_{e \in E} \bigcup_{\bar{t} \in \bar{T}} \mathit{supp}(\partial_x e \cdot \bar{t})$$

$$L(E) \coloneqq \bigcup_{e \in E} \mathit{supp}(L(e))$$

$$\mathit{NF}(E) \coloneqq \bigcup_{e \in E} \mathit{supp}(\mathit{NF}(e))$$

whenever $E \subseteq \Delta_\oplus^{(!)}$ and $\bar{T} \subseteq \Delta_\oplus$.[11]

11. In contrast with the case of syntactic constructors in Section 4.4.2, extending these operations to infinite linear combinations rather than sets requires some work.

In the case of $\partial_x - \cdot -$, we can show that each expression e' is in the support of finitely many sums of the shape $\partial_x e \cdot \bar{t}$, by observing that the size of the antecedents e and $\bar{t}$ is at most that of e' [13, Lemma 3.7]. Then one can exploit the fact that the redexes fired in the reduction from e to $L(e)$ are pairwise independent, to deduce that each e' is in the support of finitely many sums of the shape $L(e)$: this is a particular case of a result established by the second author for parallel reduction [13, Section 6.2].

Lemma 4.4.7. *Let M be a $\lambda_\oplus$-term. Then $L(T(M)) = T(L(M))$.*

Proof. The proof is the same as for λ-terms [13], the case of $\oplus$ being direct. The base case requires to prove that $T(M[N/x]) = \partial_x T(M) \cdot T(N)^!$, which is done by a straightforward induction on M. $\square$

Lemma 4.4.8. *Let $A, B \in \Lambda_\oplus^\perp$. If $A \leq B$ then $T(A) \subseteq T(B)$.*

Proof. By straightforward induction on the derivation of $A \leq B$. $\square$

Lemma 4.4.9. *For any $M \in \Lambda_\oplus$, $T(\mathcal{N}(M)) = \{s \in T(M) \mid s$ is normal$\}$.*

Proof. We have $T(\mathcal{N}(M)) \subseteq T(M)$ by Lemma 4.4.8 and the obvious fact that $\mathcal{N}(M) \leq M$. We deduce the inclusion $\subseteq$, recalling that the Taylor support of elementary Böhm trees contains normal terms only.

Conversely, if $s \in T(M)$ and s is normal, then either $M = N \oplus P$ and $s = t \oplus \bullet$ or $s = \bullet \oplus u$ with $t \in T(N)$ or $u \in T(P)$; or $M = \lambda\vec{x}.xQ_1 \cdots Q_k$ and $s = \lambda\vec{x}.\langle x\rangle\bar{q}_1 \cdots \bar{q}_k$ with $\bar{q}_i \in T(Q_i)^!$ for $1 \leq i \leq k$. We obtain inductively $t \in T(\mathcal{N}(N))$ or $u \in T(\mathcal{N}(P))$ or $\bar{q}_i \in T(\mathcal{N}(Q_i))^!$ for $1 \leq i \leq k$, and then $s \in T(\mathcal{N}(M))$. $\square$

Step 5 then follows, using the fact that $BT(M)$ is the downwards closure of $\{\mathcal{N}(L^n(M)) \mid n \in \mathbb{N}\}$:

Theorem 4.4.10 (Step 5)**.** *Let $M \in \Lambda_\oplus$. Then $T(BT(M)) = NF(T(M))$.*

Proof. Recall that $NF(T(M)) = \bigcup_{s \in T(M)} supp(NF(s))$. The proof is by double inclusion.

($\subseteq$) Let $t \in T(BT(M))$, *i.e.* $t \in T(B)$ for some $B \in BT(M)$. By the definition of $BT(M)$, there exists $n \in \mathbb{N}$ such that $B \leq \mathcal{N}(L^n(M))$, and then by Lemma 4.4.8 $t \in T(\mathcal{N}(L^n(M)))$. By Lemma 4.4.9, t is normal and $t \in T(L^n(M))$. By Lemma 4.4.7, $t \in L^n(T(M))$, hence there exists $s \in T(M)$ such that $t \in supp(L^n(s))$. Since t is normal, $t \in supp(NF(s))$.

($\supseteq$) If $t \in NF(T(M))$ we can fix $s \in T(M)$ such that $t \in supp(NF(s))$. Then there exists $n \in \mathbb{N}$ such that $NF(s) = L^n(s)$, hence $t \in \bigcup_{s \in T(M)} supp(L^n(s)) = L^n(T(M))$. By Lemma 4.4.7, $t \in T(L^n(M))$ and since t is normal, Lemma 4.4.9 entails that $t \in T(\mathcal{N}(L^n(M)))$. By the definitions of $BT(M)$ and $T(BT(M))$, we have $\mathcal{N}(L^n(M)) \subseteq BT(M)$ and then $T(\mathcal{N}(L^n(M))) \subseteq T(BT(M))$, and we obtain $t \in T(BT(M))$. $\square$

The case of $NF(-)$ is even more intricate because, given an infinite linear combination S of resource terms, the sum $\sum_{s \in \Delta_\oplus} S_s NF(s)$ is not well defined in general — indeed, it is easy to find an infinite family of resource terms, all having the same nonzero normal form. Uniformity is one solution to this issue: if the support of S is a clique then the summands $NF(s)$ for $s \in supp(S)$ have pairwise disjoint supports. This result is the main ingredient of Step 3: it will be our Theorem 4.6.8 below. For a survey of alternative approaches we refer to the study of this subject by the second author [13].

$$\frac{}{x \lhd x} \qquad \frac{a \lhd s}{\lambda x.a \lhd \lambda x.s} \qquad \frac{a \lhd s}{a \oplus \bullet \lhd s \oplus \bullet} \qquad \frac{a \lhd s}{\bullet \oplus a \lhd \bullet \oplus s}$$

$$\frac{c \lhd s \qquad \vec{d} \lhd \bar{t}}{\langle c \rangle \vec{d} \lhd \langle s \rangle \bar{t}} \qquad \frac{a_1 \lhd t_1 \quad \cdots \quad a_n \lhd t_n}{(a_1, \ldots, a_n) \lhd [t_1, \ldots, t_n]}$$

Figure 4.5 – Rules for the rigid representation relation

4.5 The groupoid of permutations of rigid resource terms

4.5.1 Rigid resource terms and permutation terms

We introduce the set D of *rigid resource terms* and the set $D^!$ of *rigid resource monomials* by mutual induction as follows:

$$D \ni a, b, c, d ::= x \mid \lambda x.a \mid \langle a \rangle \vec{b} \mid \bullet \oplus a \mid a \oplus \bullet \qquad D^! \ni \vec{a}, \vec{b}, \vec{c}, \vec{d} ::= (a_1, \ldots, a_n) \quad .$$

Rigid resource terms are considered up to renaming of bound variables: the only difference with resource terms is that rigid monomials are ordered lists rather than finite multisets. We write $|(a_1, \ldots, a_n)| := n$, and $(a_1, \ldots, a_n) :: (a_{n+1}, \ldots, a_{n+m}) := (a_1, \ldots, a_{n+m})$. We write $D^{(!)}$ for either D or $D^!$ and call *rigid resource expression* any rigid term or rigid monomial. Again, for any $r \in D^{(!)}$, we write $n_x(r)$ for the number of free occurrences of the variable x in r, and we use notations and priority conventions similar to those for non rigid expressions: e.g., we may write $\lambda \vec{x}.\langle a \rangle \vec{b}\, \vec{c} \oplus \bullet$ for $(\lambda x_1. \ldots . \lambda x_n.(\langle \langle a \rangle \vec{b} \rangle \vec{c})) \oplus \bullet$.

As we have already stated, rigid resource expressions are nothing but resource expressions for which the order of terms in monomials matter. To make this connexion formal, consider the *representation relation* $\lhd \subseteq D^{(!)} \times \Delta_\oplus^{(!)}$ defined by the rules of Figure 4.5. Observe that the relation $\lhd$ is the graph of a surjection $D^{(!)} \to \Delta_\oplus^{(!)}$: if $r \in D^{(!)}$, there exists a unique $e \in \Delta_\oplus^{(!)}$ such that $r \lhd e$, and then we write $\|r\| := e$; and any $e \in \Delta_\oplus^{(!)}$ has at least one rigid representation $r \lhd e$. Moreover observe that, if $\vec{a} \lhd \bar{t}$ and $|\vec{a}| = n$ then for any $\sigma \in \mathfrak{S}_n$, $[\sigma]\vec{a} \lhd \bar{t}$, *i.e.* $\|[\sigma]\vec{a}\| = \|\vec{a}\|$.

We now introduce a syntax for the trees of permutations that can act on monomials at any depth in a rigid expression. The language of such *permutation expressions* is given as follows:

$$\mathbb{D} \ni \alpha, \beta, \gamma, \delta ::= id_x \mid \lambda x.\alpha \mid \langle \alpha \rangle \tilde{\beta} \mid \alpha \oplus \bullet \mid \bullet \oplus \alpha \qquad \mathbb{D}^! \ni \tilde{\alpha}, \tilde{\beta}, \tilde{\gamma}, \tilde{\delta} ::= (\sigma, (\alpha_1, \ldots, \alpha_n))$$

where x ranges over variables and σ ranges over $\mathfrak{S}_n$ in the pair $(\sigma, (\alpha_1, \ldots, \alpha_n))$. In other words, a *permutation term* (resp. *permutation monomial*) is nothing but a rigid term (resp. rigid monomial), with a permutation attached with each list of arguments. In general, we will simply write $(\sigma, \alpha_1, \ldots, \alpha_n)$ for the permutation monomial $(\sigma, (\alpha_1, \ldots, \alpha_n))$.

We say $\epsilon \in \mathbb{D}^{(!)}$ maps $r \in D^{(!)}$ to $r' \in D^{(!)}$ if the statement $\epsilon : r \cong r'$ is derivable from the rules of Figure 4.6. We then write $r \cong r'$ if there exists some $\epsilon \in \mathbb{D}^{(!)}$ such that $\epsilon : r \cong r'$. As a direct consequence of the definitions, we obtain that $\cong$ is nothing but the equivalence kernel of the function $r \in D^{(!)} \mapsto \|r\| \in \Delta_\oplus^{(!)}$:

$$\frac{}{id_x : x \cong x} \qquad \frac{\alpha : a \cong a'}{\lambda x.\alpha : \lambda x.a \cong \lambda x.a'} \qquad \frac{\gamma : c \cong c' \qquad \delta : \vec{d} \cong \vec{d'}}{\langle\gamma\rangle\delta : \langle c\rangle\vec{d} \cong \langle c'\rangle\vec{d'}}$$

$$\frac{\alpha : a \cong a'}{\alpha \oplus \bullet : a \oplus \bullet \cong a' \oplus \bullet} \qquad \frac{\alpha : a \cong a'}{\bullet \oplus \alpha : \bullet \oplus a \cong \bullet \oplus a'}$$

$$\frac{\sigma \in \mathfrak{S}_n \qquad \alpha_1 : a_1 \cong a'_{\sigma(1)} \qquad \cdots \qquad \alpha_n : a_n \cong a'_{\sigma(n)}}{(\sigma, \alpha_1, \ldots, \alpha_n) : (a_1, \ldots, a_n) \cong (a'_1, \ldots, a'_n)}$$

Figure 4.6 – Permutation expressions as morphisms between rigid expressions

Lemma 4.5.1. *For all* $r, r' \in D^{(!)}$, $r \cong r'$ *iff* $\|r\| = \|r'\|$.

The equivalence classes for $\cong$ are thus exactly the sets of rigid representations of each resource expression. We can moreover organize the permutation expressions witnessing this equivalence relation into a groupoid, whose objects are resource expressions. Observe indeed that, for each $\epsilon \in \mathbb{D}^{(!)}$ there is exactly one pair (r, r') of rigid expressions such that $\epsilon : r \cong r'$. Given $r, r' \in D^{(!)}$, the set of morphisms from r to r' is then $\mathbb{D}^{(!)}(r, r') = \{\epsilon \mid \epsilon : r \cong r'\}$. The composition $\epsilon'\epsilon \in \mathbb{D}^{(!)}(r, r'')$ of $\epsilon \in \mathbb{D}^{(!)}(r, r')$ and $\epsilon' \in \mathbb{D}^{(!)}(r', r'')$ is defined by induction on the syntax of rigid resource expressions in the obvious way: the only interesting case is that of permutation monomials, for which we set $(\sigma', \alpha'_1, \ldots, \alpha'_n)(\sigma, \alpha_1, \ldots, \alpha_n) \coloneqq (\sigma'\sigma, \alpha'_{\sigma(1)}\alpha_1, \ldots, \alpha'_{\sigma(n)}\alpha_n)$. And the identity id_r on r is the same as r, with each variable occurrence x replaced with id_x, and with the identity permutation attached with each monomial. Inverses are also defined inductively, the key case of monomials being: $(\sigma, \alpha_1, \ldots, \alpha_n)^{-1} \coloneqq (\sigma^{-1}, \alpha^{-1}_{\sigma^{-1}(1)}, \ldots, \alpha^{-1}_{\sigma^{-1}(n)})$.

If $\vec{a} = (a_1, \ldots, a_n)$ and $\vec{a}' = (a'_1, \ldots, a'_n)$, we set $\vec{\mathbb{D}}(\vec{a}, \vec{a}') \coloneqq \prod_{i=1}^n \mathbb{D}(a_i, a'_i)$: with rigid monomials as objects, we obtain a groupoid $\vec{\mathbb{D}}$, which is the free strict monoidal category over $\mathbb{D}$. Moreover, $\mathbb{D}^!(\vec{a}, \vec{a}') = \sum_{\sigma \in \mathfrak{S}_n} \vec{\mathbb{D}}(\vec{a}, [\sigma^{-1}]\vec{a}')$: $\mathbb{D}^!$ is the free symmetric strict monoidal category over $\mathbb{D}$. We call *quasi-stabilizer* of $\vec{a}$ the subgroup of $\mathfrak{S}_n$ defined by

$$St_{\cong}(\vec{a}) \coloneqq \{\sigma \in \mathfrak{S}_n \mid \text{for } 1 \leq i \leq n,\ a_i \cong a_{\sigma(i)}\} \quad .$$

Observe that $St_{\cong}(\vec{a}) = St((\|a_1\|, \ldots, \|a_n\|))$ and $\sigma \in St_{\cong}(\vec{a})$ iff $\vec{\mathbb{D}}(\vec{a}, [\sigma^{-1}]\vec{a}) \neq \emptyset$.

Let us write $\mathbb{D}^{(!)}(r)$ for the group of automorphisms of r: $\mathbb{D}^{(!)}(r) \coloneqq \mathbb{D}^{(!)}(r, r)$. Similarly, we will write $\vec{\mathbb{D}}(\vec{a}) \coloneqq \vec{\mathbb{D}}(\vec{a}, \vec{a})$.

Lemma 4.5.2. *For any* $\vec{a} = (a_1, ..., a_n) \in D^!$, $Card(\mathbb{D}^!(\vec{a})) = Card(St_{\cong}(\vec{a}) \times \vec{\mathbb{D}}(\vec{a}))$.

Proof. Since $\mathbb{D}^{(!)}$ is a groupoid, for any morphism $\epsilon : r \cong r'$, postcomposition by ϵ defines a bijection from $\mathbb{D}^{(!)}(r)$ to $\mathbb{D}^{(!)}(r, r')$. It follows that $\mathbb{D}^!(\vec{a}) = \sum_{\sigma \in \mathfrak{S}_n} \vec{\mathbb{D}}(\vec{a}, [\sigma^{-1}]\vec{a}) = \sum_{\sigma \in St_{\cong}(\vec{a})} \prod_{i=1}^n \mathbb{D}(a_i, a_{\sigma(i)})$ is in bijection with $\sum_{\sigma \in St_{\cong}(\vec{a})} \prod_{i=1}^n \mathbb{D}(a_i) = St_{\cong}(\vec{a}) \times \vec{\mathbb{D}}(\vec{a})$. $\square$

We are then able to formalize the interpretation of the multiplicity of a resource term s as the number of permutations of monomials in s leaving any of its writings $a \lhd s$ unchanged:

Lemma 4.5.3. *Let $e \in \Delta_{\oplus}^{(!)}$ and let $r \lhd e$. Then $m(e) = Card(\mathbb{D}^{(!)}(r))$.*

Proof. By induction on the structure of e. We prove the multiset case. Assume $e = \bar{s}$ and $\vec{a} = (a_1, \ldots, a_n) \lhd \bar{s}$. Then we can write $\bar{s} = [s_1, \ldots, s_n]$ so that $a_i \lhd s_i$ and the induction hypothesis gives $m(s_i) = Card(\mathbb{D}^{(!)}(a_i))$ for $1 \leq i \leq n$. Then

$$m(e) = Card(St((s_1, \ldots, s_n))) \prod_{i=1}^{n} Card(\mathbb{D}^{(!)}(a_i)) = Card(St_{\cong}(\vec{a})) \times Card(\vec{\mathbb{D}}(\vec{a})),$$

and we conclude by Lemma 4.5.2. □

4.5.2 Rigid substitution

For any $r \in D^{(!)}$ and $\vec{b} \in D^!$ such that $|\vec{b}| = n_x(r) = n$, we define the n*-linear substitution* $r[\vec{b}/x]$ of $\vec{b}$ for x in r inductively as follows:

$$\begin{aligned}
x[(b)/x] &:= b \\
y[()/x] &:= y \\
(a \oplus \bullet)[\vec{b}/x] &:= a[\vec{b}/x] \oplus \bullet \\
(\bullet \oplus a)[\vec{b}/x] &:= \bullet \oplus a[\vec{b}/x] \\
(\lambda z.a)[\vec{b}/x] &:= \lambda z.a[\vec{b}/x] \\
\langle c \rangle \vec{d}\,[\vec{b}_0 :: \vec{b}_1/x] &:= \langle c[\vec{b}_0/x] \rangle \vec{d}\,[\vec{b}_1/x] \\
(a_1, \ldots, a_n)[\vec{b}_1 :: \cdots :: \vec{b}_n/x] &:= (a_1[\vec{b}_1/x], \ldots, a_n[\vec{b}_n/x]\})
\end{aligned}$$

where we assume that $y \neq x$, $z \notin \{x\} \cup FV(\vec{b})$, $|\vec{b}| = n_x(a)$, $|\vec{b}_0| = n_x(c)$, $|\vec{b}_1| = n_x(\vec{d})$, and $|\vec{b}_i| = n_x(a_i)$ for $1 \leq i \leq n$.

Observe that this substitution is only partially defined. In order to deal with the general case, we will use the nullary sum of rigid expressions $0 \in \mathbb{N}[D^{(!)}]$: again, we consider all the syntactic constructs to be linear so that we may write, e.g., $\lambda x.a$ for $a \in D \cup \{0\}$ with $\lambda x.0 = 0$. We call *partial rigid expressions* the elements of $D^{(!)} \cup \{0\}$: we generally use the same typographic conventions for partial expressions as for regular ones.

Whenever $r \in D^{(!)} \cup \{0\}$ and $\vec{b} \in D^! \cup \{0\}$, we define the *rigid substitution* $r[\vec{b}/x]$ of $\vec{b}$ for the variable x in r as above if $r \in D^{(!)}$, $\vec{b} \in D^!$ and $n_x(r) = |\vec{b}|$, and set $r[\vec{b}/x] := 0$ otherwise.

This rigid version of multilinear substitution will allow us to provide a more formal account of the intuitive definition of the symmetric multilinear substitution $\partial_x e \cdot \bar{u}$, given in Section 4.4.1: having fixed rigid representations $r \lhd e$ and $\vec{b} \lhd \bar{t} = [t_1, \ldots, t_n]$ with $n = n_x(e)$, instead of the ambiguous

$$\sum_{\sigma \in \mathfrak{S}_n} e[t_{\sigma(1)}/x_1, \ldots, t_{\sigma(n)}/x_n]$$

we can write

$$\sum_{\sigma \in \mathfrak{S}_n} \left\| r[[\sigma]\vec{b}/x] \right\| \quad .$$

To prove that this coincides with the inductive definition of $\partial_x e \cdot \bar{u}$, we need to study how the elements of $\vec{b}$ are routed to subexpressions of r in the substitution $r[[\sigma]\vec{b}/x]$.

For this, we will rely on the following constructions on permutations. First, if $\sigma \in \mathfrak{S}_n$ and $\tau \in \mathfrak{S}_p$, we define the *concatenation* $\sigma \otimes \tau \in \mathfrak{S}_{n+p}$ by:

$$(\sigma \otimes \tau)(i) \coloneqq \sigma(i) \qquad \text{and} \qquad (\sigma \otimes \tau)(n+j) \coloneqq n + \tau(j)$$

for $1 \leq i \leq n$ and $1 \leq j \leq p$. This operation is associative and, more generally, we obtain $\tau_1 \otimes \cdots \otimes \tau_n \in \mathfrak{S}_{k_1+\cdots+k_n}$ whenever $\tau_1 \in \mathfrak{S}_{k_1}, \dots, \tau_n \in \mathfrak{S}_{k_n}$. The tensor product notation is justified since, in the category $\mathbb{P}$ of natural numbers and permutations, the concatenation of permutations defines a tensor product (which is the sum of natural numbers on objects).

Moreover, for each $(I_1, \dots, I_n) \in \mathcal{Q}_n(k)$, writing $I_j = \{i^j_1, \dots, i^j_{k_j}\}$ with $i^j_1 < \cdots < i^j_{k_j}$, we set $\gamma_{I_1,\dots,I_n}(i^j_l) \coloneqq l + \sum_{r=1}^{j-1} k_r$: then $\gamma_{I_1,\dots,I_n}$ is the unique permutation $\gamma \in \mathfrak{S}_k$ such that the map $(j,l) \mapsto \gamma(i^j_l)$ is strictly increasing, considering the lexicographic order on pairs.

Given a weak n-composition of k, *i.e.* a tuple $(k_1, \dots, k_n) \in \mathbb{N}^n$ such that $k = \sum_{j=1}^n k_j$, we write $\mathcal{Q}_n^{k_1,\dots,k_n}(k)$ for the set of those $(I_1, \dots, I_n) \in \mathcal{Q}_n(k)$ such that $Card(I_i) = k_i$ for $1 \leq i \leq n$. We obtain:

Lemma 4.5.4. *For any weak n-composition $(k_1, \dots, k_n)$ of k, the function*

$$\mathcal{Q}_n^{k_1,\dots,k_n}(k) \times \prod_{j=1}^{n} \mathfrak{S}_{k_j} \to \mathfrak{S}_k$$
$$((I_1, \dots, I_n), (\sigma_1, \dots, \sigma_n)) \mapsto (\sigma_1 \otimes \cdots \otimes \sigma_n)\gamma_{I_1,\dots,I_n}$$

is bijective.

Proof. The inverse function is as follows: given $\sigma \in \mathfrak{S}_k$, we fix $I_j \coloneqq \{i \in \{1, \dots, k\} \mid \sum_{r=1}^{j-1} k_r < \sigma(i) \leq \sum_{r=1}^{j} k_r\}$; then, using the above notations for the elements of I_j, for each $l \in \{1, \dots, k_j\}$, we fix $\sigma_j(l) \in \{1, \dots, k_j\}$ to be the unique l' such that $\sigma(i^j_l) = l' + \sum_{r=1}^{j-1} k_r$. □

Now we can show that the two definitions of symmetric multilinear substitution coincide:

Lemma 4.5.5. *If $r \lhd e$ and $\vec{b} \lhd \bar{t}$ then $n_x(r) = n_x(e)$ and $|\vec{b}| = |\bar{t}|$. Moreover $\partial_x e \cdot \bar{t} = \sum_{\sigma \in \mathfrak{S}_{|\vec{b}|}} \big\| r[[\sigma]\vec{b}/x] \big\|$.*

Proof. The first two identities follow directly from the definitions. If $n_x(r) \neq |\vec{b}|$ then both sides of the third identity are 0. Otherwise, it is proved by induction on r.

Let us treat the case of a monomial: write $r = (a_1, \dots, a_n)$ and $e = [s_1, \dots, s_n]$ with $a_i \lhd s_i$ for $1 \leq i \leq n$. Then

$$\begin{aligned}
\partial_x e \cdot \bar{t} &= \sum_{(I_1,\dots,I_n) \in \mathcal{Q}_n(|\vec{b}|)} [\partial_x s_1 \cdot \bar{t}_{I_1}, \dots, \partial_x s_n \cdot \bar{t}_{I_n}] \\
&= \sum_{(I_1,\dots,I_n) \in \mathcal{Q}_n^{k_1,\dots,k_n}(|\vec{b}|)} [\partial_x s_1 \cdot \bar{t}_{I_1}, \dots, \partial_x s_n \cdot \bar{t}_{I_n}]
\end{aligned}$$

where we write $k_i = n_x(s_i)$ for $1 \leq i \leq n$.

If $I \subseteq \{1, \dots, |\vec{b}|\}$ then we write $\vec{b}_I = (b_{i_1}, \dots, b_{i_k})$ where $i_1 < \dots < i_k$ enumerate I. By induction hypothesis we obtain

$$\begin{aligned}\partial_x e \cdot \bar{t} &= \sum_{(I_1,\dots,I_n) \in \mathcal{Q}_n^{k_1,\dots,k_n}(|\vec{b}|)} \left[\sum_{\sigma_1 \in \mathfrak{S}_{k_1}} \left\| a_1[[\sigma_1]\vec{b}_{I_1}/x] \right\|, \dots, \sum_{\sigma_n \in \mathfrak{S}_{k_n}} \left\| a_n[[\sigma_n]\vec{b}_{I_n}/x] \right\| \right] \\ &= \sum_{(I_1,\dots,I_n) \in \mathcal{Q}_n^{k_1,\dots,k_n}(|\vec{b}|)} \sum_{\sigma_1 \in \mathfrak{S}_{k_1}} \cdots \sum_{\sigma_n \in \mathfrak{S}_{k_n}} \left\| r\left[[\sigma_1]\vec{b}_{I_1} :: \dots :: [\sigma_n]\vec{b}_{I_n}/x\right] \right\|\end{aligned}$$

and we conclude, observing that $[\sigma_1]\vec{b}_{I_1} :: \dots :: [\sigma_n]\vec{b}_{I_n} = [(\sigma_1 \otimes \dots \otimes \sigma_n)\gamma_{I_1,\dots,I_n}]\vec{b}$, hence the families

$$\left([\sigma_1]\vec{b}_{I_1} :: \dots :: [\sigma_n]\vec{b}_{I_n}\right)_{(I_1,\dots,I_n) \in \mathcal{Q}_n^{k_1,\dots,k_n}(|\vec{b}|),\ (\sigma_1,\dots,\sigma_n) \in \mathfrak{S}_{k_1} \times \dots \times \mathfrak{S}_{k_n}}$$

and $\left([\sigma]\vec{b}\right)_{\sigma \in \mathfrak{S}_{|\vec{b}|}}$ coincide up to reindexing *via* the bijection of Lemma 4.5.4. □

Informally, everything thus works out as if $[s_1, \dots, s_n] = \sum_{\sigma \in \mathfrak{S}_n} (s_1, \dots, s_n)$, which is to be related with the $\frac{1}{n!}$ coefficient in the Taylor expansion, cancelling out the cardinality of $\mathfrak{S}_n$. Forgetting about coefficients, we obtain:

Corollary 4.5.6. *If $r \lhd e$ and $\vec{b} \lhd \bar{t}$ with $n_x(e) = |\bar{t}|$, then $supp(\partial_x e \cdot \bar{t}) = \{\left\|r[[\sigma]\vec{b}/x]\right\| \mid \sigma \in \mathfrak{S}_{|\vec{b}|}\}$.*

Conversely, any rigid representative of a symmetric substitution is obtained as a rigid substitution:

Lemma 4.5.7. *If $r' \lhd e' \in supp(\partial_x e \cdot \bar{t})$ then $n_x(e) = |\bar{t}|$ and there exist $r \lhd e$ and $\vec{b} \lhd \bar{t}$ such that $r' = r[\vec{b}/x]$.*

Proof. By induction on e. If $e = x$ then $\bar{t} = [t]$ for some $t \in \Delta_\oplus$ and $e' = t$. If $r' \lhd e' = t$ then we can set $r = x$ and $\vec{b} = (r')$. If $e = y \neq x$ then $\bar{t} = []$ and we can set $r = y$ and $\vec{b} = ()$. The abstraction and sum cases follow immediately from the induction hypothesis.

If $e = \langle s \rangle \bar{v}$, we write $\bar{t} = [t_1, \dots, t_n]$ and obtain

$$\partial_x e \cdot \bar{t} = \sum_{(I_1,I_2) \in \mathcal{Q}_2(n)} \langle \partial_x s \cdot \bar{t}_{I_1} \rangle \partial_x \bar{v} \cdot \bar{t}_{I_2} \quad .$$

Then $e' = \langle s' \rangle \bar{v}'$ with $s' \in supp(\partial_x s \cdot \bar{t}_{I_1})$ and $\bar{v}' \in supp(\partial_x \bar{v} \cdot \bar{t}_{I_2})$ for some $(I_1, I_2) \in \mathcal{Q}_2(n)$. It follows that $r' = \langle a \rangle \vec{d}$ with $a \lhd s'$ and $\vec{d} \lhd \bar{v}'$. By induction hypothesis, we obtain $c_1 \lhd s$, $\vec{b}_1 \lhd \bar{t}_{I_1}$, $\vec{c}_2 \lhd \bar{v}$ and $\vec{b}_2 \lhd \bar{t}_{I_2}$ such that $a = c_1[\vec{b}_1/x]$ and $\vec{d} = \vec{c}_2[\vec{b}_2/x]$. Then we conclude by setting $r = \langle c_1 \rangle \vec{c}_2 \lhd \langle s \rangle \bar{v} = e$ and $\vec{b} = \vec{b}_1 :: \vec{b}_2 \lhd \bar{t}_{I_1} \cdot \bar{t}_{I_2} = \bar{t}$.

The case of monomials is similar. □

4.5.3 Substitution for permutation expressions

The key intermediate result for Step 4 is the fact that if $e \bigcirc e$ and $e' \in supp(\partial_x e \cdot \bar{t})$ then $(\partial_x e \cdot \bar{t})_{e'} = \frac{m(e)m(\bar{t})}{m(e')}$: this will be established in Lemma 4.5.20, which concludes the present section. With that goal in mind, and having characterized $m(e)$ as the cardinality of the group $\mathbb{D}^{(!)}(r)$ for any $r \lhd e$, it becomes essential to study how the automorphisms of $r' \lhd e' \in supp(\partial_x e \cdot \bar{t})$ are related with those of some $r \lhd e$ and $\vec{b} \lhd \bar{t}$: by Lemma 4.5.7, we can choose r and $\vec{b}$ such that $r' = r[\vec{b}/x]$. Then it seems natural to consider some form of substitution for permutation expressions, following the structure of rigid substitution.

We define the *substitution of permutation terms for a variable* as follows. Given $\epsilon \in \mathbb{D}^{(!)}(r, r')$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, \vec{b}')$ with $|\vec{b}| = n_x(r)$, we construct $\epsilon[\vec{\beta}/x]$ by induction on ϵ:

$$
\begin{aligned}
(id_x)[(\beta)/x] &:= \beta \\
(id_y)[()/x] &:= id_y \\
(\lambda y.\alpha)[\vec{\beta}/x] &:= \lambda y.\alpha[\vec{\beta}/x] \\
(\alpha \oplus \bullet)[\vec{\beta}/x] &:= \alpha[\vec{\beta}/x] \oplus \bullet \\
(\bullet \oplus \alpha)[\vec{\beta}/x] &:= \bullet \oplus \alpha[\vec{\beta}/x] \\
(\langle\gamma\rangle\tilde{\delta})[\vec{\beta}_1 :: \vec{\beta}_2/x] &:= \langle\gamma[\vec{\beta}_1/x]\rangle\tilde{\delta}[\vec{\beta}_2/x] \\
(\sigma, (\alpha_1, \dots, \alpha_n))[\vec{\beta}_1 :: \dots :: \vec{\beta}_n/x] &:= (\sigma, (\alpha_1[\vec{\beta}_1/x], \dots, \alpha_n[\vec{\beta}_n/x]))
\end{aligned}
$$

where we assume that $y \neq x$, $z \notin \{x\} \cup FV(\vec{\beta})$, $|\vec{\beta}_1| = n_x(\gamma)$, $|\vec{\beta}_2| = n_x(\delta)$, and $|\vec{\beta}_i| = n_x(\alpha_i)$ for $1 \leq i \leq n$.

If $\epsilon \in \mathbb{D}^{(!)}(r, r')$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, \vec{b}')$, the source of $\epsilon[\vec{\beta}/x]$ is obviously $r[\vec{b}/x]$ but describing its target is more intricate: in general, $\epsilon[\vec{\beta}/x] \notin \mathbb{D}^{(!)}(r[\vec{b}/x], r'[\vec{b}'/x])$.

Example 4.5.8. *Consider the rigid monomials $\vec{a} = (x, x)$ and $\vec{b} = (\langle z\rangle(), \langle z\rangle(z))$. Writing τ for the unique transposition of $\mathfrak{S}_2$, we obtain $\alpha = (\tau, id_x, id_x) \in \mathbb{D}^!(\vec{a})$. Let $\vec{\beta} = (id_{\langle z\rangle()}, id_{\langle z\rangle(z)}) \in \vec{\mathbb{D}}(\vec{b})$. Then $\alpha[\vec{\beta}/x] = (\tau, id_{\langle z\rangle()}, id_{\langle z\rangle(z)})$, hence $\alpha[\vec{\beta}/x] : a[\vec{b}/x] \cong (\langle z\rangle(z), \langle z\rangle()) \neq a[\vec{b}/x]$.*

To describe the image of $r[\vec{b}/x]$ through $\epsilon[\vec{\beta}/x]$, we first introduce another operation on permutations. If $\sigma \in \mathfrak{S}_n$ and $\tau_i \in \mathfrak{S}_{k_i}$ for $1 \leq i \leq n$, we define the *multiplexing* $\sigma \cdot (\tau_1, \dots, \tau_n) \in \mathfrak{S}_{k_1 + \dots + k_n}$ by:

$$(\sigma \cdot (\tau_1, \dots, \tau_n))\left(l + \sum_{j=1}^{i-1} k_j\right) := \tau_i(l) + \sum_{j=1}^{\sigma(i)-1} k_{\sigma^{-1}(j)}$$

for $1 \leq i \leq n$ and $1 \leq l \leq k_i$. Multiplexing may be described in the category $\mathbb{P}$ of natural numbers and permutations, which is symmetric strict monoidal, as follows: $\sigma \cdot (\tau_1, \dots, \tau_n) = \sigma_{k_1,\dots,k_n} \circ (\tau_1 \otimes \dots \otimes \tau_n)$ where $\sigma_{k_1,\dots,k_n}$ is the canonical symmetry map $k_1 + \dots + k_n \to k_{\sigma^{-1}(1)} + \dots + k_{\sigma^{-1}(n)} = k_1 + \dots + k_n$ associated with the left action of σ on n-ary tensor products in $\mathbb{P}$. This decomposition of multiplexing is depicted in Figure 4.7.

Multiplexed permutations compose as follows:

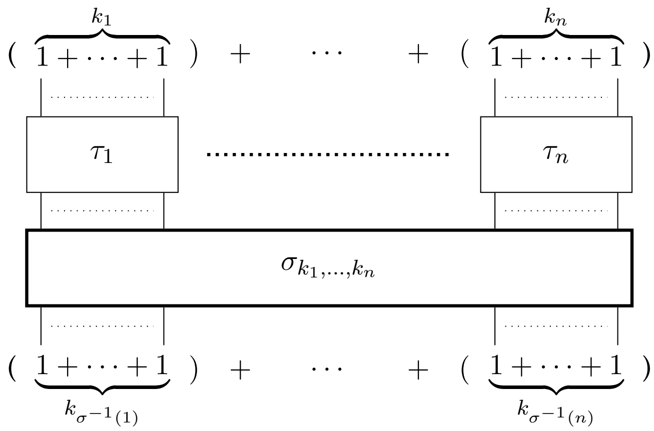

Figure 4.7 – Graphical representation of $\sigma \cdot (\tau_1, \ldots, \tau_n)$

Lemma 4.5.9. *If $\sigma, \sigma' \in \mathfrak{S}_n$, $\tau_i \in \mathfrak{S}_{k_i}$ and $\tau'_i \in \mathfrak{S}_{k_{\sigma^{-1}(i)}}$ for $1 \leq i \leq n$, then*

$$\big(\sigma' \cdot (\tau'_1, ..., \tau'_n)\big)\big(\sigma \cdot (\tau_1, ..., \tau_n)\big) = (\sigma'\sigma) \cdot (\tau'_{\sigma(1)}\tau_1, \ldots, \tau'_{\sigma(n)}\tau_n)$$

and

$$\big(\sigma \cdot (\tau_1, ..., \tau_n)\big)^{-1} = \sigma^{-1} \cdot (\tau^{-1}_{\sigma^{-1}(1)}, \ldots, \tau^{-1}_{\sigma^{-1}(n)}) \quad .$$

Proof. We detail the proof only in case the result is not obvious to the reader from the above categorical presentation of multiplexing. Let $\alpha = \sigma \cdot (\tau_1, ..., \tau_n)$ and $\alpha' = \sigma' \cdot (\tau'_1, ..., \tau'_n)$. For $1 \leq i \leq n$ and $1 \leq l \leq k_i$:

$$\begin{aligned}
\alpha'\Big(\alpha\big(\sum_{j=1}^{i-1} k_j + l\big)\Big) &= \alpha'\Big(\sum_{j=1}^{\sigma(i)-1} k_{\sigma^{-1}(j)} + \tau_i(l)\Big) \\
&= \sum_{j=1}^{\sigma'(\sigma(i))-1} k'_{\sigma'^{-1}(j)} + \tau'_{\sigma(i)}(\tau_i(l)) \qquad \text{(writing } k'_i = k_{\sigma^{-1}(i)}\text{)} \\
&= \sum_{j=1}^{(\sigma'\sigma)(i)-1} k_{(\sigma'\sigma)^{-1}(j)} + (\tau'_{\sigma(i)}\tau_i)(l)
\end{aligned}$$

which establishes the first identity. The second identity follows directly. $\square$

The action of multiplexed permutations on sequences is as follows:

Lemma 4.5.10. *Let $\vec{b}, \vec{b}_1, \ldots, \vec{b}_n \in D^!$, $\sigma \in \mathfrak{S}_n$ and $\tau_i \in \mathfrak{S}_{|\vec{b}_i|}$ for all $i \in \{1, \ldots, n\}$. If $\vec{b} = \vec{b}_1 :: \cdots :: \vec{b}_n$ then $[\sigma \cdot (\tau_1, ..., \tau_n)]\vec{b} = [\tau_{\sigma^{-1}(1)}]\vec{b}_{\sigma^{-1}(1)} :: \cdots :: [\tau_{\sigma^{-1}(n)}]\vec{b}_{\sigma^{-1}(n)}$.*

Proof. Again, we detail the proof only in case the result is not obvious from the categorical presentation. Set $|\vec{b}_i| = k_i$, so that $|\vec{b}| = \sum_{i=1}^n k_i$. Write $\vec{b}' = [\sigma \cdot (\tau_1, ..., \tau_n)]\vec{b}$. For $1 \leq p \leq |\vec{b}'| = |\vec{b}| = \sum_{j=1}^n k_{\sigma^{-1}(j)}$, we can write $p = \sum_{j=1}^{i-1} k_{\sigma^{-1}(j)} + l$ with $i \in \{1, ..., n\}$ and $l \in \{1, \ldots, k_{\sigma^{-1}(i)}\}$. Then, by Lemma 4.5.9, $(\sigma \cdot (\tau_1, ..., \tau_n))^{-1}(p) = \sum_{j=1}^{\sigma^{-1}(i)-1} k_j + \tau^{-1}_{\sigma^{-1}(i)}(l)$ and $b'_p = b_{(\sigma\cdot(\tau_1,...,\tau_n))^{-1}(p)} = (\vec{b}_{\sigma^{-1}(i)})_{\tau^{-1}_{\sigma^{-1}(i)}(l)} = ([\tau_{\sigma^{-1}(i)}]\vec{b}_{\sigma^{-1}(i)})_l$. □

We can now define the *restriction* $\epsilon\restriction_x \in \mathfrak{S}_{n_x(r)}$ of $\epsilon \in \mathbb{D}^{(!)}(r, r')$ to the occurrences of x in r, by induction on ϵ:

$$\begin{aligned}
id_x\restriction_x &:= id_{\{1\}} \\
id_y\restriction_x &:= id_\emptyset \\
\left.\begin{array}{r}(\lambda y.\alpha)\restriction_x \\ (\alpha \oplus \bullet)\restriction_x \\ (\bullet \oplus \alpha)\restriction_x\end{array}\right\} &:= \alpha\restriction_x \\
(\langle\gamma\rangle\tilde{\delta})\restriction_x &:= \gamma\restriction_x \otimes \tilde{\delta}\restriction_x \\
(\sigma, \alpha_1, \ldots, \alpha_n)\restriction_x &:= \sigma \cdot (\alpha_1\restriction_x, \cdots, \alpha_n\restriction_x)
\end{aligned}$$

where we assume $x \neq y$. Intuitively $\epsilon\restriction_x$ is the permutation induced by ϵ on the occurrences $x_1, \ldots, x_{n_x(r)}$ of x in r, taken from left to right.

We recall that $\mathbb{P}$ denotes the category of finite cardinals and permutations. For any variable x, we define an application F_x from $\mathbb{D}^{(!)}$ to $\mathbb{P}$ as follows: $F_x(r) := n_x(r)$ and $F_x(\alpha) := \alpha\restriction_x$.

Lemma 4.5.11. *F_x is a functor from $\mathbb{D}^{(!)}$ to $\mathbb{P}$.*

Proof. By induction on permutation expressions. We focus on the composition condition for the list case. Let $\tilde{\alpha} : \vec{a} = (a_1, \ldots, a_n) \cong \vec{b} = (b_1, \ldots, b_n)$ and $\tilde{\beta} : \vec{b} \cong \vec{c} = (c_1, ..., c_n)$. By definition $\tilde{\alpha} = (\sigma, \alpha_1, \cdots, \alpha_n)$ and $\tilde{\beta} = (\tau, \beta_1, \ldots, \beta_n)$, for some σ, τ in $\mathfrak{S}_n$ and with $\alpha_i : a_i \cong b_{\sigma(i)}$ and $\beta_i : b_i \cong c_{\tau(i)}$. The composition $\tilde{\beta}\tilde{\alpha}$ is then defined as the isomorphism $(\tau\sigma, \beta_{\sigma(1)}\alpha_1, \ldots \beta_{\sigma(n)}\alpha_n)$.

We have to prove that $(\tilde{\beta}\tilde{\alpha})\restriction_x = \tilde{\beta}\restriction_x\tilde{\alpha}\restriction_x$, that is

$$(\tau\sigma) \cdot \big((\beta_{\sigma(1)}\alpha_1)\restriction_x, \ldots, (\beta_{\sigma(n)}\alpha_n)\restriction_x\big) = (\tau \cdot (\beta_1\restriction_x, \ldots, \beta_n\restriction_x))(\sigma \cdot (\alpha_1\restriction_x, \cdots, \alpha_n\restriction_x))$$

which is a direct consequence of the inductive hypothesis, $(\beta_{\sigma(i)}\alpha_i)\restriction_x = \beta_{\sigma(i)}\restriction_x\alpha_i\restriction_x$ for $1 \leq i \leq n$, *via* Lemma 4.5.9. □

In particular, the restriction of F_x to the automorphism group of some rigid expression r is a group homomorphism from $\mathbb{D}^{(!)}(r)$ to $\mathfrak{S}_{n_x(r)}$: its image $\mathbb{D}^{(!)}(r)\restriction_x$ is thus a subgroup of $\mathfrak{S}_{n_x(r)}$. This homomorphism will play a crucial rôle in Section 4.5.4.

This operator allows us to describe the image of $\epsilon[\vec{\beta}/x]$ as follows:

Lemma 4.5.12. *If $\epsilon : r \cong r'$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, \vec{b}')$ with $|\vec{\beta}| = n_x(r)$ then $\epsilon[\vec{\beta}/x] : r[\vec{b}/x] \cong r'[[\epsilon\restriction_x]\vec{b}'/x]$.*

$$\frac{}{x \frown x} \qquad \frac{a \frown a'}{\lambda x.a \frown \lambda x.a'} \qquad \frac{c \frown c' \quad \vec{d} \frown \vec{d'}}{\langle c \rangle \vec{d} \frown \langle c' \rangle \vec{d'}} \qquad \frac{b_i \frown b_j \text{ for } 1 \leq i,j \leq n+m}{(b_1, \ldots, b_n) \frown (b_{n+1}, \ldots, b_{n+m})}$$

$$\frac{a \frown a'}{a \oplus \bullet \frown a' \oplus \bullet} \qquad \frac{a \frown a'}{\bullet \oplus s \frown \bullet \oplus s'} \qquad \frac{}{a \oplus \bullet \frown \bullet \oplus a'} \quad .$$

Figure 4.8 – Rules for the coherence relation on $D^{(!)}$.

Proof. By induction on the structure of r. The interesting case is the list case. Assume $r = (a_1, \ldots, a_n)$, $r' = (a'_1, \ldots, a'_n)$, $\epsilon = (\sigma, \alpha_1, \ldots, \alpha_n)$ and $\vec{\beta} = \vec{\beta}_1 :: \cdots :: \vec{\beta}_n$, with $\alpha_i : a_i \cong a'_{\sigma(i)}$, $\vec{b} = \vec{b}_1 :: \cdots :: \vec{b}_n$, $\vec{b'} = \vec{b}'_1 :: \cdots :: \vec{b}'_n$, $|\vec{\beta}_i| = n_x(a_i)$ and $\vec{\beta}_i \in \mathbb{D}(\vec{b}_i, \vec{b}'_i)$. By definition, we have $\alpha[\vec{\beta}/x] = (\sigma, \alpha_1[\vec{\beta}_1/x], \ldots, \alpha_n[\vec{\beta}_n/x])$. Since $\alpha_i : a_i \cong a'_{\sigma(i)}$, we obtain $\alpha_i[\vec{\beta}_i/x] : a_i[\vec{b}_i/x] \cong a'_{\sigma(i)}[[\alpha_i \restriction_x]\vec{b}'_i/x]$ by induction hypothesis.

We obtain

$$\begin{aligned} \alpha[\vec{\beta}/x] : r[\vec{b}/x] &\cong \left(a'_1\left[[\alpha_{\sigma^{-1}(1)}\restriction_x]\vec{b}'_{\sigma^{-1}(1)}/x\right], \ldots, a'_n\left[[\alpha_{\sigma^{-1}(n)}\restriction_x]\vec{b}'_{\sigma^{-1}(n)}/x\right]\right) \\ &= r'\left[[\alpha_{\sigma^{-1}(1)}\restriction_x]\vec{b}'_{\sigma^{-1}(1)} :: \cdots :: [\alpha_{\sigma^{-1}(n)}\restriction_x]\vec{b}'_{\sigma^{-1}(n)}/x\right] \end{aligned}$$

and we conclude by Lemma 4.5.10. □

4.5.4 The combinatorics of permutation expressions under coherent substitution

Substitution is injective on parallel permutation expressions, in the following sense:

Lemma 4.5.13. *Let $r, r' \in D^{(!)}$ and $\vec{b}, \vec{b'} \in D^!$ with $|\vec{b}| = n_x(r)$ and $|\vec{b'}| = n_x(r')$, and let $\epsilon, \epsilon' \in \mathbb{D}^{(!)}(r, r')$ and $\vec{\beta}, \vec{\beta'} \in \vec{\mathbb{D}}(\vec{b}, \vec{b'})$. If $\epsilon[\vec{\beta}/x] = \epsilon'[\vec{\beta'}/x]$ then $\epsilon = \epsilon'$ and $\vec{\beta} = \vec{\beta'}$.*

Proof. By a straightforward induction on the structure of r. □

On the other hand, surjectivity does not hold in general, because the substitution might enable new morphisms $r[\vec{b}/x] \cong r'[\vec{b'}/x]$, not induced by morphisms in $\mathbb{D}^{(!)}(r, r')$ and $\vec{\mathbb{D}}(\vec{b}, \vec{b'})$:

Example 4.5.14. *Let $a = \langle\langle y\rangle(x)\rangle\langle z\rangle(x)$, $a' = \langle\langle x\rangle(y)\rangle\langle z\rangle(x)$ and $\vec{b} = (y, z)$. Then $a[\vec{b}/x] = a'[\vec{b}/x]$ but $a \not\cong a'$.*

Observe that, in the above example, $\|a\| \not\frown \|a'\|$. Indeed, in the following, we will establish that coherence allows to restore a precise correspondence between the permutation expressions on a substitution $r[(b_1, \ldots, b_n)/x]$ and the $(1+n)$-tuples of permutation expressions on r and each of the b_i's respectively. It will be useful to consider the coherence relation defined on rigid expressions by the rules of Figure 4.8, so that $r \frown r'$ iff $\|r\| \frown \|r'\|$. Then we obtain:

Lemma 4.5.15. *Let $r, r' \in D^{(!)}$ and $\vec{b}, \vec{b'} \in D^!$ with $|\vec{b}| = n_x(r)$ and $|\vec{b'}| = n_x(r')$. If $r \frown r'$ then for all $\phi \in \mathbb{D}^{(!)}(r[\vec{b}/x], r'[\vec{b'}/x])$ there exist $\epsilon \in \mathbb{D}^{(!)}(r, r')$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, [\epsilon\restriction_x^{-1}]\vec{b'})$ such that $\phi = \epsilon[\vec{\beta}/x]$.*

Proof. By induction on the structure of r: the coherence hypothesis $r \frown r'$ induces that r and r' are of the same syntactic nature.

If $r = x$ then $r' = x$ and we can write $\vec{b} = (b)$, $\vec{b}' = (b')$ with $\phi : b \cong b'$. Then we set $\epsilon = id_x$ and $\vec{\beta} = (\phi)$. If $r = y \neq x$ then $r' = y$ and $\phi = id_y$, and we set $\epsilon = id_y$ and $\vec{\beta} = ()$. The abstraction, application and sum cases follow straightforwardly from the induction hypotheses. We detail the list case.

We have $r = (a_1, \dots, a_n)$ and $r' = (a'_1, \dots, a'_m)$. Since $\phi : r[\vec{b}/x] \cong r'[\vec{b}'/x]$ we must have $m = n$, $\vec{b} = \vec{b}_1 :: \cdots :: \vec{b}_n$, $\vec{b}' = \vec{b}'_1 :: \cdots :: \vec{b}'_n$ and $\phi = (\sigma, \gamma_1, \dots, \gamma_n)$ with $\gamma_i \in \mathbb{D}^{(!)}(a_i[\vec{b}_i/x], a'_{\sigma(i)}[\vec{b}'_{\sigma(i)}/x])$. Since $r \frown r'$ we have in particular $a_i \frown a'_{\sigma(i)}$ for $1 \leq i \leq n$.

By the induction hypothesis, we obtain $\gamma_i = \alpha_i[\vec{\beta}_i/x]$ with $\alpha_i \in \mathbb{D}^{(!)}(a_i, a'_{\sigma(i)})$ and $\vec{\beta}_i \in \vec{\mathbb{D}}(\vec{b}_i, [\alpha_i\restriction_x^{-1}]\vec{b}'_{\sigma(i)})$. Then by definition $\epsilon := (\sigma, \alpha_1, \dots, \alpha_n) : r \cong r'$ and

$$
\begin{aligned}
\vec{\beta} := \vec{\beta}_1 :: \cdots :: \vec{\beta}_n : \vec{b} &\cong [\alpha_1\restriction_x^{-1}]\vec{b}'_{\sigma(1)} :: \cdots :: [\alpha_n\restriction_x^{-1}]\vec{b}'_{\sigma(n)} \\
&= [\sigma^{-1} \cdot (\alpha_{\sigma^{-1}(1)}\restriction_x^{-1}, \dots, \alpha_{\sigma^{-1}(n)}\restriction_x^{-1})]\vec{b}' \qquad \text{(by Lemma 4.5.10)}
\end{aligned}
$$

and it remains only to prove that $\sigma^{-1} \cdot (\alpha_{\sigma^{-1}(1)}\restriction_x^{-1}, \dots, \alpha_{\sigma^{-1}(n)}\restriction_x^{-1}) = \epsilon\restriction_x^{-1}$, which follows from Lemma 4.5.9. ☐

In particular, we obtain $(\epsilon\restriction_x, \vec{\beta}) \in \mathbb{D}^!(\vec{b}, \vec{b}')$, hence:

Corollary 4.5.16. *If $r \frown r'$ and $r[\vec{b}/x] \cong r'[\vec{b}'/x]$ then $r \cong r'$ and $\vec{b} \cong \vec{b}'$.*

Given $r \lhd e$, $\vec{b} \lhd \bar{t}$ and $e' \in supp(\partial_x e \cdot \bar{t})$ such that $r[\vec{b}/x] \lhd e'$, we are about to determine the coefficient of e' in $\partial_x e \cdot \bar{t}$ by enumerating the permutations σ such that $r[[\sigma]\vec{b}/x] \lhd e'$, *i.e.* $r[[\sigma]\vec{b}/x] \cong r[\vec{b}/x]$. We thus define $\mathcal{H}_x(r, \vec{b}) := \{\sigma \in \mathfrak{S}_{n_x(r)} \mid r[\vec{b}/x] \cong r[[\sigma]\vec{b}/x]\}$ whenever $|\vec{b}| = n_x(r)$.

Lemma 4.5.17. *Let $r \in D^{(!)}$ and $\vec{b} \in D^!$ with $|\vec{b}| = n_x(r)$. If $r \frown r$ then $\mathcal{H}_x(r, \vec{b}) = \mathbb{D}^{(!)}(r)\restriction_x St_{\cong}(\vec{b})$.*

Proof. Let $\tau \in St_{\cong}(\vec{b})$: by definition, we obtain $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, [\tau]\vec{b})$. If moreover $\epsilon \in \mathbb{D}^{(!)}(r)$ then, by Lemma 4.5.12, $\epsilon[\vec{\beta}/x] \in \mathbb{D}^{(!)}(r[\vec{b}/x], r[[\epsilon\restriction_x\tau]\vec{b}/x])$ hence $\epsilon\restriction_x\tau \in \mathcal{H}_x(r, \vec{b})$. It remains only to show that the function $(\epsilon, \tau) \in \mathbb{D}^{(!)}(r) \times St_{\cong}(\vec{b}) \mapsto \epsilon\restriction_x\tau \in \mathcal{H}_x(r, \vec{b})$ is surjective.

If $\sigma \in \mathcal{H}_x(r, \vec{b})$, there exists $\phi \in \mathbb{D}^{(!)}(r[\vec{b}/x], r[[\sigma]\vec{b}/x])$. Since $r \frown r$, we can apply Lemma 4.5.15 and obtain $\epsilon \in \mathbb{D}^{(!)}(r)$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, [\epsilon\restriction_x^{-1}\sigma]\vec{b})$: in particular, $\epsilon\restriction_x^{-1}\sigma \in St_{\cong}(\vec{b})$, and we conclude since $\sigma = \epsilon\restriction_x(\epsilon\restriction_x^{-1}\sigma)$. ☐

Our argument will moreover rely on the following construction: if $|\vec{b}| = n_x(r)$, we set $\mathcal{K}_x(r, \vec{b}) := \{\epsilon \in \mathbb{D}^{(!)}(r) \mid \epsilon\restriction_x \in St_{\cong}(\vec{b})\} = F_x^{-1}(St_{\cong}(\vec{b}))$, which is a subgroup of $\mathbb{D}^{(!)}(r)$ because F_x is a group homomorphism from $\mathbb{D}^{(!)}(r)$ to $\mathfrak{S}_{n_x(r)}$ by Lemma 4.5.11.

Lemma 4.5.18. *Let $r \in D^{(!)}$ and $\vec{b} \in D^!$ with $|\vec{b}| = n_x(r)$. If $r \frown r$ then $Card(\mathbb{D}^{(!)}(r[\vec{b}/x])) = Card(\mathcal{K}_x(r, \vec{b}))Card(\vec{\mathbb{D}}(\vec{b}))$.*

Proof. By Lemma 4.5.12, if $\epsilon \in \mathbb{D}^{(!)}(r)$ and $\vec{\beta} \in \vec{\mathbb{D}}(\vec{b}, [\epsilon\restriction_x^{-1}]\vec{b})$ then $\epsilon[\vec{\beta}/x] \in \mathbb{D}^{(!)}(r[\vec{b}/x])$. If moreover $\epsilon \in \mathcal{K}_x(r, \vec{b})$ then $\epsilon\restriction_x^{-1} \in St_{\cong}(\vec{b})$: as already remarked in the proof of Lemma 4.5.2, this entails that $Card(\vec{\mathbb{D}}(\vec{b}, [\epsilon\restriction_x^{-1}]\vec{b})) = Card(\vec{\mathbb{D}}(\vec{b}))$. It is thus sufficient to establish that the substitution operation $(\epsilon, \vec{\beta}) \mapsto \epsilon[\vec{\beta}/x]$ defines a bijection from $\sum_{\epsilon \in \mathcal{K}_x(r,\vec{b})} \vec{\mathbb{D}}(\vec{b}, [\epsilon\restriction_x^{-1}]\vec{b})$ to $\mathbb{D}^{(!)}(r[\vec{b}/x])$. This fact derives immediately from Lemma 4.5.13 (injectivity) and Lemma 4.5.15 (surjectivity). □

Lemma 4.5.19. *Let $r \in D^{(!)}$ and $\vec{b} \in D^!$ with $r \circ r$ and $|\vec{b}| = n_x(r)$. Then*

$$Card(\mathcal{H}_x(r, \vec{b})) = \frac{Card(\mathbb{D}^{(!)}(r))\, Card(\mathbb{D}^!(\vec{b}))}{Card(\mathbb{D}^{(!)}(r[\vec{b}/x]))} \quad .$$

Proof. Write $k = n_x(r)$. We know that $St_{\cong}(\vec{b})$ and $\mathbb{D}^{(!)}(r)\restriction_x$ are subgroups of $\mathfrak{S}_k$. Lemma 4.5.17 and Fact 4.2.2 entail that

$$Card(\mathcal{H}_x(r, \vec{b})) = \frac{Card(\mathbb{D}^{(!)}(r)\restriction_x)\, Card(St_{\cong}(\vec{b}))}{Card(\mathbb{D}^{(!)}(r)\restriction_x \cap St_{\cong}(\vec{b}))} \quad .$$

Using Lemma 4.5.18, it will thus be sufficient to prove:

$$\frac{Card(\mathbb{D}^{(!)}(r))\, Card(\mathbb{D}^!(\vec{b}))}{Card(\mathcal{K}_x(r, \vec{b}))\, Card(\vec{\mathbb{D}}(\vec{b}))} = \frac{Card(\mathbb{D}^{(!)}(r)\restriction_x)\, Card(St_{\cong}(\vec{b}))}{Card(\mathbb{D}^{(!)}(r)\restriction_x \cap St_{\cong}(\vec{b}))}$$

which simplifies to

$$\frac{Card(\mathbb{D}^{(!)}(r))}{Card(\mathcal{K}_x(r, \vec{b}))} = \frac{Card(\mathbb{D}^{(!)}(r)\restriction_x)}{Card(\mathbb{D}^{(!)}(r)\restriction_x \cap St_{\cong}(\vec{b}))}$$

by Lemma 4.5.2. We conclude by Fact 4.2.3, recalling that $\mathbb{D}^{(!)}(r)\restriction_x = F_x(\mathbb{D}^{(!)}(r))$ and $\mathcal{K}_x(r, \vec{b}) = F_x^{-1}(St_{\cong}(\vec{b}))$. □

Lemma 4.5.20. *Let $e \in \Delta_{\oplus}^{(!)}$ be such that $e \circ e$ and let $\bar{t} \in \Delta_{\oplus}^!$. If $e' \in supp(\partial_x e \cdot \bar{t})$ then $(\partial_x e \cdot \bar{t})_{e'} = \dfrac{m(e)m(\bar{t})}{m(e')}$.*

Proof. Let $r' \lhd e'$ and $k = n_x(e)$. By Lemma 4.5.7 there exists $r \lhd e$ and $\vec{b} \lhd \bar{t}$ such that $r' = r[\vec{b}/x]$. Then, by Lemma 4.5.5, $(\partial_x e \cdot \bar{t})_{e'} = Card(\{\sigma \in \mathfrak{S}_k \mid r[[\sigma]\vec{b}/x] \lhd e'\}) = Card(\mathcal{H}_x(r, \vec{b}))$. Then we conclude by Lemmas 4.5.19 and 4.5.3. □

4.6 Normalizing the Taylor expansion

In this final section we leverage our results on the groupoid of rigid expressions and permutation expressions in order to achieve Steps 3 and 4. This allows us to complete the proof of commutation between Taylor expansion and normalization.

4.6.1 Normalizing resource expressions in a uniform setting

Lemma 4.5.20 is almost sufficient to obtain Step 4, as it fixes the coefficients in a hereditary head reduction step from a uniform expression:

Lemma 4.6.1. *Let $e \in \Delta_{\oplus}^{(!)}$ with $e \coloneq e$. If $e' \in supp(L(e))$ then $(L(e))_{e'} = \dfrac{m(e)}{m(e')}$.*

Proof. By induction on the structure of e applying Lemma 4.5.20 in the redex case: observe indeed that if $e = \lambda\vec{x}.\langle\lambda y.s\rangle\bar{t}\,\bar{u}_1\cdots\bar{u}_k$ then $e' = \lambda\vec{x}.\langle v\rangle\bar{u}_1\cdots\bar{u}_k$ with $v \in supp(\partial_y s\cdot\bar{t})$, and then $(L(e))_{e'} = (\partial_y s\cdot\bar{t})_v = \frac{m(s)m(\bar{t})}{m(v)}$ and we conclude since $\frac{m(e)}{m(e')} = \frac{m(s)m(\bar{t})}{m(v)}$. All the other cases follow directly from the induction hypothesis by multilinearity. □

To iterate Lemma 4.6.1 along the reduction sequence to the normal form, we first need to show that uniformity is preserved by L. As before, we prefer to focus on the rigid setting first, and we will only consider the hereditary head reduction defined as follows: [12]

$$
\begin{array}{rlrl}
L(a\oplus\bullet) &\coloneqq L(a)\oplus\bullet & L(\bullet\oplus a) &\coloneqq \bullet\oplus L(a)\\
L(\lambda\vec{x}.\lambda y.(a\oplus\bullet)) &\coloneqq \lambda\vec{x}.(\lambda y.a\oplus\bullet) & L(\lambda\vec{x}.\lambda y.(\bullet\oplus a)) &\coloneqq \lambda\vec{x}.(\bullet\oplus\lambda y.a)
\end{array}
$$

$$
\begin{aligned}
L(\lambda\vec{x}.\langle\langle a\oplus\bullet\rangle\vec{b}\rangle\vec{c}_1\cdots\vec{c}_k) &\coloneqq \lambda\vec{x}.\langle\langle a\rangle\vec{b}\oplus\bullet\rangle\vec{c}_1\cdots\vec{c}_k\\
L(\lambda\vec{x}.\langle\langle\bullet\oplus a\rangle\vec{b}\rangle\vec{c}_1\cdots\vec{c}_k) &\coloneqq \lambda\vec{x}.\langle\bullet\oplus\langle a\rangle\vec{b}\rangle\vec{c}_1\cdots\vec{c}_k\\
L(\lambda\vec{x}.\langle y\rangle\vec{a}_1\cdots\vec{a}_k) &\coloneqq \lambda\vec{x}.\langle y\rangle L(\vec{a}_1)\cdots L(\vec{a}_k)\\
L((a_1,\ldots,a_k)) &\coloneqq (L(a_1),\ldots,L(a_k))\\
L(\lambda\vec{x}.\langle\lambda y.a\rangle\vec{b}\,\vec{c}_1\cdots\vec{c}_k) &\coloneqq \lambda\vec{x}.\langle a[\vec{b}/y]\rangle\vec{c}_1\cdots\vec{c}_k
\end{aligned}
$$

extended to partial rigid expressions by setting $L(0) \coloneqq 0$. By an analogue of Lemma 4.4.1, for any $r \in D^{(!)}$, there exists $k \in \mathbb{N}$ such that $L^k(r)$ is normal, and then we write $NF(r) = L^k(r)$. Moreover, r is in normal form iff $L(r) = r$.

Lemma 4.6.2. *If $e \in \Delta_{\oplus}^{(!)}$ then:*

1. *$supp(L(e)) = \{\|L(r)\| \mid r \lhd e$ and $L(r) \neq 0\}$;*
2. *$supp(NF(e)) = \{\|NF(r)\| \mid r \lhd e$ and $NF(r) \neq 0\}$.*

Proof. We first prove that $r' \lhd e' \in supp(L(e))$ iff there exists $r \lhd e$ with $r' = L(r)$, which gives the first result: this is done by a straightforward induction on the structure of e, using Corollary 4.5.6 for the β-redex case.

Now fix $k \in \mathbb{N}$ such that $NF(e) = L^k(e)$: by iterating the previous result, we obtain $r' \lhd e' \in supp(NF(e))$ iff there exists $r \lhd e$ with $r' = L^k(r)$. Then we conclude, observing that if $r' \lhd e'$, then r' is in normal form iff e' is. □

12. Note that the reduction from $\langle\lambda x.a\rangle\vec{b}$ to $a[\vec{b}/x]$ is not well behaved in general: its contextual extension is not even confluent, because it forces the order in which variable occurrences are substituted. Consider for instance the term $(\lambda x.\langle\lambda y.\langle y\rangle(x)\rangle(x))(z_1,z_2)$ which has two distinct normal forms: $\langle z_1\rangle z_2$ and $\langle z_2\rangle z_1$. This rigid calculus is thus not very interesting *per se*, and we only consider it as a tool to analyze the dynamics of the resource calculus.

Lemma 4.6.3. *If $r \coh r'$ and $\vec{b} \coh \vec{b}'$ with $n_x(r) = |\vec{b}|$ and $n_x(r') = |\vec{b}'|$ then $r[\vec{b}/x] \coh r'[\vec{b}'/x]$.*

Proof. By a straightforward induction on the derivation of $r \coh r'$. $\square$

Lemma 4.6.4. *For all $r, r' \in D^{(!)}$ such that $r \coh r'$:*

1. *if $L(r) \neq 0$ and $L(r') \neq 0$ then $L(r) \coh L(r')$;*
2. *if $NF(r) \neq 0$ and $NF(r') \neq 0$ then $NF(r) \coh NF(r')$.*

Proof. The first item is easily established by induction on r, using Lemma 4.6.3 in the case of a β-redex. Having fixed k such that both $NF(r) = L^k(r)$ and $NF(r') = L^k(r')$, the second item follows by iterating the first one. $\square$

We have thus established that L preserves coherence of rigid expressions. It follows that L preserves cliques of resource expressions:

Lemma 4.6.5. *If $E \subseteq \Delta_{\oplus}^{(!)}$ is a clique, then both $L(E)$ and $NF(E)$ are cliques.*

Proof. As a direct consequence of Lemmas 4.6.2 and 4.6.4, we obtain that: if $e \coh e'$ then, for all $e_0 \in \mathit{supp}(L(e))$ and $e'_0 \in \mathit{supp}(L(e'))$ (resp. $e_0 \in \mathit{supp}(NF(e))$ and $e'_0 \in \mathit{supp}(NF(e'))$), we have $e_0 \coh e'_0$. The result follows straightforwardly. $\square$

Step 3 amounts to the fact that distinct coherent expressions have disjoint normal forms. In other words, if the normal forms of two coherent expressions intersect on a common element, then they must coincide. This result will follow from the following rigid version, which states that coherent rigid expressions with isomorphic normal forms are isomorphic:

Lemma 4.6.6. *For all $r, r' \in D^{(!)}$ such that $r \coh r'$:*

1. *if $L(r) \cong L(r')$ then $r \cong r'$;*
2. *if $NF(r) \cong NF(r')$ then $r \cong r'$.*

Proof. Observe that $\cong$ is defined on rigid expressions only so that if, e.g., $L(r) \cong L(r')$ then in particular $L(r) \neq 0 \neq L(r')$. The first item is established by induction on r, using Corollary 4.5.16 in the case of a β-redex. Having fixed k such that both $NF(r) = L^k(r)$ and $NF(r') = L^k(r')$, the second item follows by iterating the first one, thanks to Lemma 4.6.4. $\square$

Note that the converse does not hold, even in the uniform case: two uniform, isomorphic and coherent rigid expressions may yield normal forms that are not isomorphic.

Example 4.6.7. *Consider $a = \langle \lambda x.\langle x\rangle(x)\rangle(y \oplus \bullet, \bullet \oplus z)$ and $a' = \langle \lambda x.\langle x\rangle(x)\rangle(\bullet \oplus z, y \oplus \bullet)$. We have $a \coh a'$ and $a \cong a'$ but $NF(a) = L(a) = \langle y \oplus \bullet\rangle(\bullet \oplus z)$ and $NF(a') = L(a') = \langle \bullet \oplus z\rangle(y \oplus \bullet)$, hence $NF(a) \ncong NF(a')$.*

Theorem 4.6.8 (Step 3)**.** *Let $e, e' \in \Delta_{\oplus}^{(!)}$ be such that $e \coh e'$. If $\mathit{supp}(NF(e)) \cap \mathit{supp}(NF(e')) \neq \emptyset$ then $e = e'$.*

Proof. Let $e_0 \in \mathit{supp}(NF(e)) \cap \mathit{supp}(NF(e'))$. By Lemma 4.6.2, there are $r \lhd e$ and $r' \lhd e'$ such that $e_0 = \|NF(r)\| = \|NF(r')\|$. Since $e \coh e'$, we have $r \coh r'$ and, since $NF(r) \cong NF(r')$, we obtain $r \cong r'$ by Lemma 4.6.6, hence $e = e'$. $\square$

By Lemma 4.6.5, L preserves coherence; and thanks to Theorem 4.6.8 we can iterate Lemma 4.6.1 to obtain:

Theorem 4.6.9 (Step 4). *Let $e \in \Delta_{\oplus}^{(!)}$ with $e \coh e$ and let $e' \in supp(NF(e))$. Then*

$$(NF(e))_{e'} = \frac{m(e)}{m(e')} \quad .$$

Proof. Fix n such that $NF(e) = L^n(e)$: since $supp(NF(e)) = L^n(\{e\})$, there exists a sequence $e_0, \ldots, e_n$ such that $e_0 = e$, $e_n = e'$ and $e_i \in supp(L(e_{i-1}))$ for $1 \leq i \leq n$. We prove by induction on n that, given such a sequence, we have $NF(e_0)_{e_n} = m(e_0)/m(e_n)$.

If $n = 0$ the result is trivial. Otherwise, Lemma 4.6.1 gives $L(e_0)_{e_1} = m(e_0)/m(e_1)$. Moreover, Lemma 4.6.5 ensures that $supp(L(e_0))$ is a clique, and in particular $e_1 \coh e_1$ and the induction hypothesis entails $NF(e_1)_{e_n} = m(e_1)/m(e_n)$. Finally, since $e_n \in supp(NF(e_1))$, Theorem 4.6.8 entails $NF(e')_{e_n} = 0$ for each $e' \in supp(L(e_0)) \setminus \{e_1\}$. We obtain

$$NF(e_0)_{e_n} = NF(L(e_0))_{e_n} = L(e_0)_{e_1} NF(e_1)_{e_n} = \frac{m(e_0)}{m(e_1)} \frac{m(e_1)}{m(e_n)} = \frac{m(e_0)}{m(e_n)} \quad .$$

□

4.6.2 Commutation

By assembling all our previous results, we obtain the desired commutation theorem:

Theorem 4.6.10. *Let $M \in \Lambda_{\oplus}$. Then $BT(M)^* = NF(M^*)$.*

Proof. By Theorem 4.4.5

$$M^* = \sum_{s \in T(M)} \frac{1}{m(s)} s$$

and by Theorem 4.4.3 and Theorem 4.6.8 we are allowed to form

$$NF(M^*) \quad = \sum_{s \in T(M)} \frac{1}{m(s)} NF(s) \quad = \sum_{s \in T(M)} \sum_{u \in supp(NF(s))} \frac{NF(s)_u}{m(s)} u$$

the inner sums having pairwise disjoint supports. Then, if $u \in supp(NF(M^*))$, there is a unique $s \in T(M)$ such that $u \in supp(NF(s))$ and we obtain $NF(M^*)_u = \frac{NF(s)_u}{m(s)} = \frac{1}{m(u)}$ by Theorem 4.6.9. We conclude since $supp(NF(M^*)) = T(BT(M))$ by Theorem 4.4.10. □

Chapitre 5

Bibliographie

5.1 Liste des publications de l'auteur

5.1.1 Revues internationales

[1] Lionel VAUX. « The differential $\lambda\mu$-calculus ». In : *Theor. Comput. Sci.* 379.1-2 (2007), p. 166-209. DOI : 10.1016/j.tcs.2007.02.028 (cf. p. 3, 5).

[2] Lionel VAUX. « The algebraic lambda calculus ». In : *Mathematical Structures in Computer Science* 19.5 (2009), p. 1029-1059 (cf. p. 3, 5, 15, 16, 37, 38, 42, 118).

[3] Lionel VAUX. « A non-uniform finitary relational semantics of system T ». In : *RAIRO Theor. Informatics Appl.* 47.1 (2013), p. 111-132. DOI : 10.1051/ita/2012031 (cf. p. 5).

[4] Christine TASSON et Lionel VAUX. « Transport of finiteness structures and applications ». In : *Mathematical Structures in Computer Science* (2016), p. 1-36. DOI : 10.1017/S0960129516000384 (cf. p. 5, 57).

[5] Lionel VAUX. « Normalizing the Taylor expansion of non-deterministic λ-terms, via parallel reduction of resource vectors ». In : *Logical Methods in Computer Science* 15.3 (2019). DOI : 10.23638/LMCS-15(3:9)2019 (cf. p. i, ii, 6, 13).

[6] Jules CHOUQUET et Lionel VAUX AUCLAIR. « An application of parallel cut elimination in multiplicative linear logic to the Taylor expansion of proof nets ». In : *CoRR* abs/1902.05193 (2020). Accepted for publication in Logical Methods in Computer Science. arXiv : 1902.05193 (cf. p. i, 7, 75).

[7] Federico OLIMPIERI et Lionel VAUX AUCLAIR. « On the Taylor expansion of λ-terms and the groupoid structure of their rigid approximants ». In : *CoRR* abs/2008.02665 (2020). Accepted for publication in Logical Methods in Computer Science. arXiv : 2008.02665 (cf. p. ii, 8, 115).

5.1.2 Chapitre de livre

[8] Emmanuel Beffara et Lionel Vaux. « Programmes, preuves et fonctions : le ménage à trois de Curry-Howard ». In : *Informatique Mathématique : une photographie en 2013*. Sous la dir. de Philippe Langlois. Presses universitaires de Perpignan, 2013. ISBN : 9782354121839.

5.1.3 Conférences internationales

[9] Lionel Vaux. « Convolution $\bar{\lambda}\mu$-Calculus ». In : *TLCA*. Sous la dir. de Simona Ronchi Della Rocca. T. 4583. Lecture Notes in Computer Science. Springer, 2007, p. 381-395. ISBN : 978-3-540-73227-3 (cf. p. 3).

[10] Lionel Vaux. « On Linear Combinations of λ-Terms ». In : *RTA*. Sous la dir. de Franz Baader. T. 4533. Lecture Notes in Computer Science. Springer, 2007, p. 374-388. ISBN : 978-3-540-73447-5 (cf. p. 3, 16, 38).

[11] Lionel Vaux. « Differential Linear Logic and Polarization ». In : *TLCA*. Sous la dir. de Pierre-Louis Curien. T. 5608. Lecture Notes in Computer Science. Springer, 2009, p. 371-385. ISBN : 978-3-642-02272-2 (cf. p. 3).

[12] Michele Pagani, Christine Tasson et Lionel Vaux. « Strong Normalizability as a Finiteness Structure via the Taylor Expansion of λ-terms ». In : *Foundations of Software Science and Computation Structures - 19th International Conference, FOSSACS 2016, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2016, Eindhoven, The Netherlands, April 2-8, 2016, Proceedings*. Sous la dir. de Bart Jacobs et Christof Löding. T. 9634. Lecture Notes in Computer Science. Springer, 2016, p. 408-423. DOI : 10.1007/978-3-662-49630-5_24 (cf. p. 6, 15, 16, 18, 19, 62, 64, 66, 78, 116, 117, 121, 128).

[13] Lionel Vaux. « Taylor Expansion, β-Reduction and Normalization ». In : *26th EACSL Annual Conference on Computer Science Logic, CSL 2017, August 20-24, 2017, Stockholm, Sweden*. Sous la dir. de Valentin Goranko et Mads Dam. T. 82. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017, 39 :1-39 :16. ISBN : 978-3-95977-045-3. DOI : 10.4230/LIPIcs.CSL.2017.39. URL : http://www.dagstuhl.de/dagpub/978-3-95977-045-3 (cf. p. 6, 58, 78, 117, 118, 121, 125, 128, 131, 132).

[14] Jules Chouquet et Lionel Vaux Auclair. « An Application of Parallel Cut Elimination in Unit-Free Multiplicative Linear Logic to the Taylor Expansion of Proof Nets ». In : *27th EACSL Annual Conference on Computer Science Logic, CSL 2018, September 4-7, 2018, Birmingham, UK*. Sous la dir. de Dan R. Ghica et Achim Jung. T. 119. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2018, 15 :1-15 :17. ISBN : 978-3-95977-088-0. DOI : 10.4230/LIPIcs.CSL.2018.15. URL : http://www.dagstuhl.de/dagpub/978-3-95977-088-0 (cf. p. 7, 19, 44, 75, 79, 100).

5.1.4 Workshops internationaux

[15] Lionel VAUX. « A Non-uniform Finitary Relational Semantics of System T ». In : *6th Workshop on Fixed Points in Computer Science, FICS 2009, Coimbra, Portugal, September 12-13, 2009.* Sous la dir. de Ralph MATTHES et Tarmo UUSTALU. Institute of Cybernetics, 2009, p. 116-123. URL : `http://cs.ioc.ee/fics09/proceedings/contrib15.pdf` (cf. p. 5).

[16] Lionel VAUX. « On the transport of finiteness structures ». In : 5th International Conference on Topology, Algebra and Categories in Logic. TACL 2011 (Marseille, 26-30 août 2011). Sous la dir. de Luigi SANTOCANALE, Nicola OLIVETTI et Yves LAFONT. Août 2011.

[17] Jules CHOUQUET, Giulio GUERRIERI, Luc PELLISSIER et Lionel VAUX. « Normalization by Evaluation in Linear Logic ». In : *Preproceedings of the International Workshop on Trends in Linear Logic and Applications, TLLA.* Sous la dir. de Stefano GUERRINI. Sept. 2017 (cf. p. 11, 113).

[18] Federico OLIMPIERI et Lionel VAUX AUCLAIR. *On the Taylor expansion of λ-terms and the groupoid structure of their rigid approximants.* 2018 Joint Workshop on Linearity & Trends in Linear Logic and Applications. 2018 (cf. p. 8, 115).

5.1.5 Publications sur l'enseignement et la diffusion scientifique

[19] Martin ANDLER, Farouk BOUCEKKINE, Véronique CHAUVEAU et Lionel VAUX. « Activités périscolaires mathématiques et égalité des chances ». In : *La Gazette des Mathématiciens* 127 (2011), p. 73-75.

[20] Pierre ARNOUX et Lionel VAUX. « Recherche en mathématiques pour les élèves du secondaire : l'exemple des stages Hippocampe ». In : Enseignement des mathématiques et contrat social, Enjeux et défis pour le 21ème siècle. Actes du colloque EMF 2012 (Université de Genève, 3-7 fév. 2012). Sous la dir. de Jean-Luc DORIER et Sylvia COUTAT. Fév. 2012. ISBN : 978-2-8399-1115-3.

[21] Sylvie LARRAS et Lionel VAUX. « Stages Hippocampe en Mathématiques : des lycéens à la rencontre de la recherche universitaire ». In : La Réforme des Programmes de Lycée : et alors ? Actes de colloque IREM (Lyon, 24-25 mai 2013). Sous la dir. de Pascal FRÉTIGNÉ et Christian MERCAT. Mai 2013. ISBN : 978-2-86612-350-5.

5.1.6 Thèse

[22] Lionel VAUX. « λ-calcul différentiel et logique classique : interactions calculatoires ». Thèse de doctorat. Université Aix-Marseille 2, 2007.

5.2 Liste des thèses co-encadrées

[23] Michele ALBERTI. « On operational properties of quantitative extensions of λ-calculus ». Thèse de doct. Aix Marseille Université ; Università di Bologna, déc. 2014. URL : https://hal.inria.fr/tel-01096067 (cf. p. 4, 37, 38, 42, 66, 73).

[24] Thomas LEVENTIS. « Probabilistic lambda-theories ». Thèse de doct. Aix-Marseille Université, déc. 2016. URL : https://tel.archives-ouvertes.fr/tel-01427279 (cf. p. 4, 11, 18, 19, 66, 73, 122).

[25] Jules CHOUQUET. « A geometry of calculus ». Thèse de doct. Université de Paris, déc. 2019. URL : https://hal.archives-ouvertes.fr/tel-02404100 (cf. p. 7, 114).

[26] Federico OLIMPIERI. « Intersection Types and Resource Calculi in the Denotational Semantics of Lambda-Calculus ». Thèse de doct. Aix-Marseille Université ; Università degli Studi Roma Tre, nov. 2020. URL : https://tel.archives-ouvertes.fr/tel-03123485 (cf. p. 8).

[27] Zeinab GALAL. « Bicategorical Orthogonality Constructions for Linear Logic ». Thèse de doct. Université de Paris, sept. 2021 (cf. p. 8).

5.3 Autres publications citées dans le mémoire

[AD08] Pablo ARRIGHI et Gilles DOWEK. « Linear-algebraic lambda-calculus : higher-order, encodings, and confluence. » In : *Rewriting Techniques and Applications, 19th International Conference, RTA 2008, Hagenberg, Austria, July 15-17, 2008, Proceedings*. Sous la dir. d'Andrei VORONKOV. T. 5117. Lecture Notes in Computer Science. Springer, 2008, p. 17-31. ISBN : 978-3-540-70588-8. DOI : 10.1007/978-3-540-70590-1_2 (cf. p. 4, 16, 38, 42).

[AGK20] Beniamino ACCATTOLI, Stéphane GRAHAM-LENGRAND et Delia KESNER. « Tight Typings and Split Bounds, Fully Developed ». In : *Journal of Functional Programming* 30 (jan. 2020), e14. DOI : 10.1017/S095679682000012X (cf. p. 2).

[Ass+14] Ali ASSAF, Alejandro DÍAZ-CARO, Simon PERDRIX, Christine TASSON et al. « Call-by-value, call-by-name and the vectorial behaviour of the algebraic λ-calculus ». In : *Log. Methods Comput. Sci.* 10.4 (2014). DOI : 10.2168/LMCS-10(4:8)2014 (cf. p. 4, 121).

[BC21] Lison BLONDEAU-PATISSIER et Pierre CLAIRAMBAULT. « Positional Injectivity for Innocent Strategies ». In : *6th International Conference on Formal Structures for Computation and Deduction (FSCD 2021)*. Sous la dir. de Naoki KOBAYASHI. T. 195. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany : Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2021, 17 :1-17 :22. ISBN : 978-3-95977-191-7. DOI : 10.4230/LIPIcs.FSCD.2021.17 (cf. p. 11).

[BEM07] Antonio Bucciarelli, Thomas Ehrhard et Giulio Manzonetto. « Not Enough Points Is Enough ». In : *Computer Science Logic.* T. 4646. Lecture Notes in Computer Science. Springer Berlin, 2007, p. 298-312 (cf. p. 14).

[BG13] Alexis Bernadet et Stéphane Graham-Lengrand. « Non-idempotent intersection types and strong normalisation ». In : *Logical Methods in Computer Science* 9.4 (2013). doi : 10.2168/LMCS-9(4:3)2013 (cf. p. 2).

[BKV17] Antonio Bucciarelli, Delia Kesner et Daniel Ventura. « Non-idempotent intersection types for the Lambda-Calculus ». In : *Log. J. IGPL* 25.4 (2017), p. 431-464. doi : 10.1093/jigpal/jzx018 (cf. p. 2).

[BM20] Davide Barbarossa et Giulio Manzonetto. « Taylor subsumes Scott, Berry, Kahn and Plotkin ». In : *Proc. ACM Program. Lang.* 4.POPL (2020), 1 :1-1 :23. doi : 10.1145/3371069 (cf. p. 2, 116, 118).

[Bou93] Gérard Boudol. « The Lambda-Calculus with Multiplicities (Abstract) ». In : *CONCUR '93 : Proceedings of the 4th International Conference on Concurrency Theory.* London, UK : Springer-Verlag, 1993, p. 1-6. isbn : 3-540-57208-2 (cf. p. 23).

[Car07] Daniel de Carvalho. « Sémantiques de la logique linéaire et temps de calcul ». Thèse de doct. Marseille, France : Université d'Aix-Marseille II, 2007. url : http://theses.fr/2007AIX22066 (cf. p. 2, 8, 76).

[Car11] Alberto Carraro. « Models and theories of pure and resource lambda calculi ». Thèse de doct. Università Ca' Foscari Venezia, 2011. url : http://hdl.handle.net/10579/1089 (cf. p. 51).

[Car16] Daniel de Carvalho. « The Relational Model Is Injective for Multiplicative Exponential Linear Logic ». In : *25th EACSL Annual Conference on Computer Science Logic, CSL 2016, August 29 - September 1, 2016, Marseille, France.* Sous la dir. de Jean-Marc Talbot et Laurent Regnier. T. 62. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016, 41 :1-41 :19. isbn : 978-3-95977-022-4. doi : 10.4230/LIPIcs.CSL.2016.41. url : http://www.dagstuhl.de/dagpub/978-3-95977-022-4 (cf. p. 104, 106).

[Car18a] Daniel de Carvalho. « Execution time of λ-terms via denotational semantics and intersection types ». In : *Mathematical Structures in Computer Science* 28.7 (2018), p. 1169-1203. doi : 10.1017/S0960129516000396 (cf. p. 2, 8, 15, 76).

[Car18b] Daniel de Carvalho. « Taylor expansion in linear logic is invertible ». In : *Logical Methods in Computer Science* 14.4 (2018). doi : 10.23638/LMCS-14(4:21)2018 (cf. p. 114).

[CES10] Alberto Carraro, Thomas Ehrhard et Antonino Salibra. « Exponentials with Infinite Multiplicities ». In : *Computer Science Logic, 24th International Workshop, CSL 2010, 19th Annual Conference of the EACSL, Brno, Czech Republic, August 23-27, 2010. Proceedings.* Sous la dir. d'Anuj Dawar et Helmut Veith. T. 6247. Lecture Notes in Computer Science. Springer, 2010, p. 170-184. isbn : 978-3-642-15204-7. doi : 10.1007/978-3-642-15205-4_16 (cf. p. 51).

[CP18] Pierre Clairambault et Hugo Paquet. « Fully Abstract Models of the Probabilistic lambda-calculus ». In : *27th EACSL Annual Conference on Computer Science Logic (CSL 2018)*. Sous la dir. de Dan Ghica et Achim Jung. T. 119. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany : Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2018, 16 :1-16 :17. isbn : 978-3-95977-088-0. doi : 10.4230/LIPIcs.CSL.2018.16 (cf. p. 11).

[CPT11] Daniel de Carvalho, Michele Pagani et Lorenzo Tortora de Falco. « A semantic measure of the execution time in Linear Logic ». In : *Theoretical Computer Science* 412 (avr. 2011). doi : 10.1016/j.tcs.2010.12.017 (cf. p. 113).

[DE11] Vincent Danos et Thomas Ehrhard. « Probabilistic coherence spaces as a model of higher-order probabilistic computation ». In : *Inf. Comput.* 209.6 (2011), p. 966-991. doi : 10.1016/j.ic.2011.02.001 (cf. p. 15, 18, 19, 76, 116).

[Día11] Alejandro Díaz-Caro. « Du typage vectoriel ». Thèse de doct. France : Université de Grenoble, sept. 2011 (cf. p. 38, 42).

[DLMF] *NIST Digital Library of Mathematical Functions*. Version Release 1.0.28 of 2020-09-15. F. W. J. Olver, A. B. Olde Daalhuis, D. W. Lozier, B. I. Schneider, R. F. Boisvert, C. W. Clark, B. R. Miller, B. V. Saunders, H. S. Cohl, and M. A. McClain, eds. url : http://dlmf.nist.gov/ (cf. p. 33).

[DR89] Vincent Danos et Laurent Regnier. « The structure of multiplicatives ». In : *Arch. Math. Log.* 28.3 (1989), p. 181-203 (cf. p. 78, 86).

[Ehr02] Thomas Ehrhard. « On Köthe Sequence Spaces and Linear Logic ». In : *Mathematical Structures in Computer Science* 12.5 (2002), p. 579-623. doi : 10.1017/S0960129502003729 (cf. p. 76).

[Ehr05] Thomas Ehrhard. « Finiteness spaces ». In : *Mathematical Structures in Computer Science* 15.4 (2005), p. 615-646. doi : 10.1017/S0960129504004645 (cf. p. 2, 3, 14, 15, 21-23, 76, 116, 121).

[Ehr10] Thomas Ehrhard. « A Finiteness Structure on Resource Terms ». In : *Proceedings of the 25th Annual IEEE Symposium on Logic in Computer Science, LICS 2010, 11-14 July 2010, Edinburgh, United Kingdom*. IEEE Computer Society, 2010, p. 402-410. doi : 10.1109/LICS.2010.38 (cf. p. 6, 15, 16, 18, 42, 62, 66, 78, 117, 121, 128).

[Ehr12a] Thomas Ehrhard. « Collapsing non-idempotent intersection types ». In : *Computer Science Logic (CSL'12) - 26th International Workshop/21st Annual Conference of the EACSL, CSL 2012, September 3-6, 2012, Fontainebleau, France*. Sous la dir. de Patrick Cégielski et Arnaud Durand. T. 16. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012, p. 259-273. isbn : 978-3-939897-42-2. doi : 10.4230/LIPIcs.CSL.2012.259. url : http://drops.dagstuhl.de/opus/portals/extern/index.php?semnr=12009 (cf. p. 2, 9).

[Ehr12b] Thomas Ehrhard. « The Scott model of linear logic is the extensional collapse of its relational model ». In : *Theoretical Computer Science* 424 (2012), p. 20-45. issn : 0304-3975. doi : https://doi.org/10.1016/j.tcs.2011.11.027 (cf. p. 9).

[Ehr14] Thomas Ehrhard. « A new correctness criterion for MLL proof nets ». In : *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, Vienna, Austria, July 14 - 18, 2014*. 2014, 38 :1-38 :10 (cf. p. 79, 81).

[Ehr18] Thomas Ehrhard. « An introduction to differential linear logic : proof-nets, models and antiderivatives ». In : *Math. Struct. Comput. Sci.* 28.7 (2018), p. 995-1060. doi : 10.1017/S0960129516000372 (cf. p. 7, 19, 21, 76, 81, 102).

[EP13] Jörg Endrullis et Andrew Polonsky. « Infinitary Rewriting Coinductively ». In : *18th International Workshop on Types for Proofs and Programs (TYPES 2011)*. Sous la dir. de Nils Anders Danielsson et Bengt Nordström. T. 19. Leibniz International Proceedings in Informatics (LIPIcs). Dagstuhl, Germany : Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2013, p. 16-27. isbn : 978-3-939897-49-1. doi : 10.4230/LIPIcs.TYPES.2011.16 (cf. p. 10).

[ER03] Thomas Ehrhard et Laurent Regnier. « The differential lambda-calculus ». In : *Theoretical Computer Science* 309.1-3 (2003). doi : 10.1016/S0304-3975(03)00392-X (cf. p. ii, 2-5, 14, 23, 25, 76, 116, 121).

[ER06a] Thomas Ehrhard et Laurent Regnier. « Böhm Trees, Krivine's Machine and the Taylor Expansion of λ-Terms ». In : *CiE*. Sous la dir. d'Arnold Beckmann, Ulrich Berger, Benedikt Löwe et John V. Tucker. T. 3988. Lecture Notes in Computer Science. Springer, 2006, p. 186-197. isbn : 3-540-35466-2 (cf. p. 2, 5, 15, 16, 18, 30, 62, 68, 116, 117, 121).

[ER06b] Thomas Ehrhard et Laurent Regnier. « Differential Interaction Nets ». In : *Theoretical Computer Science* 364.2 (2006). doi : 10.1016/j.tcs.2006.08.003 (cf. p. 3, 7, 30, 76, 77, 112, 116).

[ER08] Thomas Ehrhard et Laurent Regnier. « Uniformity and the Taylor expansion of ordinary lambda-terms ». In : *Theoretical Computer Science* 403.2-3 (2008), p. 347-372. doi : 10.1016/j.tcs.2008.06.001 (cf. p. 2, 5, 14-16, 18, 23, 26, 30, 36, 62, 68, 76, 78, 116, 117, 121, 125, 128-130).

[Fio+08] M. Fiore, N. Gambino, M. Hyland et G. Winskel. « The cartesian closed bicategory of generalised species of structures ». In : *J. of the London Mathematical Society* (2008). doi : 10.1112/jlms/jdm096 (cf. p. 7, 118, 119).

[FM99] Maribel Fernández et Ian Mackie. « A Calculus for Interaction Nets ». In : *Principles and Practice of Declarative Programming, International Conference PPDP'99, Paris, France, September 29 - October 1, 1999, Proceedings*. Sous la dir. de Gopalan Nadathur. T. 1702. Lecture Notes in Computer Science. Springer, 1999, p. 170-187. doi : 10.1007/10704567_10 (cf. p. 82).

[Gir86] Jean-Yves Girard. « The System F of Variable Types, Fifteen Years Later ». In : *Theor. Comput. Sci.* 45.2 (1986), p. 159-192. doi : 10.1016/0304-3975(86)90044-7 (cf. p. 14).

[Gir87] Jean-Yves Girard. « Linear Logic ». In : *Theoretical Computer Science* 50 (1987), p. 1-102 (cf. p. 19, 76, 77, 115, 129).

[Gir88] Jean-Yves Girard. « Normal Functors, Power Series and Lambda-Calculus ». In : *Annals of Pure and Applied Logic* 37.2 (1988), p. 129 (cf. p. 2, 4, 14, 76, 115).

[Gir96] Jean-Yves Girard. *Proof-nets : the parallel syntax for proof-theory*. Sous la dir. d'Aldo Ursini et Paolo Aglianò. Marcel Dekker, New York, 1996 (cf. p. 79, 82).

[Gol13] J.S. Golan. *Semirings and their Applications*. SpringerLink : Bücher. Springer Netherlands, 2013. isbn : 9789401593335 (cf. p. 20).

[GPT16] Giulio Guerrieri, Luc Pellissier et Lorenzo Tortora de Falco. « Computing Connected Proof(-Structure)s From Their Taylor Expansion ». In : *1st International Conference on Formal Structures for Computation and Deduction, FSCD 2016, June 22-26, 2016, Porto, Portugal*. 2016, 20 :1-20 :18 (cf. p. 106, 113, 114).

[Has02] Ryu Hasegawa. « Two applications of analytic functors ». In : *Theor. Comput. Sci.* 272.1-2 (2002), p. 113-175. doi : 10.1016/S0304-3975(00)00349-2 (cf. p. 7, 116).

[Has96] Ryu Hasegawa. « The Generating Functions of Lambda Terms ». In : *First Conference of the Centre for Discrete Mathematics and Theoretical Computer Science, DMTCS 1996, Auckland, New Zealand, December, 9-13, 1996*. Sous la dir. de Douglas S. Bridges, Cristian S. Calude, Jeremy Gibbons, Steve Reeves et al. Springer-Verlag, Singapore, 1996, p. 253-263. isbn : 981-3083-14-X. url : https://www.cs.auckland.ac.nz/research/groups/CDMTCS/conferences/dmtcs96/ (cf. p. 15).

[HH16] Willem Heijltjes et Robin Houston. « Proof equivalence in MLL is PSPACE-complete ». In : *Logical Methods in Computer Science* 12.1 (2016). doi : 10.2168/LMCS-12(1:2)2016 (cf. p. 83).

[Joy86] André Joyal. « Foncteurs analytiques et espèces de structures ». In : *Combinatoire énumérative : Proceedings of the "Colloque de combinatoire énumérative", held at Université du Québec à Montréal, May 28 – June 1, 1985*. Sous la dir. de Gilbert Labelle et Pierre Leroux. Berlin, Heidelberg : Springer Berlin Heidelberg, 1986, p. 126-159. isbn : 978-3-540-47402-9. doi : 10.1007/BFb0072514 (cf. p. 7, 14).

[Ken+97] J.R. Kennaway, J.W. Klop, M.R. Sleep et F.J. de Vries. « Infinitary lambda calculus ». In : *Theoretical Computer Science* 175.1 (1997), p. 93-125. issn : 0304-3975. doi : http://dx.doi.org/10.1016/S0304-3975(96)00171-5 (cf. p. 10, 19).

[Kri90] Jean-Louis Krivine. *Lambda-calcul, types et modèles*. Masson, Paris, 1990 (cf. p. 35).

[Laf90] Yves Lafont. « Interaction Nets ». In : *Conference Record of the Seventeenth Annual ACM Symposium on Principles of Programming Languages, San Francisco, California, USA, January 1990*. Sous la dir. de Frances E. Allen. ACM Press, 1990, p. 95-108. doi : 10.1145/96709.96718 (cf. p. 82).

[Lai+13] Jim Laird, Giulio Manzonetto, Guy McCusker et Michele Pagani. « Weighted Relational Models of Typed Lambda-Calculi ». In : *28th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2013, New Orleans, LA, USA, June 25-28, 2013*. IEEE Computer Society, 2013, p. 301-310. doi : 10.1109/LICS.2013.36 (cf. p. 15, 18, 76, 116).

[Lai16] J. Laird. « Fixed Points In Quantitative Semantics ». In : *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '16, New York, NY, USA, July 5-8, 2016*. Sous la dir. de Martin Grohe, Eric Koskinen et Natarajan Shankar. ACM, 2016, p. 347-356. doi : 10.1145/2933575.2934569 (cf. p. 15, 18, 116).

[Lam92] François Lamarche. « Quantitative Domains and Infinitary Algebras ». In : *Theor. Comput. Sci.* 94.1 (1992), p. 37-62. doi : 10.1016/0304-3975(92)90323-8 (cf. p. 116).

[Lan02] S. Lang. *Algebra*. Graduate Texts in Mathematics. Springer New York, 2002. isbn : 9780387953854 (cf. p. 120).

[Lau02] Olivier Laurent. « Etude de la polarisation en logique ». Thèse de doct. U, mars 2002 (cf. p. 3).

[Lev18] Thomas Leventis. « Probabilistic BöHm Trees and Probabilistic Separation ». In : *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS '18. Oxford, United Kingdom : Association for Computing Machinery, 2018, p. 649-658. isbn : 9781450355834. doi : 10.1145/3209108.3209126 (cf. p. 11).

[Lev19] Thomas Leventis. « A deterministic rewrite system for the probabilistic λ-calculus ». In : *Mathematical Structures in Computer Science* 29.10 (2019), p. 1479-1512. doi : 10.1017/S0960129519000045 (cf. p. 4, 121-123).

[LL19] Ugo Dal Lago et Thomas Leventis. « On the Taylor Expansion of Probabilistic λ-terms ». In : *4th International Conference on Formal Structures for Computation and Deduction, FSCD 2019, June 24-30, 2019, Dortmund, Germany*. Sous la dir. d'Herman Geuvers. T. 131. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, 13 :1-13 :16. doi : 10.4230/LIPIcs.FSCD.2019.13 (cf. p. 11, 19, 116, 118, 129).

[LP95] Ugo de'Liguoro et Adolfo Piperno. « Nondeterministic Extensions of Untyped λ-Calculus ». In : *Information and Computation* 149-177.122 (1995) (cf. p. 16, 18, 37).

[Maz12] Damiano Mazza. « An Infinitary Affine Lambda-Calculus Isomorphic to the Full Lambda-Calculus ». In : *2012 27th Annual IEEE Symposium on Logic in Computer Science*. 2012, p. 471-480 (cf. p. 118).

[MP11] Giulio Manzonetto et Michele Pagani. « Böhm's Theorem for Resource Lambda Calculus through Taylor Expansion ». In : *Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011, Novi Sad, Serbia, June 1-3, 2011. Proceedings*. Sous la dir. de C.-H. Luke Ong. T. 6690. Lecture Notes in Computer Science. Springer, 2011, p. 153-168. doi : 10.1007/978-3-642-21691-6_14 (cf. p. 116).

[MPV18] Damiano Mazza, Luc Pellissier et Pierre Vial. « Polyadic approximations, fibrations and intersection types ». In : 2018. doi : 10.1145/3158094 (cf. p. 118).

[MS08] Ian Mackie et Shinya Sato. « A Calculus for Interaction Nets Based on the Linear Chemical Abstract Machine ». In : *Electron. Notes Theor. Comput. Sci.* 192.3 (2008), p. 59-70. doi : 10.1016/j.entcs.2008.10.027 (cf. p. 82).

[Oli21] Federico Olimpieri. « Intersection Type Distributors ». In : (2021), p. 1-15. doi : 10.1109/LICS52264.2021.9470617 (cf. p. 119).

[Ong17] C.-H. Luke Ong. « Quantitative semantics of the lambda calculus : Some generalisations of the relational model ». In : *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017.* IEEE Computer Society, 2017, p. 1-12. doi : 10.1109/LICS.2017.8005064 (cf. p. 116).

[PT09] Michele Pagani et Christine Tasson. « The Inverse Taylor Expansion Problem in Linear Logic ». In : *Proceedings of the 24th Annual IEEE Symposium on Logic in Computer Science, LICS 2009.* Sous la dir. d'Andrew M. Pitts. IEEE Computer Society, 2009, p. 222-231 (cf. p. 106).

[Reg92] Laurent Regnier. « Lambda-calcul et réseaux ». Thèse de doct. Paris, France : Université Paris 7, déc. 1992 (cf. p. 79).

[Suz82] Michio Suzuki. *Group theory I.* Springer-Verlag Berlin ; New York, 1982 (cf. p. 120).

[TAO17] Takeshi Tsukada, Kazuyuki Asada et C.-H. Luke Ong. « Generalised species of rigid resource terms ». In : *32nd Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2017, Reykjavik, Iceland, June 20-23, 2017.* IEEE Computer Society, 2017, p. 1-12. isbn : 978-1-5090-3018-7. doi : 10.1109/LICS.2017.8005093. url : http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=7999337 (cf. p. 8, 11, 16, 18, 62, 116, 118, 119, 121, 124, 128).

[TAO18] Takeshi Tsukada, Kazuyuki Asada et C.-H. Luke Ong. « Species, Profunctors and Taylor Expansion Weighted by SMCC : A Unified Framework for Modelling Nondeterministic, Probabilistic and Quantum Programs ». In : *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science.* LICS '18. Oxford, United Kingdom : ACM, 2018, p. 889-898. isbn : 978-1-4503-5583-4. doi : 10.1145/3209108.3209157 (cf. p. 116, 118).

[Tas09] Christine Tasson. « Sémantiques et syntaxes vectorielles de la logique linéaire ». Thèse de doct. Paris, France : Université Paris Diderot, déc. 2009 (cf. p. 7, 19, 78).

[Tor00] Lorenzo Tortora de Falco. « Réseaux, cohérence et expériences obsessionnelles ». Thèse de doct. Paris, France : Université Paris 7, 2000 (cf. p. 84).

[Tor03] Lorenzo Tortora de Falco. « Obsessional Experiments For Linear Logic Proof-Nets ». In : *Mathematical Structures in Computer Science* 13.6 (2003), p. 799-855. doi : 10.1017/S0960129503003967 (cf. p. 19).

Résumé Le développement de Taylor des λ-termes et des preuves de la logique linéaire est le fruit d'une relecture syntaxique par Ehrhard et Regnier de la sémantique quantitative de Girard : il associe à chaque terme ou preuve une combinaison linéaire infinie d'approximations multilinéaires et finies de l'objet de départ. Il matérialise une correspondance étroite entre le comportement calculatoire des termes, défini par la β-réduction, et leur interprétation dans certains modèles dénotationnels : le développement de Taylor d'un terme est toujours normalisable, et sa forme normale correspond exactement à l'arbre de Böhm du terme. Cette correspondance se retrouve dans le fait que, pour de nombreux modèles de la logique linéaire, la promotion d'un morphisme s'obtient comme une superposition d'opérations multilinéaires, faisant du développement de Taylor des preuves une structure sous-jacente de ces modèles.

Ce mémoire présente quelques avancées récentes visant à raffiner l'analyse de la normalisation (qui est un processus potentiellement infini) offerte par le développement de Taylor pour la ramener au niveau de la β-réduction ou de l'élimination des coupures (qui correspond à un calcul fini).

On démontre que cette approche permet d'étendre l'analyse à un cadre non-uniforme, susceptible de prendre en compte par exemple une forme de non-déterminisme calculatoire — alors que la normalisation peut échouer dans ce cadre. On démontre également que la même approche peut être appliquée aux réseaux de démonstration de la logique linéaire. Enfin les techniques développées précédemment permettent de revisiter et simplifier le résultat originel d'Ehrhard et Regnier pour la normalisation dans le cas uniforme, tout en l'adaptant à une forme restreinte de non-déterminisme.

Abstract The Taylor expansion of λ-terms, and of linear logic proof trees, was devised by Ehrhard and Regnier after a syntactic reinterpretation of Girard's quantitative semantics : to each term or proof, it associates an infinite linear combination of finite, multilinear approximations of the original object. It embodies a tight correspondence between the computational behavior of terms, as defined by β-reduction, and their interpretation in some particular denotational models : the Taylor expansion of a term is always normalizable, and its normal form is isomorphic to the Böhm tree of that term. This correspondence also shows in the fact that, for many models of linear logic, the promotion of a morphism is obtained by a superposition of multilinear operations : the Taylor expansion of proofs underlies the structure of those models.

The Taylor expansion of terms and proofs thus offers an analysis of normalization — which is a potentially infinite process : in this thesis, we present some recent advances, refining this analysis to the level of a single β-reduction or cut-elimination step — which is always computationally finite.

We show that this approach allows to extend the analysis to a non-uniform setting, which can accommodate a form of computational non-determinism — by contrast, normalization can fail in this setting. We also show that the same approach can be applied to linear logic proof nets. Finally, the previous techniques allow us to revisit the original result of Ehrhard and Regnier for normalization in the uniform case, and to adapt it to a controlled form of non-determinism.