NUMERO D'ORDRE : **1075**

ANNEE : 1993

USTL

LIFL
U.R.A. 369 du C.N.R.S.

LABORATOIRE D'INFORMATIQUE FONDAMENTALE DE LILLE

THESE

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

pour obtenir le titre de

DOCTEUR en INFORMATIQUE

par

Dominique GONZALEZ

Décomposition des semi Commutations

Thèse soutenue le 28 janvier 1993, devant la commission d'examen :

Président :	J.L. DEKEYSER	LIFL
Directeur de Thèse	M. CLERBOUT	LIFL
Rapporteurs :	C. CHOFRUT	LITP
	Y. METIVIER	LABRI
Examinateurs :	M. CLERBOUT	LIFL
	M. LATTEUX	LIFL
	Y. ROOS	LIFL

UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE
U.F.R. d'I.E.E.A. Bât M3. 59655 Villeneuve d'Ascq CEDEX
Tél. 20.43.47.24 Fax. 20.43.65.66

UNIVERSITE DES SCIENCES
ET TECHNOLOGIES DE LILLE

DOYENS HONORAIRES DE L'ANCIENNE FACULTE DES SCIENCES

M. H. LEFEBVRE, M. PARREAU

PROFESSEURS HONORAIRES DES ANCIENNES FACULTES DE DROIT ET SCIENCES ECONOMIQUES, DES SCIENCES ET DES LETTRES

MM. ARNOULT, BONTE, BROCHARD, CHAPPELON, CHAUDRON, CORDONNIER, DECUYPER, DEHEUVELS, DEHORS, DION, FAUVEL, FLEURY, GERMAIN, GLACET, GONTIER, KOURGANOFF, LAMOTTE, LASSERRE, LELONG, LHOMME, LIEBAERT, MARTINOT-LAGARDE, MAZET, MICHEL, PEREZ, ROIG, ROSEAU, ROUELLE, SCHILTZ, SAVARD, ZAMANSKI, Mes BEAUJEU, LELONG.

PROFESSEUR EMERITE

M. A. LEBRUN

ANCIENS PRESIDENTS DE L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE

MM. M. PARREAU, J. LOMBARD, M. MIGEON, J. CORTOIS, A.DUBRULLE

PRESIDENT DE L'UNIVERSITE DES SCIENCES ET TECHNOLOGIES DE LILLE

M. P. LOUIS

PROFESSEURS - CLASSE EXCEPTIONNELLE

M. CHAMLEY Hervé	Géotechnique
M. CONSTANT Eugène	Electronique
M. ESCAIG Bertrand	Physique du solide
M. FOURET René	Physique du solide
M. GABILLARD Robert	Electronique
M. LABLACHE COMBIER Alain	Chimie
M. LOMBARD Jacques	Sociologie
M. MACKE Bruno	Physique moléculaire et rayonnements atmosphériques

M. MIGEON Michel	EUDIL
M. MONTREUIL Jean	Biochimie
M. PARREAU Michel	Analyse
M. TRIDOT Gabriel	Chimie appliquée

PROFESSEURS - 1ère CLASSE

M. BACCHUS Pierre	Astronomie
M. BIAYS Pierre	Géographie
M. BILLARD Jean	Physique du Solide
M. BOILLY Bénoni	Biologie
M. BONNELLE Jean Pierre	Chimie-Physique
M. BOSCQ Denis	Probabilités
M. BOUGHON Pierre	Algèbre
M. BOURIQUET Robert	Biologie Végétale
M. BRASSELET Jean Paul	Géométrie et topologie
M. BREZINSKI Claude	Analyse numérique
M. BRIDOUX Michel	Chimie Physique
M. BRUYELLE Pierre	Géographie
M. CARREZ Christian	Informatique
M. CELET Paul	Géologie générale
M. COEURE Gérard	Analyse
M. CORDONNIER Vincent	Informatique
M. CROSNIER Yves	Electronique
Mme DACHARRY Monique	Géographie
M. DAUCHET Max	Informatique
M. DEBOURSE Jean Pierre	Gestion des entreprises
M. DEBRABANT Pierre	Géologie appliquée
M. DECLERCQ Roger	Sciences de gestion
M. DEGAUQUE Pierre	Electronique
M. DESCHEPPER Joseph	Sciences de gestion
Mme DESSAUX Odile	Spectroscopie de la réactivité chimique
M. DHAINAUT André	Biologie animale
Mme DHAINAUT Nicole	Biologie animale
M. DJAFARI Rouhani	Physique
M. DORMARD Serge	Sciences Economiques
M. DOUKHAN Jean Claude	Physique du solide
M. DUBRULLE Alain	Spectroscopie hertzienne
M. DUPOUY Jean Paul	Biologie
M. DYMENT Arthur	Mécanique
M. FOCT Jacques Jacques	Métallurgie
M. FOUQUART Yves	Optique atmosphérique
M. FOURNET Bernard	Biochimie structurale
M. FRONTIER Serge	Ecologie numérique
M. GLORIEUX Pierre	Physique moléculaire et rayonnements atmosphériques
M. GOSSELIN Gabriel	Sociologie
M. GOUDMAND Pierre	Chimie-Physique
M. GRANELLE Jean Jacques	Sciences Economiques
M. GRUSON Laurent	Algèbre
M. GUILBAULT Pierre	Physiologie animale
M. GUILLAUME Jean	Microbiologie
M. HECTOR Joseph	Géométrie
M. HENRY Jean Pierre	Génie mécanique
M. HERMAN Maurice	Physique spatiale
M. LACOSTE Louis	Biologie Végétale
M. LANGRAND Claude	Probabilités et statistiques

M. LATTEUX Michel	Informatique
M. LAVEINE Jean Pierre	Paléontologie
Mme LECLERCQ Ginette	Catalyse
M. LEHMANN Daniel	Géométrie
Mme LENOBLE Jacqueline	Physique atomique et moléculaire
M. LEROY Jean Marie	Spectrochimie
M. LHENAFF René	Géographie
M. LHOMME Jean	Chimie organique biologique
M. LOUAGE Françis	Electronique
M. LOUCHEUX Claude	Chimie-Physique
M. LUCQUIN Michel	Chimie physique
M. MAILLET Pierre	Sciences Economiques
M. MAROUF Nadir	Sociologie
M. MICHEAU Pierre	Mécanique des fluides
M. PAQUET Jacques	Géologie générale
M. PASZKOWSKI Stéfan	Mathématiques
M. PETIT Francis	Chimie organique
M. PORCHET Maurice	Biologie animale
M. POUZET Pierre	Modélisation - calcul scientifique
M. POVY Lucien	Automatique
M. PROUVOST Jean	Minéralogie
M. RACZY Ladislas	Electronique
M. RAMAN Jean Pierre	Sciences de gestion
M. SALMER Georges	Electronique
M. SCHAMPS Joël	Spectroscopie moléculaire
Mme SCHWARZBACH Yvette	Géométrie
M. SEGUIER Guy	Electrotechnique
M. SIMON Michel	Sociologie
M. SLIWA Henri	Chimie organique
M. SOMME Jean	Géographie
Melle SPIK Geneviève	Biochimie
M. STANKIEWICZ François	Sciences Economiques
M. THIEBAULT François	Sciences de la Terre
M. THOMAS Jean Claude	Géométrie - Topologie
M. THUMERELLE Pierre	Démographie - Géographie humaine
M. TILLIEU Jacques	Physique théorique
M. TOULOTTE Jean Marc	Automatique
M. TREANTON Jean René	Sociologie du travail
M. TURRELL Georges	Spectrochimie infrarouge et raman
M. VANEECLOO Nicolas	Sciences Economiques
M. VAST Pierre	Chimie inorganique
M. VERBERT André	Biochimie
M. VERNET Philippe	Génétique
M. VIDAL Pierre	Automatique
M. WALLART Françis	Spectrochimie infrarouge et raman
M. WEINSTEIN Olivier	Analyse économique de la recherche et développement
M. ZEYTOUNIAN Radyadour	Mécanique

PROFESSEURS - 2ème CLASSE

M. ABRAHAM Francis	
M. ALLAMANDO Etienne	Composants électroniques
M. ANDRIES Jean Claude	Biologie des organismes
M. ANTOINE Philippe	Analyse
M. BALL Steven	Génétique
M. BART André	Biologie animale
M. BASSERY Louis	Génie des procédés et réactions chimiques
Mme BATTIAU Yvonne	Géographie
M. BAUSIERE Robert	Systèmes électroniques
M. BEGUIN Paul	Mécanique
M. BELLET Jean	Physique atomique et moléculaire
M. BERNAGE Pascal	Physique atomique, moléculaire et du rayonnement
M. BERTHOUD Arnaud	Sciences Economiques
M. BERTRAND Hugues	Sciences Economiques
M. BERZIN Robert	Analyse
M. BISKUPSKI Gérard	Physique de l'état condensé et cristallographie
M. BKOUCHE Rudolphe	Algèbre
M. BODARD Marcel	Biologie végétale
M. BOHIN Jean Pierre	Biochimie métabolique et cellulaire
M. BOIS Pierre	Mécanique
M. BOISSIER Daniel	Génie civil
M. BOIVIN Jean Claude	Spectrochimie
M. BOUCHER Daniel	Physique
M. BOUQUELET Stéphane	Biologie appliquée aux enzymes
M. BOUQUIN Henri	Gestion
M. BROCARD Jacques	Chimie
Mme BROUSMICHE Claudine	Paléontologie
M. BUISINE Daniel	Mécanique
M. CAPURON Alfred	Biologie animale
M. CARRE François	Géographie humaine
M. CATTEAU Jean Pierre	Chimie organique
M. CAYATTE Jean Louis	Sciences Economiques
M. CHAPOTON Alain	Electronique
M. CHARET Pierre	Biochimie structurale
M. CHIVE Maurice	Composants électroniques optiques
M. COMYN Gérard	Informatique théorique
Mme CONSTANT Monique	Composants électroniques et optiques
M. COQUERY Jean Marie	Psychophysiologie
M. CORIAT Benjamin	Sciences Economiques
Mme CORSIN Paule	Paléontologie
M. CORTOIS Jean	Physique nucléaire et corpusculaire
M. COUTURIER Daniel	Chimie organique
M. CRAMPON Norbert	Tectolique géodynamique
M. CURGY Jean Jacques	Biologie
M. DANGOISSE Didier	Physique théorique
M. DE PARIS Jean Claude	Analyse
M. DECOSTER Didier	Composants électroniques et optiques
M. DEJAEGER Roger	Electrochimie et Cinétique
M. DELAHAYE Jean Paul	Informatique
M. DELORME Pierre	Physiologie animale
M. DELORME Robert	Sciences Economiques
M. DEMUNTER Paul	Sociologie
Mme DEMUYNCK Claire	Physique atomique, moléculaire et du rayonnement
M. DENEL Jacques	Informatique
M. DEPREZ Gilbert	Physique du solide - christallographie

M. LE MAROIS Henri	Vie de la firme
M. LEMOINE Yves	Biologie et physiologie végétales
M. LESCURE François	Algèbre
M. LESENNE Jacques	Systèmes électroniques
M. LOCQUENEUX Robert	Physique théorique
Mme LOPES Maria	Mathématiques
M. LOSFELD Joseph	Informatique
M. LOUAGE Francis	Electronique
M. MAHIEU François	Sciences économiques
M. MAHIEU Jean Marie	Optique - Physique atomique
M. MAIZIERES Christian	Automatique
M. MANSY Jean Louis	Géologie
M. MAURISSON Patrick	Sciences Economiques
M. MERIAUX Michel	EUDIL
M. MERLIN Jean Claude	Chimie
M. MESMACQUE Gérard	Génie mécanique
M. MESSELYN Jean	Physique atomique et moléculaire
M. MOCHE Raymond	Modélisation.calcul scientifique,statistiques
M. MONTEL Marc	Physique du solide
M. MORCELLET Michel	Chimie organique
M. MORE Marcel	Physique de l'état condensé et cristallographie
M. MORTREUX André	Chimie organique
Mme MOUNIER Yvonne	Physiologie des structures contractiles
M. NIAY Pierre	Physique atomique,moléculaire et du rayonnement
M. NICOLE Jacques	Spectrochimie
M. NOTELET Francis	Systèmes électroniques
M. PALAVIT Gérard	Génie chimique
M. PARSY Fernand	Mécanique
M. PECQUE Marcel	Chimie organique
M. PERROT Pierre	Chimie appliquée
M. PERTUZON Emile	Physiologie animale
M. PETIT Daniel	Biologie des populations et écosystèmes
M. PLIHON Dominique	Sciences Economiques
M. PONSOLLE Louis	Chimie physique
M. POSTAIRE Jack	Informatique industrielle
M. RAMBOUR Serge	Biologie
M. RENARD Jean Pierre	Géographie humaine
M. RENARD Philippe	Sciences de gestion
M. RICHARD Alain	Biologie animale
M. RIETSCH François	Physique des polymères
M. ROBINET Jean Claude	EUDIL
M. ROGALSKI Marc	Analyse
M. ROLLAND Paul	Composants électroniques et optiques
M. ROLLET Philippe	Sciences Economiques
Mme ROUSSEL Isabelle	Géographie physique
M. ROUSSIGNOL Michel	Modélisation,calcul scientifique,statistiques
M. ROY Jean Claude	Psychophysiologie
M. SALERNO Françis	Sciences de gestion
M. SANCHOLLE Michel	Biologie et physiologie végétales
Mme SANDIG Anna Margarette	
M. SAWERYSYN Jean Pierre	Chimie physique
M. STAROSWIECKI Marcel	Informatique
M. STEEN Jean Pierre	Informatique
Mme STELLMACHER Irène	Astronomie - Météorologie
M. STERBOUL François	Informatique
M. TAILLIEZ Roger	Génie alimentaire
M. TANRE Daniel	Géométrie - Topologie
M. THERY Pierre	Systèmes électroniques
Mme TJOTTA Jacqueline	Mathématiques
M. TOURSEL Bernard	Informatique
M. TREANTON Jean René	Sociologie du travail

M. DERIEUX Jean Claude	Microbiologie
M. DERYCKE Alain	Informatique
M. DESCAMPS Marc	Physique de l'état condensé et cristallographie
M. DEVRAINNE Pierre	Chimie minérale
M. DEWAILLY Jean Michel	Géographie humaine
M. DHAMELINCOURT Paul	Chimie physique
M. DI PERSIO Jean	Physique de l'état condensé et cristallographie
M. DUBAR Claude	Sociologie démographique
M. DUBOIS Henri	Spectroscopie hertzienne
M. DUBOIS Jean Jacques	Géographie
M. DUBUS Jean Paul	Spectrométrie des solides
M. DUPONT Christophe	Vie de la firme
M. DUTHOIT Bruno	Génie civil
Mme DUVAL Anne	Algèbre
Mme EVRARD Micheline	Génie des procédés et réactions chimiques
M. FAKIR Sabah	Algèbre
M. FARVACQUE Jean Louis	Physique de l'état condensé et cristallographie
M. FAUQUEMBERGUE Renaud	Composants électroniques
M. FELIX Yves	Mathématiques
M. FERRIERE Jacky	Tectonique - Géodynamique
M. FISCHER Jean Claude	Chimie organique, minérale et analytique
M. FONTAINE Hubert	Dynamique des cristaux
M. FORSE Michel	Sociologie
M. GADREY Jean	Sciences économiques
M. GAMBLIN André	Géographie urbaine, industrielle et démographie
M. GOBLOT Rémi	Algèbre
M. GOURIEROUX Christian	Probabilités et statistiques
M. GREGORY Pierre	I.A.E.
M. GREMY Jean Paul	Sociologie
M. GREVET Patrice	Sciences Economiques
M. GRIMBLOT Jean	Chimie organique
M. GUELTON Michel	Chimie physique
M. GUICHAOUA André	Sociologie
M. HAIMAN Georges	Modélisation,calcul scientifique, statistiques
M. HOUDART René	Physique atomique
M. HUEBSCHMANN Johannes	Mathématiques
M. HUTTNER Marc	Algèbre
M. ISAERT Noël	Physique de l'état condensé et cristallographie
M. JACOB Gérard	Informatique
M. JACOB Pierre	Probabilités et statistiques
M. JEAN Raymond	Biologie des populations végétales
M. JOFFRE Patrick	Vie de la firme
M. JOURNEL Gérard	Spectroscopie hertzienne
M. KOENIG Gérard	Sciences de gestion
M. KOSTRUBIEC Benjamin	Géographie
M. KREMBEL Jean	Biochimie
Mme KRIFA Hadjila	Sciences Economiques
M. LANGEVIN Michel	Algèbre
M. LASSALLE Bernard	Embryologie et biologie de la différenciation
M. LE MEHAUTE Alain	Modélisation,calcul scientifique,statistiques
M. LEBFEVRE Yannic	Physique atomique,moléculaire et du rayonnement
M. LECLERCQ Lucien	Chimie physique
M. LEFEBVRE Jacques	Physique
M. LEFEBVRE Marc	Composants électroniques et optiques
M. LEFEVRE Christian	Pétrologie
Melle LEGRAND Denise	Algèbre
M. LEGRAND Michel	Astronomie - Météorologie
M. LEGRAND Pierre	Chimie
Mme LEGRAND Solange	Algèbre
Mme LEHMANN Josiane	Analyse
M. LEMAIRE Jean	Spectroscopie hertzienne

M. TURREL Georges	Spectrochimie infrarouge et raman
M. VANDIJK Hendrik	
Mme VAN ISEGHEM Jeanine	Modélisation,calcul scientifique,statistiques
M. VANDORPE Bernard	Chimie minérale
M. VASSEUR Christian	Automatique
M. VASSEUR Jacques	Biologie
Mme VIANO Marie Claude	
M. WACRENIER Jean Marie	Electronique
M. WARTEL Michel	Chimie inorganique
M. WATERLOT Michel	géologie générale
M. WEICHERT Dieter	Génie mécanique
M. WERNER Georges	Informatique théorique
M. WIGNACOURT Jean Pierre	
M. WOZNIAK Michel	Spectrochimie
Mme ZINN JUSTIN Nicole	Algèbre

Merci à Jean-Luc Dekeyser qui a accepté de présider le jury.

Merci à Christian Choffrut et Yves Métivier qui m'ont fait l'honneur de bien vouloir être rapporteurs de cette thèse.

Merci à Mireille Clerbout, Michel Latteux, Yves Roos et Pierre-André Wacrenier pour leur patience et leur aide précieuse, et merci tout particulièrement à Mireille pour avoir dirigé ce travail.

Merci à Alain, François et Rémi,
Merci à Sophie, Christophe(s), Sylvain, Bruno(s), Tof, Gilles, Anne-Cécile, Bernard(s), Lenneke, Philippe(s), Max, Anne, Marc, Eric(s) et Erick, Michel, Stéphane, Jean-Marc, Jean-François...
... tout simplement pour avoir été là (et merci, pour leur pardon, à ceux que j'ai oubliés).

Merci à Anne, Christelle et Marie-Pierre. Parce que!

Merci à Elsa, sans qui ce travail n'aurait sans doute pas vu le jour.

The most exciting phrase to hear in science,
the one that heralds new discoveries,
is not "Eureka!" (I found it!)
but "That's funny ..."

-Isaac Asimov-

Table des matières

Introduction

Pour décrire le comportement des premières générations d'ordinateur, mono-utilisateur, l'outil formel de base fut la théorie des automates qui permet de représenter le fonctionnement séquentiel d'un programme.

L'apparition du parallélisme dans les calculs et les programmes, de réseaux, nécessitant souvent la mise en place de mécanismes complexes de synchronisation ou d'exclusion mutuelle pour l'accès à une ressource, a entrainé le besoin de modèles formels performants pour décrire et étudier les systèmes et les processus parallèles. Le modèle mathématique qui est le plus utilisé (et le mieux adapté) pour la description du parallélisme est celui des réseaux de Pétri, développés dans les années 60-62 par C.A. Pétri, pour modéliser les concepts d'actions asynchrones et concurrentes.

Pour décrire plus facilement le comportement d'un réseau de Pétri, Mazurkiewicz a utilisé un outil très simple : les *traces* et les *langages traces* (voir [25]). De la même façon qu'un automate permet de représenter un programme séquentiel et que le langage associé décrit son comportement (un ensemble de processus séquentiels), une trace permet de représenter le déroulement d'un programme parallèle associé à un réseau de Pétri.

L'étude des langages de traces, qui sont des sous-ensembles du monoïde libre partiellement commutatif, fut introduite pour la première fois par Cartier et Foata en 1969 (voir [4]) pour l'étude des problèmes de réarrangements dans les mots. Tandis qu'un mot, au sens usuel, est une suite de noms d'actions, une trace est définie comme un ensemble partiellement ordonné de noms d'actions. De nombreux et importants résultats ont été trouvés sur ce sujet depuis (voir [1, 3, 5, 15, 16, 23, 26, 28, 27, 29, 32, 36, 41]).

Une première étape dans l'étude des systèmes formels destinés à représenter un système parallèle consiste à associer un nom à chaque action élémentaire et à décrire par une relation binaire sur l'ensemble des noms d'actions le fait que deux actions (ou plus) puissent être exécutées simultanément. Cette relation exprime qu'il est indifférent d'effectuer ces actions dans un ordre précis. Plus précisément, si X est un alphabet, une relation de commutation partielle θ définie sur X est une relation irréflexive et symétrique. Il s'agit d'une relation d'indépendance.

Si X désigne alors l'ensemble des noms (l'alphabet) et θ la relation définie sur X, on associe à (X,θ) une relation d'équivalence notée $\sim$ sur X^* qui sera la plus petite relation vérifiant la propriété suivante :

Si $w = uxyv$, $w' = uyxv$ et $(x,y) \in \theta$, alors w et w' sont équivalents ($x \sim y$).

Une trace se définit alors comme une classe d'équivalence pour cette relation et un langage trace est un ensemble de traces, autrement dit un sous-ensemble du monoïde libre partiellement commutatif correspondant à la structure quotient $X^*_{/\sim}$: les traces sont les classes d'équivalence de la congruence engendrée par la relation θ sur X^*.

Associée à cette relation d'indépendance θ on définit ainsi une application

$$f_\theta : 2^{A^*} \longrightarrow 2^{A^*} .$$

$f_\theta(L)$ est l'ensemble des mots qui sont équivalents à un mot de L pour la congruence engendrée par θ.

Ainsi f_θ est une opération unaire sur les langages qui est appelée *fonction de commutation partielle associée à θ.*

M. Clerbout et M. Latteux ([6] et [11]) ont introduit la notion de semi commutation qui généralise la notion de commutation : une semi commutation est une relation irréflexive d'indépendance sur A. A cause de l'absence de symétrie les semi commutations permettent de modéliser le comportement d'un programme sur une machine à mémoire partagée comme, par exemple, le problème producteur/consommateur : Si p et c désignent respectivement le producteur et le consommateur, la règle non symétrique $cp \rightarrow pc$ signifie que pour tout programme $A = ucpv$, le programme $B = upcv$ a le même résultat.

On peut identifier une relation de semi commutation à un système de réécriture (système de Semi-Thue) où toutes les règles sont de la forme $xy \rightarrow yx$, x et y étant deux lettres distinctes. L'opération qui à un mot associe l'ensemble de tous les mots obtenus par réécritures dans le système est appelée fonction de semi commutation. Ainsi la règle $ba \rightarrow ab$ définit sur l'alphabet $\{a,b\}$ une relation de semi commutation. Alors que les relations de commutation partielle formalisent le parallélisme, les relations de semi commutation expriment le fait que certaines actions doivent être effectuées avant d'autres. Supposons que la lettre a représente l'action *produire* un message dans un buffer, et que la lettre b représente l'action *consommer* un message dans ce même buffer. Si le buffer est de taille 1, le langage représentant le bon déroulement des opérations est $(ab)^*$.

Si on considère maintenant un buffer potentiellement infini, on peut alors produire plus vite qu'on ne consomme mais on ne peut évidemment pas consommer un message qui n'a pas encore été produit. Le langage représentant le bon déroulement des opérations est alors l'union des images des mots du langage $(ab)^*$ par la fonction de semi commutation associée à cette seule règle $ba \rightarrow ab$, c'est-à-dire le langage de Semi-Dyck sur une paire de parenthèses.

Comme les réseaux de Petri, modèle bien connu des programmes concurrents, ne sont que des regroupements de problèmes producteur/consommateur, les semi commutations sont très utiles pour modéliser le comportement des réseaux de Petri. Cette observation a été récemment faite dans [19], [33] et [34]. Elle fut appliquée dans [35].

De nombreux travaux ont été réalisés sur les semi commutations. Nous en donnerons les principaux résultats dans les différents chapitres qui suivent.

Le premier chapitre rappelle des notions de base de la théorie des langages formels et des graphes.

Le second chapitre est consacrée à l'introduction des fonctions de commutation. Il commence par les motivations et les bases théoriques de l'étude des semi commutations. Il se poursuit avec le rappel de définitions et de résultats importants, avec, en particulier, un lemme de projection qui permet de vérifier simplement si un mot donné peut être déduit d'un autre mot donné par une semi commutation donnée. On rappellera également les définitions concernant les commutations partielles et les commutations partitionnées. Le chapitre se terminera par deux résultats, importants et utiles, exposés, le premier par Y. Roos dans sa thèse ([38]), et l'autre par Y. Roos et P.A. Wacrenier dans [39] : il s'agit de la notion de numérotations, et du théorème de composition des semi commutations.

Le troisième chapitre a pour but de présenter un outil dans l'étude des dérivations d'un mot par une semi commutation. Cet outil permettra de simplifier les preuves des chapitres suivants. Il s'agit de la notion de *distance* entre deux mots commutativement équivalents. Cette distance correspond intuitivement au plus petit nombre de dérivations élémentaires que l'on puisse effectuer pour passer d'un mot à l'autre. On aura ainsi un moyen de savoir si une dérivation

$$u \xrightarrow[\theta]{*} w$$

est intéressante dans l'étude d'une dérivation de u en v. On pourra en effet savoir si w nous *rapproche* de v, il suffira de savoir si la distance de w à v est inférieure à celle de u à v.

Il est naturel de se demander si une opération peut être simulée aux moyen d'opérations plus simples : l'exemple typique en est le théorème de Nivat qui prouve que toute transduction rationnelle peut être réalisée par la composition d'un homomorphisme inverse, d'une intersection avec un rationnel, et d'un homomorphisme direct. Dans le quatrième chapitre nous prouverons que toute semi commutation peut être vue comme la composition de semi commutations élémentaires que nous appellerons semi commutations atomiques. Dans ce but, nous définissons les semi commutations atomiques comme étant de la forme $A \times B$ où A et B sont deux sous-alphabets. Nous prouvons ensuite que toute semi commutation peut être décomposée en semi commutations plus petites si et seulement si elle n'est pas atomique. Nous en déduisons enfin que chaque semi commutation peut-être obtenue par une composition de semi commutations atomiques et nous proposons un algorithme effectif de décomposition. Il est intéressant de noter qu'Y. Roos et P.A. Wacrenier ont travaillé sur le sujet inverse : comment déterminer si la composition de deux semi commutations est une semi commutation ? En effet cette question est moins triviale qu'il n'y parait au premier abord : quand on compose deux fonctions de semi commutation, on peut ne pas trouver une fonction de semi commutation. Ils ont trouvé ([39]) une condition nécessaire et suffisante qui utilise les cycles dans les graphes. Ce chapitre se terminera par quelques remarques sur la complexité des différents algorithmes.

Dans le cinquième chapitre nous proposons une nouvelle démonstration pour un résultat que M. Clerbout avait donné dans sa thèse : *toute semi commutation peut être décomposée en homomorphismes non effaçants, homomorphismes inverses et commutations partielles.* Cette nouvelle preuve est plus simple que la preuve originale, et ceci est dû à l'utilisation des décompositions des semi commutations en semi commutations atomiques.

Dans le sixième chapitre nous donnons une caractérisation des semi commutations qui préservent la famille des langages multicompteurs. Dans ce but nous définissons les semi commutations à compteurs comme étant celles qui contiennent

$$\{a\} \times (X \setminus \{a\}) \text{ ou } (X \setminus \{b\}) \times \{b\}$$

à chaque fois qu'elles contiennent un couple (a, b). Nous prouvons alors que les semi commutations à compteurs sont les semi commutations qui préservent les langages multicompteurs. Nous pourrons alors remarquer que certains problèmes dans la famille des langages rationnels, qui étaient indécidables en utilisant des semi commutations quelconques, deviennent décidables en utilisant des semi commutations à compteurs. On pourra également remarquer que ces problèmes restent décidables quand ils concernent la famille des langages multicompteurs, alors que cette dernière est un sur-ensemble de la famille des langages rationnels.

La complexité de l'algorithme de décomposition d'une semi commutation en semi commutations atomiques peut être mesurée par le nombre de semi commutations obtenues. Or, tel qu'il a été défini dans le quatrième chapitre, cet algorithme semble *encombrant*: une étude rapide nous permet facilement d'évaluer le nombre maximum de semi commutations atomiques que l'on peut obtenir pour une semi commutation de n règles comme étant de 2^n. Cependant l'étude des semi commutations à compteurs nous a amenés à nous pencher sur certaines semi commutations atomiques que nous avons appelées semi commutations atomiques maximales. Nous les étudions dans le septième chapitre et leur étude nous permet de prouver que la complexité de l'algorithme est bien exponentielle par rapport au nombre de règles de la semi commutation à décomposer.

Les problèmes de fermeture d'un ensemble donné par une opération donnée sont classiques :

il n'est pas toujours évident de décider si un langage rationnel est fermé par une semi commutation. Dans le dernier chapitre nous proposons un algorithme simple qui permet de calculer, pour un langage rationnel R donné, la plus grande semi commutation θ telle que $f_\theta(R) = R$, et, par la-même, de décider si un rationnel (dont on connaît l'automate réduit déterministe) est fermé par une semi commutation donnée.

Chapitre 1

Définitions et Résultats préliminaires

1.1 Notations

Nous utiliserons dans la suite des notations qui sont, pour la plupart, classiques en mathématiques.

On notera $\mathbb{N}$ l'ensemble des entiers naturels et $\mathbb{N}^+$ l'ensemble des entiers naturels non nuls.

Pour deux entiers a et b tels que $a \leq b$, $[a..b]$ désignera l'intervalle de $\mathbb{N}$:

$$\{a, a+1, a+2, \ldots, b-1, b\} \ .$$

Si $a > b$, $[a..b]$ sera alors l'ensemble vide.

Pour tout ensemble X, Card(X) désignera le cardinal de X dans son sens le plus habituel (c'est-à-dire le nombre des éléments de X), 2^X l'ensemble des parties de X, et Δ_X la diagonale de X :

$$\Delta_X = \{(x, x) \,|\, x \in X\} \ .$$

Pour tout ensemble X et toute partie Y de cet ensemble, $X \backslash Y$ désignera le complémentaire de Y dans X :

$$X \setminus Y = \{x \,|\, x \in X \text{ et } x \notin Y\} \ .$$

Si f est une application d'un ensemble X dans un ensemble Y, et x un élément de X, on notera $f(x)$ l'image de x par f. Par extension, si A est une partie de X, $f(A)$ désignera l'ensemble $\bigcup_{x \in A} f(x)$.

Soit f une application d'un ensemble X dans un ensemble Y. Soit A un sous-ensemble de X. $f_{|_A}$ désignera la restriction de f à A : $f_{|_A}$ est l'application de A dans Y, qui coïncide avec f sur A :

$$\forall x \in A, f_{|_A}(x) = f(x) \ .$$

Nous désignerons l'identité sur X par Id_X :

$$\forall x \in X, \mathrm{Id}_X(x) = x \ .$$

D'une manière plus générale, si aucune confusion n'est possible nous écrirons Id.

1.2 Mots, langages, opérations sur les mots

Nous supposerons connus les résultats classiques de la théorie des langages formels (voir [3]), avec, entre autres, les notions de *monoïde*, *mot*, *langage* et *alphabet*.

Si X est un alphabet, X^* désignera le monoïde libre engendré par l'alphabet X, et ε désignera le mot vide. Si w est un mot de X^*, x et y deux lettres de X, et L un langage de 2^{X^*}, on utilisera les notations suivantes :

- $|w|$ désigne la longueur du mot w.
- $|w|_x$ désigne le nombre d'occurrences de la lettre x dans le mot w.
- $|w|_{x,y}$ désigne le nombre $|w|_x + |w|_y$; cette notation sera étendue de la même manière à un nombre quelconque de lettres, et, pour tout mot u, $|w|_u$ désignera $|w|_{\mathrm{Alph}(u)}$.
- $\mathrm{alph}(w) = \{x \in X \mid |w|_x > 0\}$ est l'*alphabet* du mot w : c'est l'ensemble des lettres apparaissant au moins une fois dans w.
- La *clôture commutative* du mot w sera l'ensemble

$$\mathrm{com}(w) = \{u \in X^* \mid \forall x \in X, |w|_x = |u|_x\} \ .$$

- $\mathrm{FG}(w)$ désignera l'ensemble des facteurs gauches de w :

$$\mathrm{FG}(w) = \{u \in X^* \mid \exists v \in X^*, uv = w\} \ .$$

Définition 1. Deux mots u et v de X^* seront dits *commutativement équivalents* si et seulement si

$$\forall x \in X, |u|_x = |v|_x \ ,$$

ou, plus simplement, si et seulement si

$$\mathrm{com}(u) = \mathrm{com}(v) \ .$$

Définition 2. Pour deux mots w et w' dans X^*, nous noterons $w \sqcup\!\sqcup\, w'$ le *produit de mélange* (ou *shuffle*) de w et w', qui est défini par :

$$w \sqcup\!\sqcup\, w' = \{u_1v_1 \dots u_nv_n \,|\, w = u_1 \dots u_n, w' = v_1 \dots v_n, \forall i \in [1..n], u_i \in X^*, v_i \in X^*\} \ .$$

Définition 3. Un *homomorphisme* f d'un monoïde X^* dans un monoïde Y^* est une application de X^* dans Y^* vérifiant :

- $\forall u, v \in X^*, f(uv) = f(u)f(v)$,
- et $f(\varepsilon) = \varepsilon$.

Un homomorphisme f de X^* dans Y^* sera dit :

- *alphabétique* si $f(X) \subset Y \cup \{\varepsilon\}$,
- *strictement alphabétique* si $f(X) \subset Y$,
- *non effaçant* si $f(X) \subset Y^+$.

Définition 4. Une application $\tau : X^* \to 2^{Y^*}$ est une *transduction rationnelle* s'il existe un alphabet Z, un langage rationnel $R \subset Z^*$ et deux homomorphismes alphabétiques $h : Z^* \to X^*$ et $g : Z^* \to Y^*$ tels que pour tout mot w de X^*, $\tau(w) = g(h^{-1}(w) \cap R)$ (Figure 1.1).

$$\begin{tikzcd} & \cap R \arrow[dl, "h"'] \arrow[dr, "g"] & \\ X^* \arrow[rr, "\tau"] & & 2^{Y^*} \end{tikzcd}$$

Figure 1.1 : Transduction rationnelle

Définition 5. Une famille de langages est un *cône rationnel* si elle est fermée par transduction rationnelle.

Définition 6. Soit un sous-alphabet Y d'un alphabet X. On appelera *projection sur le sous-alphabet* Y l'homomorphisme Π_Y défini par :

$$\forall x \in X, \text{ si } x \in Y \text{ alors } \Pi_Y(x) = x, \text{ sinon } \Pi_Y(x) = \varepsilon \ .$$

On définit également la *projection sur un mot* qui sera la projection sur l'alphabet du mot : si u est un mot de X^*, alors $\Pi_u(w)$ désignera $\Pi_{\mathrm{alph}(u)}(w)$.

Définition 7. Une *substitution* s de X^* dans 2^{Y^*} est une application vérifiant :

- $\forall u, v \in X^*, s(uv) = s(u)s(v)$,
- et $s(\varepsilon) = \varepsilon$.

Une substitution sera dite *alphabétique* si $\forall x \in X, s(x) \subset Y \cup \{\varepsilon\}$.

Le *langage de Dyck* sur l'alphabet $\{x, y\}$ est :

$$D_1^*(x, y) = \{w \mid w \in (x + y)^*, |w|_x = |w|_y\} \ .$$

Le *langage de Semi-Dyck* sur l'alphabet $\{x, y\}$ est :

$$D_1'^*(x, y) = \{w \mid w \in (x + y)^*, |w|_x = |w|_y \text{ et } \forall w' \in \mathrm{FG}(w), |w'|_x \geq |w'|_y\} \ .$$

Il s'agit d'un langage *bien parenthésé* dans lequel x représente la parenthèse ouvrante, y représente la parenthèse fermante.

Quand il ne sera pas nécessaire de préciser l'alphabet, nous noterons D_1^* (respectivement $D_1'^*$) le langage de Dyck (respectivement Semi-Dyck) sur deux lettres.

1.3 Graphes

Nous présentons dans cette section quelques définitions de base de la théorie des graphes.

Un *graphe* G est un couple (S, A) où S est un ensemble fini de *sommets* et A est une partie de $S \times S$ dont les éléments sont appelés *arcs*. Une *arête* est une paire de sommets réunis par au moins un arc. On peut représenter un graphe comme sur la figure 1.2 :

Ici l'ensemble des sommets est

$$\{a, b, c, d, e\} \ ,$$

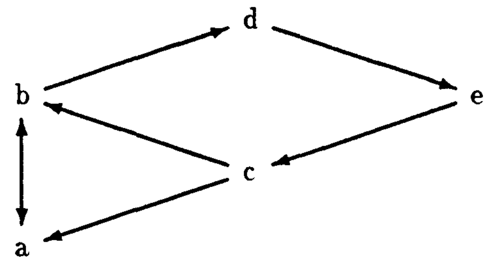


Figure 1.2: Exemple de graphe

l'ensemble des arcs est

$$\{(a,b),(b,a),(b,d),(c,a),(c,b),(d,e),(e,c)\} ,$$

tandis que l'ensemble des arêtes est

$$\{\{a,b\},\{a,c\},\{b,c\},\{b,d\},\{c,e\},\{d,e\}\} .$$

La notion d'arêtes est particulièrement utile quand A est une partie symétrique de $S \times S$, G est alors appelé *graphe non orienté*. On peut représenter un graphe non orienté de deux manières (Figure 1.3).

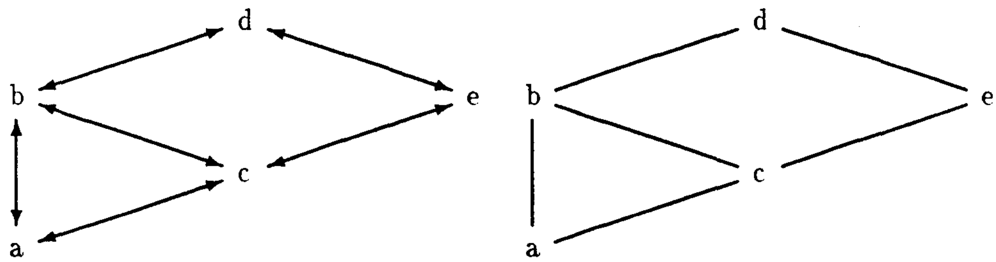


Figure 1.3: Graphe non orienté

Un *chemin* dans un graphe est une suite d'arcs telle que l'extrémité terminale de chaque arc coïncide avec l'extrémité initiale du suivant. Une *chaîne* est une suite d'arêtes telle que chaque arête ait une extrémité commune avec la suivante. La *longueur d'un chemin* est le nombre d'arcs qui le composent.

Par exemple, dans le premier graphe (Figure 1.2), d, e, c et a forment un chemin (et une chaîne), tandis que b, a, c et e forment une chaîne mais pas un chemin.

Un *circuit* est un chemin fini dont le sommet initial et le sommet final coïncident. Si on considère les arêtes d'un graphe, on parlera de *cycle* au lieu de circuit.

Par exemple, dans le premier graphe (Figure 1.2), b, d, e et c forment un circuit.

Un graphe $G = (S, A)$ où pour toute paire de sommets (x, y) il existe au moins une chaîne réunissant x et y est dit *connexe*. Si il existe toujours au moins un chemin de x vers y et au moins un chemin de y vers x, G est dit *fortement connexe*.

Le graphe de la figure 1.4 est fortement connexe.

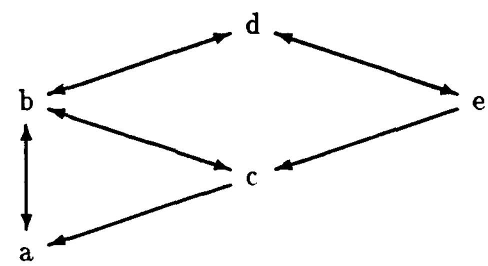


Figure 1.4 : Graphe fortement connexe

Le graphe de la figure 1.5 est connexe, mais pas fortement connexe.

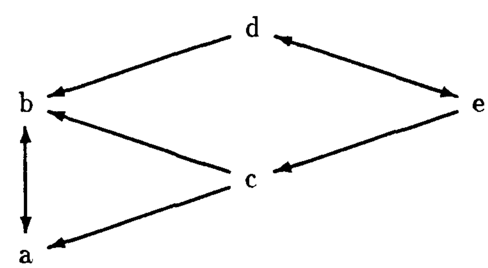


Figure 1.5 : Graphe connexe, mais non fortement connexe

Le dernier graphe n'est pas connexe (Figure 1.6).

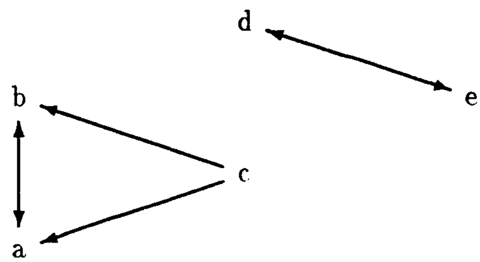


Figure 1.6 : Graphe non connexe

Chapitre 2

Semi commutations

Ce chapitre présente les semi commutations. On y exposera les raisons de l'utilisation des semi commutations, les bases théoriques qui s'y rapportent (fonction de semi commutation, graphe de non-commutation, dérivations...) ainsi que quelques résultats généraux importants qui seront utilisés dans la suite (lemmes de projection, théorèmes de composition).

2.1 Motivations

Les semi commutations sont la modélisation des phénomènes d'indépendance rencontrés dans l'écriture d'un programme : il n'est pas rare de trouver dans un algorithme des événements indépendants :

- Ils peuvent être semi-indépendants : des événements A et B seront semi-indépendants si à chaque fois que l'exécution de l'événement A suivie de l'exécution de l'événement B produit un résultat, on peut assurer que l'exécution de B suivie de celle de A produira le même résultat ; par contre on ne peut pas assurer la réciproque. Il s'agit en général de cas dans lesquels une certaine action doit être exécutée avant une certaine autre. C'est la propriété classique des problèmes *producteur/consommateur* : il est évident qu'un programme qui comporte, par exemple, la séquence :

  ```
  Séquence 1 :
    Prélever dans le buffer B
    Alimenter le buffer B
  ```

  et qui fonctionne sans créer de carence dans le buffer B, fonctionnera tout aussi bien et fournira le même résultat si on remplace la séquence précédente par :

```
Séquence 2 :
  Alimenter le buffer B
  Prélever dans le buffer B
```

Par contre remplacer la séquence 2 par la séquence 1 ne nous garantira évidemment pas un fonctionnement identique.

Il est évident que le fait d'exécuter d'autres actions entre les précédentes peut faire en sorte que ces deux affectations ne soient plus échangeables, mais quand elles sont consécutives l'échange peut être fait.

De tels comportements sont modélisés par des règles anti-symétriques de la forme :

$$cp \longrightarrow pc$$

où p désigne le producteur et c désigne le consommateur. Ainsi, si le buffer est de longueur 1 l'ensemble des programmes représentant un bon fonctionnement de ce buffer sera $(pc)^*$. Mais si le buffer est de longueur potentiellement infinie on peut alors utiliser la règle $cp \longrightarrow pc$ et l'ensemble des programmes représentant un bon fonctionnement de ce buffer sera $D_1'^*(p,c)$, le langage de Semi-Dyck sur d et c.

On parle alors de semi commutation.

- Ils peuvent être aussi totalement indépendants (l'ordre dans lequel ils sont exécutés n'a aucune conséquence sur le résultat de l'algorithme).

  Il peut s'agir, par exemple, de l'affectation ou de la lecture dans deux variables différentes, de l'alimentation de deux buffers distincts, etc...

  Ainsi un programme qui contient la séquence suivante :

```
Séquence 3 :
  Alimenter le buffer B1
  Alimenter le buffer B2
```

fonctionnera tout aussi bien et fournira le même résultat si on remplace la séquence précédente par :

```
Séquence 4 :
  Alimenter le buffer B2
  Alimenter le buffer B1
```

Il doit être aussi entendu que cet échange garantit le même comportement uniquement si aucune autre instruction n'est exécutée entre les deux de la séquence.

De tels comportements sont modélisés par des règles symétriques de la forme :

$$ab \longleftrightarrow ba$$

où a et b désignent les deux actions indépendantes.

Ainsi, il est évident que tout programme représenté par un mot de $a^m \sqcup\!\sqcup\, b^n$ pourra être remplacé indifféremment par un autre mot quelconque de $a^m \sqcup\!\sqcup\, b^n$.

On parle alors de commutation partielle.

Le fait de pouvoir ainsi représenter l'indépendance de deux instructions permet de modéliser un éventuel parallélisme dans un programme, car ceci ne peut être rendu possible qu'en ayant une connaissance complète des dépendances et indépendances entre les instructions.

2.2 Bases théoriques

Définition 8. Une *relation de semi commutation* définie sur l'alphabet X est une relation irréflexive : c'est un sous-ensemble de $X \times X \setminus \Delta_X$.

Dans la suite de ce chapitre, sauf mention explicite d'un autre alphabet, on parlera de semi commutations définies sur l'alphabet X.

(Dans la suite, nous ne considérerons que des relations de semi commutation non vides.)

Définition 9. La *relation de non-commutation* $\bar{\theta}$ associée à θ est la relation complémentaire de θ :

$$\bar{\theta} = X \times X \setminus \theta \ .$$

Définition 10. La *relation de commutation réciproque* correspondant à θ est :

$$\theta^{-1} = \{(a,b) \,|\, (b,a) \in \theta\} \ .$$

Remarque. $\bar{\theta}^{-1}$ désignera indifféremment $\overline{(\theta^{-1})}$ ou $(\bar{\theta})^{-1}$. En effet :

$$\begin{aligned} (x,y) \in \overline{(\theta^{-1})} &\Leftrightarrow (x,y) \notin \theta^{-1} \\ &\Leftrightarrow (y,x) \notin \theta \\ &\Leftrightarrow (y,x) \in \bar{\theta} \\ &\Leftrightarrow (x,y) \in (\bar{\theta})^{-1} \ . \end{aligned}$$

A chaque relation de semi commutation θ sur l'alphabet X est associé un système de réécriture $S = < X, P >$ (système de Semi-Thue) qui est appelé *système de semi commutation* et dans lequel $P = \{xy \to yx \,|\, (x,y) \in \theta\}$.

Nous écrirons $u \xrightarrow[\theta]{} v$ s'il existe une règle $xy \to yx$ dans P et deux mots w et w' tels que $u = wxyw'$ et $v = wyxw'$.

Nous écrirons $u \xrightarrow[\theta]{*} v$ s'il existe des mots $w_1, w_2, \ldots, w_n$ ($n \geq 1$) tels que $w_1 = u$, $w_n = v$, et pour chaque $i \in [1..n-1]$, $w_i \xrightarrow[\theta]{} w_{i+1}$. Nous écrirons alors qu'il y a une *dérivation* de u en v.

Quand nous aurons $u \xrightarrow[\theta]{*} v$ avec une dérivation de longueur l connue, c'est-à-dire s'il existe $l-1$ mots $u_1, u_2, \ldots, u_{l-1}$ tels que

$$u \xrightarrow[\theta]{} u_1 \xrightarrow[\theta]{} u_2 \xrightarrow[\theta]{} \ldots \xrightarrow[\theta]{} u_{l-1} \xrightarrow[\theta]{} v \ ,$$

nous écrirons aussi :

$$u \xrightarrow[\theta]{l} v \ .$$

Définition 11. A chaque relation de semi commutation θ est associée une *fonction de semi commutation* f_θ définie par

$$\forall w \in X^*, f_\theta(w) = \{u \in X^* \mid w \xrightarrow[\theta]{*} u\} \ .$$

Par extension : si L est un langage (sous-ensemble de X^*) alors

$$f_\theta(L) = \bigcup_{w \in L} f_\theta(w) \ .$$

Exemple 1. Ainsi pour la semi commutation $\theta = \{(a,b),(b,c),(c,b)\}$ définie sur l'alphabet $X = \{a,b,c,d\}$:

- P sera égal à :
$$\{ab \to ba, bc \to cb, cb \to bc\} \ .$$
- θ pourra être représentée par son graphe de commutation (Figure 2.1).
- θ^{-1} sera alors représentée par le graphe de la figure 2.2.
- θ pourra aussi être représentée par graphe de non-commutation qui n'est autre que le graphe de $\bar{\theta}$ (Figure 2.3).

  Remarque. Pour des raisons de lisibilité, on ne représentera pas les couples de Δ_X.

- $\bar{\theta}^{-1}$ pourra alors être représentée comme sur la figure 2.4.

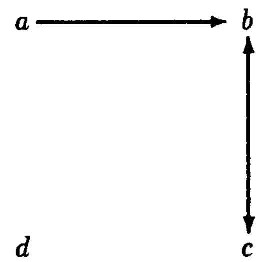


Figure 2.1 : Graphe de commutation

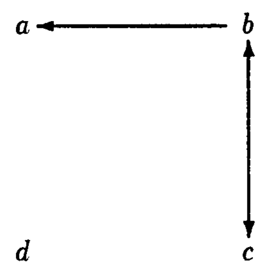


Figure 2.2 : Graphe de θ^{-1}

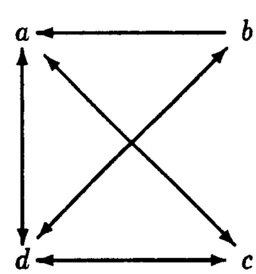


Figure 2.3 : Graphe de non-commutation

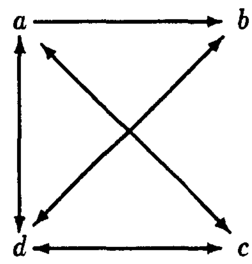


Figure 2.4 : Graphe de $\bar{\theta}^{-1}$

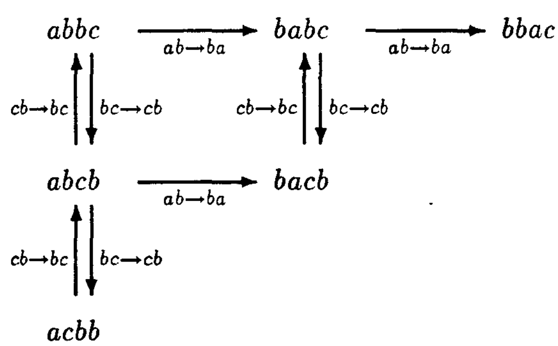


Figure 2.5 : Exemple de dérivation

- Le mot $w = abbc$ pourra être dérivé comme indiqué sur la figure 2.5.
- On aura alors :

$$f(w) = \{abbc, abcb, acbb, babc, bacb, bbac\} \ .$$

Nous écrirons $f_{\theta_1} f_{\theta_2}$ pour la composition de f_{θ_1} et f_{θ_2}. Ainsi :

$$f_{\theta_1} f_{\theta_2}(w) = f_{\theta_1}[f_{\theta_2}(w)] \text{ et } f_{\theta_1} f_{\theta_2}(L) = f_{\theta_1}[f_{\theta_2}(L)] \ .$$

Remarque. Pour toute semi commutation θ, on a évidemment :

$$f_\theta f_\theta = f_\theta \ .$$

Remarque. Si deux semi commutations θ_1 et θ_2 sont telles que $\theta_1 \subset \theta_2$, on aura alors pour tout mot w :

$$f_{\theta_1}(w) \subset f_{\theta_2}(w) \ ,$$

on aura donc :

$$f_{\theta_1} f_{\theta_2}(w) \subset f_{\theta_2}(w) \text{ et } f_{\theta_2} f_{\theta_1}(w) \subset f_{\theta_2}(w) \ .$$

Comme on a évidemment :

$$f_{\theta_2}(w) \subset f_{\theta_1} f_{\theta_2}(w) \text{ et } f_{\theta_2}(w) \subset f_{\theta_2} f_{\theta_1}(w) \ ,$$

on en conclut :

$$f_{\theta_1} f_{\theta_2} = f_{\theta_2} f_{\theta_1} = f_{\theta_2} \ .$$

Définition 12. La fonction de semi commutation f_θ sera dite *décomposable* s'il existe des fonctions de semi commutation $f_{\theta_1}, f_{\theta_2}, \ldots, f_{\theta_n}$ $(n > 1)$ telles que :

- $f = f_{\theta_1} f_{\theta_2} \cdots f_{\theta_n}$,
- et $\forall i \in [1..n], \theta_i \subsetneq \theta$.

Définition 13. La *commutation totale* sur un alphabet X sera la semi commutation

$$\theta = X \times X \setminus \Delta_X \ .$$

Remarque. Si θ est la commutation totale, alors, pour tout mot u de X^*, $f_\theta(u) = \mathrm{com}(u)$.

2.3 Lemme de projection

Le lemme suivant (**Lemme de Projection**) qui fut présenté par M. Clerbout ([6]) permet de décider si un mot appartient à l'image d'un autre mot par une fonction de semi commutation en limitant la vérification à tout sous-alphabet de deux lettres.

Lemme 14 (Lemme de Projection). *Soient w et w' deux mots sur un alphabet X. Alors :*

$$w \xrightarrow[\theta]{*} w' \Leftrightarrow \forall\{x,y\} \subset X, \Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w') \ .$$

Le lemme suivant, qui fut également présenté par M. Clerbout dans [6], est à rapprocher des lemmes de projection. Il énonce un fait intuitivement évident : si une semi commutation sur un alphabet $X = \{a,b\}$ ne contient que la règle $ba \rightarrow ab$, on trouve plus de a à gauche après l'application de cette semi commutation.

Lemme 15. *Sur un alphabet $X = \{a,b\}$, on définit la semi commutation*

$$\Theta = \{(b,a)\} \ .$$

Pour tous mots w et w' de X^*, on aura

$$w \xrightarrow[\Theta]{*} w'$$

si et seulement si

$$w' \in \mathrm{com}(w) \tag{2.1}$$

et

$$\forall w_1 \in \mathrm{FG}(w), \forall w_2 \in \mathrm{FG}(w'), |w_1| = |w_2| \Rightarrow |w_1|_a \leq |w_2|_a \ . \tag{2.2}$$

Ce lemme peut être facilement étendu à un alphabet de plus de deux lettres :

Lemme 16. *Sur un alphabet X tel que $\{a,b\} \subset X$, on définit la semi commutation*

$$\theta = \{(b,a)\} \ .$$

Pour tous mots w et w' de X^*, on aura

$$w \xrightarrow[\theta]{*} w'$$

si et seulement si

$$w' \in \mathrm{com}(w) \tag{2.3}$$

et

$$\forall w_1 \in \mathrm{FG}(w), \forall w_2 \in \mathrm{FG}(w'), |w_1|_{ab} = |w_2|_{ab} \Rightarrow |w_1|_a \leq |w_2|_a \ . \tag{2.4}$$

Preuve. 1. $w \xrightarrow[\theta]{*} w' \Rightarrow (2.3)$?

Evident.

2. $w \xrightarrow[\theta]{*} w' \Rightarrow (2.4)$?

Soient w et w' tels que $w \xrightarrow[\theta]{*} w'$.

Soit alors $w_1 \in \mathrm{FG}(w)$ et $w_2 \in \mathrm{FG}(w')$ tels que $|w_1|_{ab} = |w_2|_{ab}$.

Nous définissons w_o, w'_o, w'_1 et w'_2 tels que

$$w_o = \Pi_{ab}(w)\,, \quad w'_o = \Pi_{ab}(w')\,, \quad w'_1 = \Pi_{ab}(w_1)\,, \quad w'_2 = \Pi_{ab}(w_2) \ .$$

On a bien $w'_1 \in \mathrm{FG}(w_o)$, $w'_2 \in \mathrm{FG}(w'_o)$ et $|w'_1| = |w'_2|$, ce qui donne, en utilisant le **Lemme 15**:

$$|w'_1|_a \leq |w'_2|_a \ .$$

Nous pouvons alors affirmer

$$|w_1|_a \leq |w_2|_a$$

car $|w_1|_a = |w'_1|_a$ et $|w_2|_a = |w'_2|_a$.

3. (2.3) et $(2.4) \Rightarrow w \xrightarrow[\theta]{*} w'$?

Définissons comme précédemment w_o et w'_o tels que

$$w_o = \Pi_{ab}(w)\,, \quad w'_o = \Pi_{ab}(w') \ .$$

Soient $w'_1 \in \mathrm{FG}(w_o)$, $w'_2 \in \mathrm{FG}(w'_o)$ quelconques tels que $|w'_1| = |w'_2|$.

Il existe deux mots $w_1 \in \mathrm{FG}(w)$ et $w_2 \in \mathrm{FG}(w')$ tels que

$$w'_1 = \Pi_{ab}(w_1)\,,\; w'_2 = \Pi_{ab}(w_2) \text{ et } |w_1|_{ab} = |w_2|_{ab} \ .$$

D'après (2.4) on a $|w_1|_a \leq |w_2|_a$.

Comme $|w_1|_a = |w'_1|_a$ et $|w_2|_a = |w'_2|_a$, nous pouvons affirmer que

$$|w'_1|_a \leq |w'_2|_a \ .$$

Comme nous avons évidemment $w_o \in \mathrm{com}(w'_o)$, nous pouvons utiliser le **Lemme 15** pour affirmer que:

$$w_o \xrightarrow[\theta]{*} w'_o, \text{ c'est-à-dire } \Pi_{ab}(w) \xrightarrow[\theta]{*} \Pi_{ab}(w') \ .$$

Or, pour tout autre paire de lettres $\{x, y\}$ différente de $\{a, b\}$ on aura

$$\Pi_{xy}(w) = \Pi_{xy}(w') \ ,$$

ce qui donne :

$$\forall \{x,y\} \subset X, \Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w') \ .$$

En utilisant le **Lemme de Projection** on obtient immédiatement

$$w \xrightarrow[\theta]{*} w' \ .$$

Le lemme est donc démontré. □

2.4 Commutations partielles et commutations partitionnées

Définition 17. Une *commutation partielle* est une semi commutation symétrique.

Le système et la fonction associés sont appelés système de commutation partielle et fonction de commutation partielle.

Les *commutations partitionnées* sont un cas particulier des commutations partielles, donc des semi commutations. Il s'agit en effet de relations dont le complémentaire est une relation d'équivalence.

Si θ est une commutation partitionnée sur un alphabet X, les classes d'équivalence de $\bar{\theta}$ définissent alors une partition de l'alphabet X : $\{X_1, X_2, \ldots, X_k\}$. On a alors la définition équivalente suivante :

Définition 18. Une semi commutation θ est *relation de commutation partitionnée* s'il existe une partition de X, $\{X_1, X_2, \ldots, X_k\}$ telle que :

$$\theta = \{(x,y) \in X \times X \mid \exists i,j \in [1,k], i \neq j, x \in X_i, y \in X_j\} \ .$$

Ou encore :

$$\theta = \bigcup_{i \neq j} X_i \times X_j \ .$$

Nous dirons alors que la semi commutation θ est une relation de commutation partitionnée, associée à la partition $\{X_1, X_2, \ldots, X_k\}$. Le système et la fonction associés sont appelés système et fonction de commutation k-partitionnée.

Enfin, pour finir les rappels sur les fonctions de commutation partitionnée, citons ce résultat de décomposition, dû à M. Clerbout ([6]), qui permet d'exprimer toute fonction de semi commutation (donc non symétrique) au moyen de fonctions de commutation partitionnée (donc symétrique) :

Proposition 19. *Toute fonction de semi commutation peut être obtenue par compositions de homomorphismes, homomorphismes inverses et de fonctions de commutation partitionnée.*

2.5 Numérotations

Il est souvent utile de considérer des mots contenant une seule occurrence de chacune de leurs lettres. Pour pouvoir distinguer entre plusieurs occurrences de la même lettre dans un mot, Y. Roos ([38]) a introduit la définition suivante et il a démontré le résultat important qui suit :

Définition 20. Soit θ une relation de semi commutation qui est définie sur un alphabet X. A chaque entier strictement positif k sont associés :

- l'alphabet $X_k = X \times \{1, 2, \dots, k\}$,
- la relation de semi commutation

$$\theta_k = \{((a,i),(b,j)) \in X_k \times X_k \mid (a,b) \in \theta\} \ ,$$

- l'application $\mathrm{num}_k : X^* \to X_k^*$ qui est définie par :
  - $\mathrm{num}_k(\varepsilon) = \varepsilon$,
  - et $\forall u \in X^*, \forall x \in X, \mathrm{num}_k(ux) = (\mathrm{num}_k(u))(x,p)$,

  avec $p = \inf(k, |ux|_x)$.

Résultat 21. *Soit f_θ une fonction de semi commutation qui est définie sur un alphabet X et u, v deux mots de X^* ; alors*

$$v \in f_\theta(u) \text{ si et seulement si } \mathrm{num}_k(v) \in f_{\theta_k}(\mathrm{num}_k(u)) \ .$$

Dans la suite nous utiliserons de telles *numérotations* pour travailler sur des mots dont chaque lettre n'a qu'une occurrence.

2.6 Lemme d'inclusion

Rappelons quelques résultats qui seront utilisés dans la suite :

Résultat 22. *Si θ est une semi commutation, si Π est une projection, alors*

$$\textit{pour tout mot } u, \ \Pi f_\theta(u) \subset f_\theta \Pi(u) \ .$$

(Voir [38].)

Résultat 23. *Si θ est une semi commutation définie sur A, si g est un homomorphisme strictement alphabétique de B vers A, si θ' est la relation définie sur B par :*

$$\theta' = \{(x,y) \,|\, (g(x), g(y)) \in \theta\} \ ,$$

alors : $f_\theta g = g f_{\theta'}$.

(Voir [12].)

Résultat 24. *Si g est un homomorphisme strictement alphabétique, si A et B sont deux ensembles de mots, alors :*

$$g(A \sqcup B) = g(A) \sqcup g(B) \ .$$

Lemme 25. *Si θ est une semi commutation, si u et v sont deux mots :*

$$f_\theta(u \sqcup v) \subset f_\theta(u) \sqcup f_\theta(v) \ .$$

Preuve. La preuve sera faite en utilisant les résultats précédents. Deux cas peuvent se produire :

- Les mots u et v sont définis sur deux alphabets disjoints A et B :

$$\begin{aligned} f_\theta(u \sqcup v) & \subset \Pi_A f_\theta(u \sqcup v) \sqcup \Pi_B f_\theta(u \sqcup v) \\ & \subset f_\theta \Pi_A(u \sqcup v) \sqcup f_\theta \Pi_B(u \sqcup v) \ (\text{voir } \textbf{Résultat 22}) \\ & = f_\theta(u) \sqcup f_\theta(v) \ . \end{aligned}$$

- Les mots u et v sont définis sur le même alphabet A :

  Soit $\bar{A}$ un alphabet, disjoint de A, contenant les lettres de A barrées :

$$\bar{A} = \{\bar{a} \,|\, a \in A\} \ .$$

  Sur $A \cup \overline{A}$ nous définissons l'homomorphisme g par :

$$\forall a \in A, g(a) = g(\bar{a}) = a \ .$$

  Alors :

$$\begin{aligned}
f_\theta(u \sqcup\!\sqcup\, v) &= f_\theta g(\overline{u} \sqcup\!\sqcup\, v) = g f_{\theta'}(\overline{u} \sqcup\!\sqcup\, v) && \text{(voir } \textbf{Résultat 23}\text{)}\\
&\subset g(f_{\theta'}(\overline{u}) \sqcup\!\sqcup\, f_{\theta'}(v)) && \text{(d'après les lignes précédentes)}\\
&= g f_{\theta'}(\overline{u}) \sqcup\!\sqcup\, g f_{\theta'}(v) && \text{(voir } \textbf{Résultat 24}\text{)}\\
&= f_\theta g(\overline{u}) \sqcup\!\sqcup\, f_\theta g(v) && \text{(voir } \textbf{Résultat 23}\text{)}\\
&= f_\theta(u) \sqcup\!\sqcup\, f_\theta(v) \ .
\end{aligned}$$

On a donc, dans les deux cas,

$$f_\theta(u \sqcup\!\sqcup\, v) \subset f_\theta(u) \sqcup\!\sqcup\, f_\theta(v) \ .$$

□

2.7 Compositions des semi commutations

Souvent la composition de deux semi commutations n'est pas une semi commutation.

Exemple 2. Prenons les deux semi commutations

$$\theta_1 = \{(a,b),(b,c)\} \text{ et } \theta_2 = \{(a,c)\} \ .$$

Si elles sont composables, c'est-à-dire s'il existe une semi commutation θ telle que

$$f_\theta = f_{\theta_1} f_{\theta_2} \ ,$$

cette semi commutation θ ne peut être que

$$\theta = \theta_1 \cup \theta_2 = \{(a,b),(b,c),(a,c)\} \ .$$

En effet, on doit avoir :

- $\theta \subset \theta_1 \cup \theta_2$:
  car toute dérivation effectuée grâce à θ doit pouvoir être faite par θ_1 ou θ_2.
- $\theta_1 \cup \theta_2 \subset \theta$:
  car toute dérivation effectuée grâce à θ_1 ou θ_2 doit pouvoir être faite par θ.

(Ce résultat est d'ailleurs tout à fait général.)

Or, ici, $\theta \neq \theta_1 \cup \theta_2$. Pour nous en convaincre, comparons $f_{\theta_1 \cup \theta_2}(abc)$ et $f_{\theta_1} f_{\theta_2}(abc)$. Il est clair que $cba \in f_{\theta_1 \cup \theta_2}(abc)$. Pourtant $f_{\theta_2}(abc) = \{abc\}$ et $f_{\theta_1} f_{\theta_2}(abc) = \{abc, bac, cab\}$.

Donc

$$f_{\theta_1 \cup \theta_2}(abc) \neq f_{\theta_1} f_{\theta_2}(abc) \ ,$$

donc

$$f_{\theta_1 \cup \theta_2} \neq f_{\theta_1} f_{\theta_2} \ .$$

f_{θ_1} et f_{θ_2} ne sont donc pas composables.

Quand nous aurons besoin de savoir si deux semi commutations se composent en une autre semi commutation nous utiliserons un théorème qui a été démontré par Y. Roos et P.A. Wacrenier dans [39] :

Théorème 26. *Deux semi commutations θ_1 et θ_2 sont composables si et seulement si pour tout sous-alphabet $\{x_o, x_1, \ldots, x_j, x_{j+1}, \ldots, x_n\}$ (avec $n \geq 3$ et $\forall i \neq k, x_i \neq x_k$) tel que :*

- $(x_j, x_{j+1}) \in \bar{\theta}_1 \cap \theta_2$,
- $(x_o, x_n) \in \theta_1 \cap \bar{\theta}_2$,
- $\forall i \in [0..j-1] \cup [j..n-1], (x_i, x_{i+1}) \in \bar{\theta}_1 \cup \bar{\theta}_2^{-1}$,

il existe $k_1 \in [0..j]$ et $k_2 \in [j+1..n]$ tels que $(x_{k_1}, x_{k_2}) \in \bar{\theta}_1 \cap \bar{\theta}_2$.

C'est-à-dire que pour chaque cycle tel que celui de la figure 2.6, on a en fait la figure 2.7.

$$
\begin{tikzcd}
 & x_n & & x_o \arrow[ll, "\theta_1 \cap \bar{\theta}_2"'] \arrow[d] & \\
 & x_{n-1} \arrow[u] & & x_1 & \\
\bar{\theta}_1 \cup \bar{\theta}_2^{-1} & \vdots & & \vdots & \bar{\theta}_1 \cup \bar{\theta}_2^{-1} \\
 & x_{j+2} & & x_{j-1} \arrow[d] & \\
 & x_{j+1} \arrow[u] & & x_j \arrow[ll, "\bar{\theta}_1 \cap \theta_2"] &
\end{tikzcd}
$$

Figure 2.6 : Un tel cycle...

$$
\begin{tikzcd}
 & x_n & & x_o \arrow[ll, "\theta_1 \cap \bar{\theta}_2"'] \arrow[d] & \\
 & x_{n-1} \arrow[u] & & x_1 & \\
\bar{\theta}_1 \cup \bar{\theta}_2^{-1} & \vdots & & \vdots & \bar{\theta}_1 \cup \bar{\theta}_2^{-1} \\
 & x_{k_2} & & x_{k_1} \arrow[ll, "\bar{\theta}_1 \cap \bar{\theta}_2"'] & \\
 & \vdots & & \vdots & \\
 & x_{j+2} & & x_{j-1} \arrow[d] & \\
 & x_{j+1} \arrow[u] & & x_j \arrow[ll, "\bar{\theta}_1 \cap \theta_2"] &
\end{tikzcd}
$$

Figure 2.7 : ... doit contenir $(x_{k_1}, x_{k_2}) \in \bar{\theta}_1 \cap \bar{\theta}_2$

Chapitre 3

Distances

Ce chapitre a pour but de présenter un outil dans l'étude des dérivations d'un mot par une semi commutation. Cet outil permettra de simplifier les preuves des chapitres suivants. Il s'agit de la notion de *distance* entre deux mots commutativement équivalents. Cette distance correspond intuitivement au plus petit nombre de dérivations élémentaires que l'on peut effectuer pour passer d'un mot à l'autre.

On aura ainsi un moyen de savoir si une dérivation $u \xrightarrow[\theta]{*} w$ est intéressante dans l'étude d'une dérivation de u en v. On pourra en effet savoir si w nous *rapproche* de v, il suffira de savoir si la distance de w à v est inférieure à celle de u à v.

Pour cela nous définirons d'abord la notion de *distance*. Ensuite, en utilisant un résultat général sur les transpositions, nous prouverons que la distance entre deux mots est bien le nombre de dérivations élémentaires nécessaires pour passer de l'un à l'autre. Ceci nous permettra de conclure de manière positive quant à l'intérêt de cette notion.

3.1 Distance entre deux mots

Définition 27. Soient u et v deux mots commutativement équivalents. Soit num_k une numérotation telle que toutes les lettres de $\mathrm{num}_k(u)$ sont différentes.

Alors la *distance entre u et v* est :

$$d(u,v) = \mathrm{Card}\{((a,i),(b,j)) \in [\mathrm{alph}(\mathrm{num}_k(u))]^2 \mid \Pi_{(a,i)(b,j)}(\mathrm{num}_k(u)) \neq \Pi_{(a,i)(b,j)}(\mathrm{num}_k(v))\} \; .$$

Il est évident que cette définition ne dépend pas du choix de num_k. (Cela signifie que si

num_k et $\mathrm{num}_{k'}$ sont deux numérotations qui vérifient les hypothèses de la définition alors elles donneront le même résultat pour $d(u,v)$.)

Remarque. Comme dit préalablement, on peut intuitivement comprendre la distance entre deux mots comme la longueur de la plus petite dérivation qu'il peut y avoir de l'un vers l'autre par la commutation totale, ou mieux, que si

$$u \xrightarrow[\theta]{*} v \text{ et } d(u,v) = n \ ,$$

alors n est la plus petite valeur telle que

$$u \xrightarrow[\theta]{n} v \ .$$

Ce résultat sera prouvé plus tard.

Il sera aussi prouvé que le mot de *distance* est utilisé à bon escient.

Nous pouvons aussi utiliser la formulation récursive suivante qui est, d'une manière évidente, équivalente à la précédente:

Définition 28. En posant pour deux mots u et v tels que $|u| = |v| = 2$ et commutativement équivalents:

- $d(u,v) = 0$ si $u = v$,
- $d(u,v) = 1$ si $u \neq v$,

on définit:

$$d(u,v) = \sum_{\substack{(a,i)\in(\mathrm{alph}(\mathrm{num}_k(u)))\\(b,j)\in(\mathrm{alph}(\mathrm{num}_k(u)))}} d(\Pi_{(a,i)(b,j)}(\mathrm{num}_k(u)), \Pi_{(a,i)(b,j)}(\mathrm{num}_k(v))) \ ,$$

Si toutes les lettres sont différentes on peut écrire:

$$d(u,v) = \sum_{\substack{a\in\mathrm{alph}(u)\\b\in\mathrm{alph}(u)}} d(\Pi_{ab}(u), \Pi_{ab}(v)) \ .$$

3.2 Lien entre distances et dérivations

Nous allons commencer par deux lemmes sur les transpositions :

Soit X un alphabet potentiellement infini, et soit

$$U(X^*) = \{w \in X^* \mid \forall x \in X, |w|_x \leq 1\} \ .$$

Pour $v, u \in U(X^*)$ nous écrirons $v \underset{(x,y)}{\longrightarrow} u$ si et seulement si

$$v = w_1xyw_2 \text{ et } u = w_1yxw_2 \ ,$$

où $(x, y) \in X^2$, w_1 et w_2 sont des mots de X^*.

De plus nous écrirons $v \longrightarrow u$ si et seulement si

$$\exists (x, y) \in X^2 \text{ tel que } v \underset{(x,y)}{\longrightarrow} u \ .$$

Soit $w = w_1 \dots w_n \in U(X^*)$, tel que, pour tout i de $[1..n]$, $w_i \in X$, alors nous définissons :

$$E_w = \{(w_i, w_j) \mid 1 \leq i < j \leq n\} \ .$$

Lemme 29. *Soient* $u, v \in U(X^*)$ *tels que* $\mathrm{Alph}(u) = \mathrm{Alph}(v)$.

Soient $u_o, \dots, u_n$ *et* $(x_o, y_o), \dots, (x_{n-1}, y_{n-1})$ *des suites telles que :*

$$u = u_o \ , \ v = u_n \ , \ u_i \underset{(x_i,y_i)}{\longrightarrow} u_{i+1} \text{ et pour tout couple } (i, j),\ (x_i, y_i) \neq (y_j, x_j) \ .$$

Alors $E_u \backslash E_v \subset \{(x_i, y_i) \mid 0 \leq i < n\}$.

Preuve. Le résultat provient directement de

$$E_{u_{i+1}} = (E_{u_i} \backslash \{(x_i, y_i)\}) \cup \{(y_i, x_i)\} \ .$$

La preuve est immédiate. □

Lemme 30. *Soient* $u, v \in U(X^*)$ *tels que* $\mathrm{Alph}(u) = \mathrm{Alph}(v)$.

Alors il existe des suites $u_o, \dots, u_n$ *et* $(x_o, y_o), \dots, (x_{n-1}, y_{n-1})$ *telles que :*

- $u = u_o$ *et* $v = u_n$,

- $(x_i, y_i) \neq (x_k, y_k)$ *pour* $i \neq k$,
- $u_i \underset{(x_i,y_i)}{\longrightarrow} u_{i+1}$,
- $E_u \backslash E_v = \{(x_i, y_i) \mid 0 \leq i < n\}$.

Preuve. Soit f une permutation sur n-élément telle que $v = x_{f(1)} \ldots x_{f(n)}$, où $n = |u|$ et x_k est la k-ième lettre de u. Nous pouvons alors utiliser le fait suivant ([2]):

Résultat 31. *Toute permutation f peut être écrite comme composition de $i(f)$ transpositions standard, où $i(f)$ est le nombre d'inversions de f.*

Il suffit alors de noter:

- que $i(f) = \text{Card}(E_u \backslash E_v)$,
- qu'une transposition standard est une transposition échangeant deux éléments voisins, i.e. qu'elle correspond à une réécriture de la forme $\underset{(x,y)}{\longrightarrow}$ pour $(x, y) \in X^2$,
- et que le passage de u à v est bien une permutation.

Ceci implique que $u \xrightarrow{m} v$, où $m = \text{Card}(E_u \backslash E_v)$. Mais le lemme précédent montre que chaque chaîne de réécriture de u vers v contient toutes les réécritures $\underset{(x_i,y_i)}{\longrightarrow}$ pour $(x_i, y_i) \in E_u \backslash E_v$, d'où le résultat. $\square$

On peut également remarquer que la chaîne de réécriture construite dans le présent lemme est la plus courte possible.

Ainsi nous pouvons présenter le lemme suivant pour les *distances*. C'est la preuve de la remarque préliminaire au sujet de la signification intuitive des *distances*.

Lemme 32. *($u \underset{\theta}{\overset{*}{\longrightarrow}} v$ et $d(u, v) = n$) si et seulement si*

$$u \underset{\theta}{\overset{n}{\longrightarrow}} v \text{ et } \nexists p < n \text{ tel que } u \underset{\theta}{\overset{p}{\longrightarrow}} v \;.$$

Preuve. Evident en utilisant les résultats sur les transpositions: il suffit de remarquer que

$$\text{Card}(E_{\mathbf{num}_k(u)} \backslash E_{\mathbf{num}_k(v)}) = d(u, v)$$

et que l'application d'une semi commutation sur un mot est exactement une permutation sur ce mot. $\square$

Ainsi quand on aura le choix entre deux dérivations il sera possible de choisir la plus courte, quand on aura à choisir le premier mot d'une suite de dérivations, on pourra choisir qui est le plus proche du mot de destination.

3.3 La distance est-elle une distance ?

Nous avons dit, quand nous avons présenté le mot *distance*, qu'il pouvait être utilisé dans son acceptation habituelle en mathématiques. Nous allons maintenant prouver l'exactitude de cette affirmation.

Lemme 33. *d est une distance dans tout ensemble de mots commutativement équivalents.*

Preuve. Vérifions les axiomes des distances pour d:

- $d(u, v) = 0$ si et seulement si $u = v$?
  D'après la **Définition 28**, $d(u, v)$ est nul si et seulement si les projections de u et v sont les mêmes sur chaque couple de lettres, c'est-à-dire si et seulement si $u = v$.
- $d(u, v) = d(v, u)$?
  D'une manière évidente par la symétrie de la définition.
- $d(u, v) + d(v, w) \geq d(u, w)$?
  $d(u, v)$, $d(v, w)$ et $d(u, w)$ étant les longueurs des plus courtes dérivations permettant de passer par la commutation totale respectivement de u à v, de v à w et de u à w, le résultat est évident.

Les axiomes des distances sont vérifiés. d est une distance. □

Il doit bien être entendu que la distance entre deux mots n'a de sens que sur un ensemble de mots commutativement équivalents.

Chapitre 4

Décomposition en semi commutations atomiques

Il est naturel de se demander si une opération peut être simulée aux moyens d'opérations plus simples : l'exemple typique en est le théorème de Nivat ([31]) qui prouve que toute transduction rationnelle peut être réalisée par la composition d'un homomorphisme inverse, d'une intersection avec un rationnel, et d'un homomorphisme direct.

Nous prouverons que toute semi commutation peut être vue comme la composition de semi commutations élémentaires que nous appellerons semi commutations atomiques.

Dans ce but, nous définissons les semi commutations atomiques comme étant de la forme $A \times B$ où A et B sont deux sous-alphabets.

Nous prouvons ensuite que toute semi commutation peut être décomposée en semi commutations plus petites si et seulement si elle n'est pas atomique. Nous en déduisons enfin que chaque semi commutation peut-être obtenue par une composition de semi commutations atomique et nous proposons un algorithme effectif de décomposition.

Il est intéressant de noter qu'Y. Roos et P.A. Wacrenier ont travaillé sur le sujet inverse : comment déterminer si la composition de deux semi commutations est une semi commutation ? En effet cette question est moins triviale qu'il n'y parait au premier abord : quand on compose deux fonctions de semi commutation, on peut ne pas trouver une fonction de semi commutation. Ils ont trouvé ([39]) une condition nécessaire et suffisante qui utilise les cycles dans les graphes. C'est ce résultat qui a été donné dans les préliminaires (**Théorème 26**). En corollaire ils ont prouvé que la composition (ou la décomposition) de deux semi commutations est commutative : si θ, θ_1, θ_2 sont trois semi commutations telles que $f_\theta = f_{\theta_1} f_{\theta_2}$ alors $f_\theta = f_{\theta_2} f_{\theta_1}$.

4.1 Semi commutations atomiques

Nous définissons dans cette section les semi commutations atomiques. Leur nom a été choisi par référence aux *atomes*, tels qu'ils étaient imaginés à l'origine, et qui partagent avec elles la propriété de ne pas être décomposables. Mais contrairement aux atomes originels, pour lesquels il a depuis lontemps été prouvé que l'indécomposabilité n'était qu'une conjecture erronée, nous prouverons que les semi commutations atomiques sont indécomposables, et qu'elles sont les seules semi commutations à avoir cette propriété.

Définition 34. On appelera *semi commutation atomique* toute semi commutation θ pour laquelle il existe deux sous-alphabets A et B tels que:

$$\theta = A \times B \ .$$

Remarque. Si θ est atomique avec $\theta = A \times B$, on aura $A \cap B = \emptyset$, sinon il y aurait au moins un élément commun x à A et B, et (x, x) serait un élément de θ, ce qui est impossible (une semi commutation est une relation irréflexive).

On peut remarquer qu'il existe une autre caractérisation des semi commutations atomiques:

Lemme 35. *Il y a équivalence entre:*

$$\theta \textit{ est une semi commutation atomique} \ , \tag{4.1}$$

et

$$(a, b) \in \theta \textit{ et } (c, d) \in \theta \Rightarrow (a, d) \in \theta \ . \tag{4.2}$$

Preuve. 1. (4.1)⇒(4.2)?

Soit θ une semi commutation atomique: $\theta = A \times B$.

Soit $(a, b) \in \theta$ et $(c, d) \in \theta$; alors $\{a, c\} \subset A$ et $\{b, d\} \subset B$, donc $(a, d) \in \theta$.

2. (4.2)⇒(4.1)?

Soit θ telle que:

$$(a, b) \in \theta \text{ et } (c, d) \in \theta \Rightarrow (a, d) \in \theta \ .$$

Définissons les deux ensembles A et B:

$$A = \{x \in X \mid (x, y) \in \theta\} \text{ et } B = \{y \in X \mid (x, y) \in \theta\} \ .$$

A-t-on $\theta = A \times B$?

(a) $A \times B \subset \theta$?
Soit (a, d) quelconque tel que $(a, d) \in A \times B$. Comme $a \in A, \exists b$ tel que $(a, b) \in \theta$. Comme $d \in B$, $\exists c$ tel que $(c, d) \in \theta$. Donc $(a, d) \in \theta$.

(b) $\theta \subset A \times B$?
Evident d'après la construction de A et B.

On a donc l'implication.

Il y a bien équivalence. □

4.2 Condition suffisante de décomposabilité des semi commutations

Dans cette partie nous allons prouver que toute semi commutation non atomique est décomposable.

En utilisant les lemmes sur les distances nous pourrons presque donner une décomposition effective d'une semi commutation.

Proposition 36. *Soit θ une semi commutation telle que $(b, a) \in \theta$. Si $u \xrightarrow[\theta]{*} v$ alors il existe v' tel que $u \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v$ avec :*

$$\theta' = \{(y, x) \in \theta \,|\, (b, x) \in \theta \text{ et } (y, a) \in \theta\} \text{ et } \theta'' = \theta \setminus \{(b, a)\} .$$

Le fait que θ' ait cette forme s'explique aisément : θ'' contient tous les couples de θ, sauf (a, b). Dans θ' il suffit donc de ne garder que les couples qui pourraient être nécessaires <u>dans</u> ou <u>avant</u> une dérivation qui utilise la règle $ba \to ab$. Nous allons donc garder dans θ' les couples (y, x) de la figure 4.1.

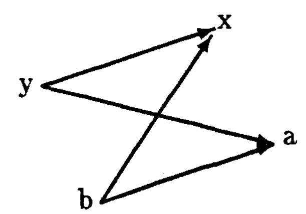


Figure 4.1 : Exemple de couple (y, x) à garder dans θ'

En effet il est assez facile de voir que les dérivations qui doivent être effectuées avant l'utilisation de $ba \to ab$ sont celles pour lesquelles on a, par exemple,

$$byxa \xrightarrow[\theta]{*} xaby \ ,$$

c'est-à-dire les couples (y, x) de θ pour lesquels on a aussi $(b, x) \in \theta$ et $(y, a) \in \theta$.

Exemple 3. Ainsi pour la semi commutation $\theta = \{(b, a), (b, c), (c, a)\}$ (voir figure 4.2), en

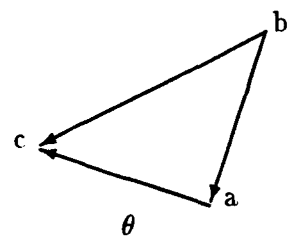


Figure 4.2: Semi commutation à décomposer

prenant $\theta'' = \theta \setminus \{(b, a)\}$, dans θ' on gardera (b, a), mais aussi (b, c) car on a bien $(b, c) \in \theta$ et $(b, a) \in \theta$ (voir figure 4.3). Par contre il ne faut pas garder (a, c) dans θ' car $(b, b) \notin \theta$.

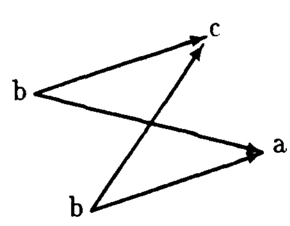


Figure 4.3: Il faut garder (b, c).

On obtient donc la décomposition de la figure 4.4. sera décomposée en θ' et θ'' comme sur la figure 4.4.

Preuve. La preuve sera faite par récurrence sur $d(u, v)$.

- $d(u, v) = 0$

  Alors $u = v$. Le lemme est évidemment vérifié avec $v' = u$.

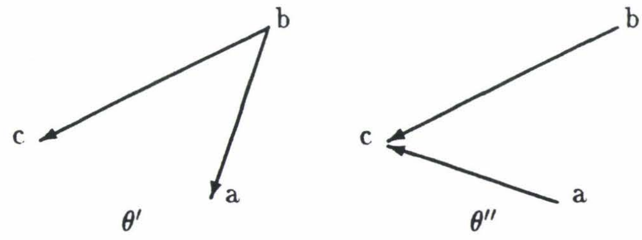


Figure 4.4 : Résultat de la décomposition

- Nous supposons maintenant (hypothèse de récurrence) que le lemme est vérifié pour chaque couple (u, v) tel que $d(u, v) \leq n$:

$$p \leq n, u \xrightarrow[\theta]{p} v \Rightarrow \exists v' \text{ tel que } u \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v \ .$$

- Soient u et v des mots tels que $d(u, v) = n+1$. L'hypothèse est-elle toujours vérifiée ?

  Deux cas peuvent se produire :

  1. $u \xrightarrow[\theta'']{*} v$:
     Le lemme est évidemment vérifié avec $v' = u$.
  2. $u \xrightarrow[\theta]{*} v$ ne peut être fait sans utiliser (b, a) :
     En un tel cas nous pouvons déduire que $u = u_1bu_2au_3$ et $v = v_1av_2bv_3$ avec les mêmes occurrences de a et b, c'est-à-dire

     $$|u_1bu_2|_a = |v_1|_a \text{ et } |u_1|_b = |v_1av_2|_b \ .$$

     Nous noterons : $bu_2a = x_ox_1 \ldots x_q$ où x_o, $x_1, \ldots, x_q \in X$.
     Soit k la plus petite valeur telle que x_k est avant x_o dans v. (k existe car il existe au moins un x_i qui est avant x_o dans v : il s'agit de $x_q = a$.)
     Comme k est la plus petite valeur, nous pouvons en déduire que x_k est avant chaque x_i tels que $i < k$. Ainsi $(x_{k-1}, x_k) \in \theta$, $(x_{k-1}, a) \in \theta$; de plus $(b, x_k) \in \theta$, ainsi, d'après la définition de θ' :

     $$(x_{k-1}, x_k) \in \theta' \ .$$

     Soit u' tel que $u \xrightarrow[\theta']{} u'$ (en utilisant (x_{k-1}, x_k)) alors :

     $$d(u', v) < d(u, v) = n + 1 \ ,$$

donc $d(u',v) \leq n$.
D'où nous pouvons utiliser l'hypothèse de récurrence :

$$\exists v' \text{ tel que } u' \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v \ .$$

D'où : $u \xrightarrow[\theta']{} u' \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v$. Donc : $u \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v$.

L'hypothèse de récurrence est donc vérifiée avec $d(u,v) = n+1$. Elle sera donc vérifiée par chaque valeur de $d(u,v)$.

Ce lemme est donc vérifié dans tous les cas. □

Pour avoir une décomposition effective il nous faut encore voir si nous pourrons toujours trouver (a,b) tel que $\theta' \subsetneq \theta$. Ce sera fait avec les prochains lemmes.

Pour alléger les notations nous définissons θ_{ab} :

Définition 37. Soit a et b deux lettres de X. Soit θ une semi commutation. Alors :

$$\theta_{ab} = \{(y,x) \in \theta \,|\, (b,x) \notin \theta \text{ ou } (y,a) \notin \theta\} \ .$$

(Il faut remarquer que $\theta_{ab} = \theta \setminus \theta'$, ou mieux, que $\theta' = \theta \setminus \theta_{ab}$, en prenant pour θ' l'expression qui avait été donnée dans la **Proposition 36**.)

Lemme 38. *Si θ est une semi commutation qui n'est pas atomique, alors il existe a et b tels que $(b,a) \in \theta$ et $\theta_{ab} \neq \emptyset$.*

Preuve. La preuve sera faite par l'absurde.

Soit une semi commutation θ qui n'est pas atomique. Supposons, en hypothèse, que pour tout couple (b,a) de θ on ait $\theta_{ab} = \emptyset$.

Considérons deux couples (b,a) et (y,x) de θ (il est possible de trouver deux couples distincts car θ n'est pas atomique). θ_{ab} étant vide, (y,x) n'est donc pas dans θ_{ab}. D'après la définition de θ_{ab}, on doit donc avoir $(b,x) \in \theta$ et $(y,a) \in \theta$.

Or d'après le **Lemme 35**, cela signifie que θ est atomique.

On aboutit donc à une contradiction. □

Proposition 39. *Si θ est une semi commutation non atomique alors il existe des lettres a et b telles que f_θ soit décomposable en $f_{\theta''} f_{\theta'}$ avec :*

$$\theta' = \{(y,x) \in \theta \,|\, (b,x) \in \theta \text{ et } (y,a) \in \theta\} \text{ et } \theta'' = \theta \setminus \{(b,a)\} \ .$$

Preuve. Le **Lemme 38** montre que $\theta' \subsetneq \theta$ (puisque $\theta' = \theta \setminus \theta_{ab}$ et que $\theta_{ab} \neq \emptyset$). On a aussi $\theta'' \subsetneq \theta$. Nous avons aussi prouvé :

$$u \xrightarrow[\theta]{*} v \Rightarrow \exists v' \text{ tel que } u \xrightarrow[\theta']{*} v' \xrightarrow[\theta'']{*} v \ .$$

f_θ est donc décomposable en $f_{\theta''} f_{\theta'}$. □

4.3 Condition nécessaire de décomposabilité

Dans cette partie nous allons donner une condition nécessaire pour qu'une semi commutation puisse être décomposable en deux semi commutations strictement plus petites : elle doit être non atomique.

C'est le but de la proposition suivante.

Proposition 40. *Si θ est une semi commutation décomposable alors θ n'est pas atomique.*

Preuve. Soit θ une semi commutation décomposable : $f_\theta = f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}$ avec $\theta_i \subsetneq \theta$ pour tout $i \leq n$.

Supposons θ atomique : $\theta = A \times B$ avec $A \cap B = \emptyset$.

Pour tout i, $\theta_i \subsetneq \theta$ alors $\exists a_i \in A$, $\exists b_i \in B$ tels que $(a_i, b_i) \in \theta \setminus \theta_i$.

Alors soient u et v des mots tels que :

$$u = a_n a_{n-1} a_{n-2} \dots a_3 a_2 a_1 b_1 b_2 b_3 \dots b_{n-2} b_{n-1} b_n \ ,$$

$$v = b_1 b_2 b_3 \dots b_{n-2} b_{n-1} b_n a_n a_{n-1} a_{n-2} \dots a_3 a_2 a_1 \ .$$

Pour tout k, pour tout j, $a_j \in A$, $b_k \in B$, alors $(a_j, b_k) \in \theta$, donc $v \in f_\theta(u)$.

Nous allons maintenant prouver que $v \notin f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}(u)$.

Pour cela nous prouverons, par récurrence, que, pour tout k :

$$f_{\theta_k} f_{\theta_{k-1}} \cdots f_{\theta_2} f_{\theta_1}(u) \subset (a_n \dots a_k b_k \dots b_n) \sqcup\!\sqcup X^* \ .$$

Pour $k = 1$, la propriété est évidemment vérifiée car $f_{\theta_1}(u) = \{u\}$.

Supposons que la propriété soit vérifiée pour k. L'est-elle pour $k+1$?
$f_{\theta_{k+1}} \cdots f_{\theta_2} f_{\theta_1}(u) \subset f_{\theta_{k+1}}((a_n \dots a_k b_k \dots b_n) \sqcup\!\sqcup X^*)$

$$\begin{aligned}
&\subset f_{\theta_{k+1}}(a_n \dots a_k b_k \dots b_n) \sqcup\!\sqcup\, f_{\theta_{k+1}}(X^*) \text{ (par le \textbf{Lemme 25})} \\
&\subset ((a_n \dots a_{k+1} b_{k+1} \dots b_n) \sqcup\!\sqcup \{a_k b_k, b_k a_k\}) \sqcup\!\sqcup\, X^* \\
&\quad \text{(les seules règles utilisables sont } a_i b_i \to b_i a_i \text{ avec } i \geq k) \\
&= (a_n \dots a_{k+1} b_{k+1} \dots b_n) \sqcup\!\sqcup (\{a_k b_k, b_k a_k\} \sqcup\!\sqcup\, X^*) \\
&\quad \text{(associativité de l'opérateur \textit{shuffle})} \\
&= (a_n \dots a_{k+1} b_{k+1} \dots b_n) \sqcup\!\sqcup\, X^*
\end{aligned}$$

La propriété est vérifiée pour $k+1$. Nous pouvons donc en conclure qu'elle est vérifiée pour chaque valeur de k.

Nous pouvons donc dire que, si $w \in f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}(u)$, alors $w \in (a_n b_n) \sqcup\!\sqcup\, X^*$, ce qui n'est évidemment pas vérifié pour v.

D'où $v \notin f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}(u)$, d'où $f_\theta \neq f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}$.

Nous sommes amenés à une contradiction : θ ne peut pas être atomique. □

4.4 Décomposition d'une semi commutation

En rassemblant la **Proposition 39** et la **Proposition 40** on obtient immédiatement :

Théorème 41. *θ est une semi commutation décomposable si et seulement si θ n'est pas atomique.*

4.5 Algorithme de décomposition

Il ne reste plus qu'à fournir un algorithme effectif de décomposition.

Soit θ une semi commutation. Comment la décomposeren semi commutations atomiques ?

- Si θ est atomique :

  θ n'a pas à être décomposée.

- Si θ n'est pas atomique :

  Dans ce cas, par la **Proposition 39**, θ peut être décomposée en deux semi commutations qui sont strictement incluses dans θ.

  Si ces deux semi commutations sont atomiques, le but est atteint.

  Si au moins l'une d'elles n'est pas atomique, l'opération de décomposition est exécutée récursivement à nouveau sur celle(s) qui n'est (ne sont) pas atomique(s).

L'opération est recommencée jusqu'à ce que l'on n'obtienne plus que des semi commutations atomiques.

Cet algorithme se terminera car chaque semi commutation obtenue est strictement plus petite que la précédente (en nombre de couples). D'où il est impossible de ne pas rencontrer une semi commutation atomique car chaque semi commutation qui ne comprend qu'un seul couple est une semi commutation atomique.

Cet algorithme de décomposition, basé sur le **Théorème 41** nous donne immédiatement le corollaire suivant :

Corollaire 42. *Toute semi commutation est décomposable en semi commutations atomiques.*

4.6 Exemple de décomposition

Quand nous dirons dans la suite que θ est décomposable en $\theta'\theta''$ cela signifiera que pour toute dérivation

$$u \xrightarrow[\theta]{*} v \ ,$$

on peut utiliser θ' suivie de θ'', c'est-à-dire qu'il existe un mot w tel que

$$u \xrightarrow[\theta']{*} w \xrightarrow[\theta'']{*} v \ .$$

On aura alors $f_\theta = f_{\theta''} f_{\theta'}$.

Voici un exemple de décomposition que l'on peut obtenir :

Soit $\theta_1 = \{(a,b)\,;(c,a)\,;(b,c)\}$.

- θ_1 peut être décomposée en $\theta_2\theta_3$ avec $\theta_2 = \{(c,a)\,;(b,c)\}$ et $\theta_3 = \{(a,b)\}$.
- θ_3 est atomique et ne peut pas être décomposée.
- θ_2 peut être décomposée en $\theta_4\theta_5$ avec $\theta_4 = \{(b,c)\}$ et $\theta_5 = \{(c,a)\}$.
- θ_4 et θ_5 sont atomiques.

Nous pouvons tracer l'arbre de décomposition (Figure 4.5).

Il suffit de prendre les feuilles de cet arbre pour obtenir la décomposition de θ_1. On obtient : $\theta_1 = \theta_4\theta_5\theta_3$ (donc $f_{\theta_1} = f_{\theta_3} f_{\theta_5} f_{\theta_4}$).

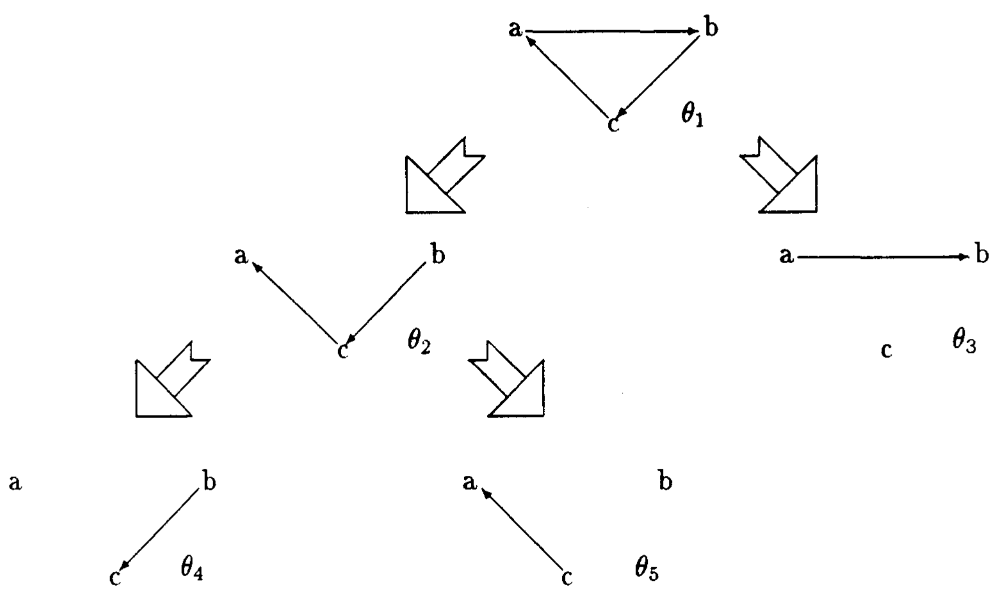


Figure 4.5 : Exemple de décomposition

4.7 Remarque

Notre résultat peut être utilisé pour en démontrer un autre, plus ancien, de M. Clerbout, M. Latteux et Y. Roos ([13]) qui apportait une décomposition équivalente sur les commutations partielles. Il existe en effet une définition équivalente à celle des semi commutations atomiques ; il s'agit des commutations partielles atomiques :

Définition 43. Une commutation partielle θ sera dite *atomique* s'il existe deux sous-alphabets A et B tels que $\theta = (A \times B) \cup (B \times A)$.

C'est la version symétrique de la définition des semi commutations atomiques.

Il avait alors été démontré :

Théorème 44. *Une commutation partielle est décomposable si et seulement si elle n'est pas une commutation partielle atomique. Toute commutation partielle non atomique peut être décomposée en commutations partielles atomiques.*

Preuve. Les commutations partielles atomiques sont définies de façon analogue aux semi commutations atomiques : il est facile de prouver avec une preuve équivalente à celle de la **Proposition 40** que si θ est une commutation partielle décomposable alors θ n'est une commutation partielle atomique.

De plus, si une commutation partielle θ n'est pas une commutation partielle atomique, alors θ n'est pas une semi commutation atomique. Alors, en utilisant notre résultat sur les semi commutations, nous pouvons décomposer :

$$f_\theta = f_{\theta_1} f_{\theta_2} \cdots f_{\theta_n} \quad ,$$

où les θ_i sont des semi commutations atomiques. Pour chaque f_{θ_i} on a

$$f_{\theta_i} \subset f_{s(\theta_i)} \subset f_\theta \quad ,$$

où $s(\theta_i)$ est la version symétrique de θ_i, i.e. $s(\theta_i) = \theta_i \cup \theta_i^{-1}$.

Donc :

$$f_\theta = f_{\theta_1} f_{\theta_2} \cdots f_{\theta_n} \subset f_{s(\theta_1)} f_{s(\theta_2)} \cdots f_{s(\theta_n)} \subset f_\theta f_\theta \cdots f_\theta = f_\theta \quad .$$

Nous n'avons qu'à remarquer que si θ_i est une semi commutation atomique, alors $s(\theta_i)$ est une commutation partielle atomique, et nous avons une décomposition de θ en commutations partielles atomiques :

$$f_\theta = f_{s(\theta_1)} f_{s(\theta_2)} \cdots f_{s(\theta_n)} \quad ,$$

donc si θ n'est pas une commutation partielle atomique alors θ est décomposable en commutations partielles atomiques.

Donc le résultat est prouvé. □

4.8 Atomiques et complexité

Le but de cette partie est de fournir un algorithme effectif (et facilement implémentable) qui décide si une semi commutation est atomique.

Afin de vérifier que cet algorithme est effectivement utilisable, nous étudierons sa complexité.

4.8.1 Structure de données

- `SC` : une liste de couples, représentant la semi commutation, chaque couple étant un arc du graphe de commutation. `P` est la longueur de cette liste.
- `CA` et `CB` : deux compteurs ; `CA` pour les origines d'arcs, `CB` pour les extrémités.
- `A` et `B` : deux tableaux de booléens de taille `N` (`N`=nombre de lettres dans l'alphabet), indicés par les lettres. Ces tableaux marqueront les lettres déjà utilisées, en origine et en extrémité. Ils sont destinés, si la semi commutation θ est atomique, à représenter les deux ensembles A et B tels que $\theta = A \times B$.

4.8.2 Algorithme en pseudo-langage

Nous utiliserons un pseudo-langage, proche du Pascal, pour exprimer cet algorithme.

```
Pour chaque couple de la liste
  Faire
    Soit a son origine,
    Soit b son extrémité,
    Si A[b] ou B[a]
        {c'est-à-dire si a a déjà servi en extrémité
        ou si b a déjà servi en origine}
      Alors
        la semi commutation n'est pas atomique
        EXIT
      Sinon
        Si not(A[a])
            {c'est-à-dire si a n'a pas encore servi en origine}
          Alors
            mettre A[a] à VRAI
            incrémenter CA
        FinSi
```

```
        Si not(B[b])
           {c'est-à-dire si b n'a pas encore servi en extrémité}
          Alors
           mettre B[b] à VRAI
           incrémenter CB
        FinSi
    FinSi
FinPour
Si CA*CB=P
    {car Card(A × B) = Card(A) × Card(B)}
  Alors
    la semi commutation est atomique
  Sinon
    la semi commutation n'est pas atomique
FinSi
```

4.8.3 Complexité en temps

La complexité en temps est clairement linéaire en P, nombre de couples de la semi commutation.

4.8.4 Complexité en espace

La complexité en espace est clairement linéaire en N, taille de l'alphabet.

4.8.5 Complexité en nombre de semi commutations

Une indication de la complexité de cet algorithme pourrait également être le nombre de semi commutations atomiques que l'on peut obtenir en décomposant une semi commutation quelconque.

Contrairement aux précédentes, cette valeur est, quant à elle, exponentielle par rapport au nombre de règles de la semi commutation de départ.

Pour plus de précisions, voir le chapitre 7.

Chapitre 5

Décomposition en homomorphismes et commutations partielles

Dans ce chapitre nous proposons une nouvelle preuve pour un résultat que M. Clerbout avait démontré dans sa thèse : *les semi commutations peuvent être décomposées en homomorphismes non effaçants, homomorphismes inverses et commutations partielles.*

Cette nouvelle preuve est plus simple que la preuve originale, et ceci est dû à l'utilisation des décompositions des semi commutations en semi commutations atomiques. En effet nous nous contenterons de démontrer que cette propriété est vérifiée par les semi commutations atomiques, le cas des semi commutations quelconques devenant alors trivial.

Pour prouver que toute semi commutation peut être décomposée en homomorphismes non effaçants, homomorphismes inverses et commutations partielles, nous aurons besoin d'un certain nombre de résultats préliminaires qui seront donnés dans les lemmes suivants :

- le **Lemme 45**, qui nous apprend que l'intersection avec un rationnel à l'étoile peut être réalisée à l'aide d'homomorphismes et d'homomorphismes inverses ;
- le **Lemme 46**, grâce auquel nous voyons qu'une semi commutation ne possédant qu'une règle et agissant sur un alphabet de trois lettres, peut être simulée sur un langage particulier par des homomorphismes, des homomorphismes inverses et des commutations partielles, la démonstration étant faite en utilisant le **Lemme 45** ;
- le **Lemme 47** qui nous fournit un moyen de construire une semi commutation à partir d'homomorphismes, d'homomorphismes inverses, de commutations partielles, et de commutations partitionnées ; ce lemme provient de [6] ;

- le **Lemme 48**, démontré à partir des précédents, qui nous montre qu'une semi commutation ne possédant qu'une règle peut être simulée par des homomorphismes, des homomorphismes inverses et des commutations partielles.

On est alors très proche du résultat final qui sera établi avec l'aide du **Lemme 49**.

5.1 Notations

Nous écrirons $f \in \mathcal{H}$ quand f est un homomorphisme non effaçant.

Nous écrirons $f \in \mathcal{H}^{-1}$ quand f est un homomorphisme inverse.

Nous écrirons $f \in \mathcal{PC}$ quand f est une fonction de commutation partielle.

Nous écrirons $f \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}$ quand f est un homomorphisme ou un homomorphisme ou une fonction de commutation partielle.

Nous écrirons $f \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ quand f est une composition d'homomorphismes, d'homomorphismes inverses et de fonctions de commutation partielle.

Nous utiliserons des définitions similaires pour $\{\mathcal{H}, \mathcal{H}^{-1}\}$ et $\{\mathcal{H}, \mathcal{H}^{-1}\}^*$.

5.2 Résultats préliminaires

Dans [21] nous trouvons ce lemme utile :

Lemme 45. *Soit R un langage rationnel.*

Soit f définie par $f(L) = L \cap R^*$ pour tout langage L.

Alors $f \in \{\mathcal{H}, \mathcal{H}^{-1}\}^*$.

Le lemme suivant est une extension d'un résultat que M. Clerbout avait donné, ainsi que sa preuve, dans [6].

Lemme 46. *Soit $Z = \{c, d, \#\}$ un ensemble de trois lettres, et soit f la fonction de semi commutation associée à la semi commutation $\{(d, c)\}$.*

Alors il existe $g \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tels que $g_{|(\#+cd)^*} = f_{|(\#+cd)^*}$.

Preuve. Nous allons simuler la semi commutation $\{(d,c)\}$ sur $(\# + cd)^*$ avec des homomorphismes, des homomorphismes inverses, et des commutations partielles.

Nous utiliserons la propriété suivante (voir [22]) :

$$f((cd)^p) = \mathcal{D}'^p_1 = \{w \mid w \in \mathcal{D}'^*_1, |w| = 2p\} = (C_1^* \sqcup\!\sqcup C_1^*) \cap \{c,d\}^{2p} \ ,$$

où

$$C_1 = \{c^n d^n \mid n \geq 0\} \ ,$$

et

$$\mathcal{D}'^*_1 = \{w \mid w \in (c+d)^*, |w|_c = |w|_d, \forall u \in \mathrm{FG}(w), |u|_c \geq |u|_d\} \ .$$

(On peut remarquer que $C_1^* \sqcup\!\sqcup C_1^* = \mathcal{D}'^*_1$.)

D'où :

$$\begin{aligned} f((\# + cd)^p) &= \bigcup_{i_k \geq 0, j_k \geq 0, \sum(i_k + j_k) = p} (\#^{j_1}\mathcal{D}'^{i_1}_1 \#^{j_2}\mathcal{D}'^{i_2}_1 \ldots \#^{j_n}\mathcal{D}'^{i_n}_1) \\ &= \{u_1 u_2 \ldots u_n \mid n \geq 0, u_i \in \#^{j_i}[(c^{l_i}d^{l_i}) \sqcup\!\sqcup (c^{k_i}d^{k_i})], \\ &\qquad \textstyle\sum j_i + \sum l_i + \sum k_i = p, j_i \geq 0, l_i \geq 0, k_i \geq 0\} \end{aligned}$$

Nous allons construire ce langage, à partir de $(\# + cd)^p$, en utilisant seulement des homomorphismes, des homomorphismes inverses et des commutations partielles.

En partant d'un mot de $(\# + cd)^p$ (par exemple $\#^{i_1}(cd)^{j_1} \ldots \#^{i_n}(cd)^{j_n}$) la première idée est de transformer chaque facteur $(cd)^{j_i}$ en $c^{l_i}d^{l_i}c'^{k_i}d'^{k_i}$ avec $l_i + k_i = j_i$.

Soit l'homomorphisme :

$$h \ : \{c, c', d, d', s, s', \#\}^* \longrightarrow \{c, d, \#\}^*$$

défini par

$$h(c) = h(c') = c, \ h(d) = h(d') = h(s) = h(s') = d \text{ et } h(\#) = \# \ .$$

Soit $R_1 = ((cd)^*cs + (c'd')^*c's' + \#)^*$.

Soit $L_1 = R_1 \cap h^{-1}((cd + \#)^p)$. D'après le **Lemme 45** nous savons qu'il existe $m \in \{\mathcal{H}, \mathcal{H}^{-1}\}^*$ tel que $L_1 = m((cd + \#)^p)$.

Alors $L_1 = \{u_1 u_2 \ldots u_n \mid u_i = \#^{j_i}$ ou $u_i = (cd)^{k_i}cs$ ou $u_i = (c'd')^{k_i}c's'$,
$n \geq 1, j_i \geq 0, k_i \geq 0, \sum j_i + \sum(k_i + 1) = p\}$.

Soit f' la fonction de commutation partielle associée à :

$$\theta = \{(c,d), (c',d'), (d,c), (d',c')\} \ .$$

Soit R_2 le langage rationnel $(c^+d^*s + c'^+d'^*s' + \#)^*$. (L'utilisation de s et s' nous assure que les mouvements de c et d transformeront chaque facteur $(cd)^j$ en un facteur c^jd^j, et non en un facteur $c^pd^qc^{p'}d^{q'}$ avec $p \neq q$.)

Soit $L_2 = f'(L_1) \cap R_2$.

D'après le **Lemme 45** nous savons qu'il existe $l \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tels que

$$L_2 = l((cd + \#)^p) \ .$$

Alors $L_2 = \{x_1x_2 \dots x_n \mid x_i = \#^{j_i}$ ou $x_i c^{k_i+1}d^{k_i}s$ ou $x_i = c'^{k_i+1}d'^{k_i}s,$
$n \geq 0, j_i \geq 0, k_i \geq 0, \sum j_i + \sum (k_i + 1) = p\}$.

Soit l'homomorphisme :

$$h_1 \ : \{c, c', d, d', s, s', \#\}^* \longrightarrow \{c, c', d, d', \#\}^*$$

défini par

$$h_1(s) = d, \ h_1(s') = d' \text{ et } \forall x \notin \{s, s'\}, h_1(x) = x \ .$$

Soit $L_3 = h_1(L_2)$.

Alors $L_3 = \{x_1x_2 \dots x_n \mid x_i = \#^{j_i}$ ou $x_i = c^{k_i}d^{k_i}$, ou $x_i = c'^{k_i}d'^{k_i},$
$n \geq 1, j_i \geq 0, k_i \geq 1, \sum j_i + \sum k_i = p\}$.

Soit g la fonction de commutation partielle qui est associée à

$$\{\{c, d\}, \{c', d'\}\} \ ,$$

une partition de $\{c, d, c', d'\}$. La fonction g nous permettra de placer les $c^{k_i}d^{k_i}$ et $c'^{k_i}c'^{k_i}$ entre les blocs de $\#$. Il ne restera alors plus qu'à effacer les marques (c'est-à-dire changer chaque c' en c et chaque d' en d).

Soit l'homomorphisme $k \ : \{c, c', d, d', \#\}^* \longrightarrow \{c, d, \#\}^*$ défini par

$$k(c) = k(c') = c, k(d) = k(d') = d \text{ et } k(\#) = \# \ .$$

Alors

$$k \circ g(L_3) = f((cd + \#)^p) \ .$$

D'où

$$f \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^* \ .$$

Le lemme est démontré. □

Le lemme suivant est donné tel qu'il était dans [6]. Seules les notations ont été modifiées.

Lemme 47. *Soit $P = \{Z_1, Z_2\}$ une partition d'un alphabet Z.*

Soit f_P la fonction de commutation partielle associée à la partition P.

Soit f la fonction de semi commutation associée à une semi commutation

$$\theta \subset Z_2 \times Z_2 \ .$$

Soit f' la fonction de semi commutation associée à la semi commutation

$$\theta' = \theta_P \cup \theta \ .$$

Soit $A \subset Z_2^*$.

S'il existe $g \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tel que

$$f_{|_A} = g_{|_A} \ ,$$

alors nous pouvons trouver $g' \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tel que

$$f'_{|_{A \sqcup\, Z_1^*}} = (f_P \circ g' \circ f_P)_{|_{A \sqcup\, Z_1^*}} \ .$$

Le lemme suivant est basé sur un autre lemme qui avait été prouvé par M. Clerbout.

Lemme 48. *Soit $\{\{a, b\}, Y\}$ une partition d'un ensemble Y_o. Soit F la fonction de semi commutation associée à la semi commutation $\{(b, a)\}$ sur Y_o.*

Alors $F \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$.

Preuve. Dans un mot de Y_o^* les commutations entre a et b seront locales à chaque facteur de $\{a, b\}^*$; dans chacun de ces facteurs, nous allons donc simuler les mouvements possibles des a avec d'autres lettres : nous allons ajouter d qui représentera les emplacements des a dans le mot original w, et c qui représentera les emplacements des a dans les mots de $F(w)$. La lettre $\#$ assurera les mêmes blocages pour c que pour a (a est bloqué par les lettres de Y) :

Soit c, d et $\#$ trois lettres qui n'apparaissent pas dans Y_o.

Soit $Z = Y_o \cup \{c, d, \#\}$.

Soit l'homomorphisme $h : Y_o^* \longrightarrow Z^*$ défini par

$$h(a) = acd, \ h(b) = b \text{ et } \forall x \in Y, h(x) = x\# \ .$$

Soit f la fonction de semi commutation associée à la semi commutation

$$\theta = \{(d,c)\} \ ,$$

définie sur Z.

Soit $Z_1 = Y_o$ et $Z_2 = \{c,d,\#\}$. $P = \{Z_1, Z_2\}$ est une partition de Z.

Soit f_P la fonction de commutation partielle associée à P. (f_P est associée à la commutation partielle θ_P.)

D'après le **Lemme 46**, il existe $g \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tel que

$$g_{|(\#+cd)^*} = f_{|(\#+cd)^*} \ .$$

Soit $A = (\# + cd)^*$. Alors $A \subset Z_2^*$.

Alors, en utilisant le **Lemme 47**, nous pouvons construire $g' \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ tel que

$$(f_P \circ g' \circ f_P)_{|A \sqcup Z_1^*} = f'_{|A \sqcup Z_1^*} \ ,$$

où f' est la fonction de semi commutation associée à $\theta' = \theta_P \cup \theta$.

Soit $w \in Y_o^*$.

Soit $L = f_P \circ g' \circ f_P \circ h(w)$. Comme $h(w) \subset (A \sqcup Z_1^*)$, alors $L = f' \circ h(w)$. D'où

$$L = \{w' \in Z^* \mid \Pi_{Y_o}(w') = w, \ \Pi_{cd\#}(w') \in f((cd + \#)^*)\} \ .$$

(En réalité $\Pi_{cd\#}(w') \in f((cd + \#)^i)$ avec $i = |w|_a + |w|_Y$.)

Soit $R = (ad + b + c + Y\#)^*$.

Soit h_o tel que $h_o(L) = L \cap R$. Alors: $h_o \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ (**Lemme 45**).

Nous devons maintenant effacer chaque $\#$, a et d, puis changer chaque c en a. Pour éviter d'utiliser des morphismes effaçants il est facile le faire avec k qui est la composition de h^{-1} (h a été défini plus haut) et de la fonction de commutation partielle associée à la partition $\{\{a,d\}, \{c,b,\#\} \cup Y\}$: il est évident que $k \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$.

Soit $w_o \in k \circ h_o(L)$. Alors il existe un mot $w' \in h_o(L)$ tel que $w_o = k(w')$.

Comme $w' \in h_o(L)$, c'est le shuffle de deux mots :

- le mot w de l'origine qui peut être vu comme une référence à ce que furent les choses avant les changements,

- un mot de $(c+d+\#)^*$ qui va nous permettre de simuler les déplacements des a.

Les d indiquent les emplacements originaux des a (car ils leur sont collés). Les c ne peuvent se déplacer que vers la gauche et ils sont bloqués par les $\#$; ils indiquent tous les emplacements que les a peuvent atteindre (car les a ne peuvent se déplacer que vers la gauche et ils sont bloqués par les lettres de Y qui sont elles-mêmes collées aux $\#$).

Donc quand nous effaçons les d, a et $\#$ avec k, nous obtenons tous les mots de $F(w)$ (et rien d'autre).

Alors $F \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$. □

Le dernier résultat préliminaire fut donné par M. Clerbout et M. Latteux dans [12].

Lemme 49. *Soit $S = < X, P >$ un système de semi commutation et*

$$\sigma : X^* \longrightarrow 2^{Y^*} ,$$

une substitution alphabétique. Alors l'image de S par σ est $S_\sigma = < Y, P_\sigma >$ où

$$P_\sigma = \{y_1y_2 \longrightarrow y_2y_1 \mid y_1, y_2 \in Y, y_1 \neq y_2, \exists u \longrightarrow v \in P \text{ tel que } y_1y_2 \in \sigma(u)\} .$$

Les fonctions de semi commutation f et f_σ sont respectivement associées à S et S_σ.

Alors les propriétés suivantes sont vérifiées:

- $\forall w \in X^*, \sigma \circ f(w) \subset f_\sigma \circ \sigma(w)$ *et* $\forall w' \in Y^*, f \circ \sigma^{-1}(w') \subset \sigma^{-1} \circ f_\sigma(w')$.
- *De plus, si $\sigma = g^{-1}$, où g est un morphisme strictement alphabétique, alors*

$$\sigma \circ f = f_\sigma \circ \sigma \text{ et } g \circ f_\sigma = f \circ g \quad .$$

5.3 Décomposition des semi commutations atomiques

Nous allons maintenant prouver que toute semi commutation atomique peut être décomposée en homomorphismes, homomorphismes inverses, et commutations partielles.

Proposition 50. *Soit $\{A, B, C\}$ une partition de l'alphabet X.*

Soit f la fonction de semi commutation sur X^* associée à la semi commutation atomique $\theta = B \times A$.

Alors $f \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$.

Preuve. Comme nous utilisons des ensembles finis nous pouvons écrire :

$$A = \{a_1, a_2, \ldots, a_q\},\ B = \{b_1, b_2, \ldots, b_p\} \text{ et } C = \{c_1, c_2, \ldots, c_r\}\ ,$$

où $q = \mathrm{Card}(A)$, $p = \mathrm{Card}(B)$ et $r = \mathrm{Card}(C)$.

Soient a et b deux lettres telles que $X \cap \{a, b\} = \emptyset$.

Soit l'homomorphisme $h : X^* \longrightarrow (X \cup \{a, b\})^*$ tel que :

$$\begin{aligned} h(a_i) &= a_i a \text{ pour tout } i \text{ de } [1..q], \\ h(b_i) &= b_i b \text{ pour tout } i \text{ de } [1..p], \\ h(c_i) &= c_i \text{ pour tout } i \text{ de } [1..r]. \end{aligned}$$

Soit f' la fonction de commutation partielle associée à la commutation partitionnée θ' qui est elle-même associée à la partition $\{A, B, C \cup \{a, b\}\}$ de $X \cup \{a, b\}$.

Soit F la fonction de semi commutation qui est associée à la semi commutation

$$\Theta = \{(b, a)\} \text{ sur } (X \cup \{a, b\})^*\ .$$

Alors $F \in \{\mathcal{H}, \mathcal{H}^{-1}, \mathcal{PC}\}^*$ (voir **Lemme 48**).

Comme toutes les lettres de A ont le même comportement par rapport aux lettres de B, l'idée est de *simuler* les déplacements des lettres de A par rapport à celles de B sur une copie : $\{a, b\}$.

Soit $w \in X^*$. Nous allons démontrer que $f(w) = h^{-1} \circ f' \circ F \circ f' \circ h(w)$.

1. $f(w) \subset h^{-1} \circ f' \circ F \circ f' \circ h(w)$?

   Soit $w' \in f(w)$.

   h_o est le morphisme défini par :

$$\begin{aligned} h_o(a_i) &= a \text{ pour tout } i \text{ de } [1..q], \\ h_o(b_i) &= b \text{ pour tout } i \text{ de } [1..p], \\ h_o(c_i) &= c_i \text{ pour tout } i \text{ de } [1..r]. \end{aligned}$$

   Soit $w_o = h(w)$.

Alors : $w_o \xrightarrow[\theta']{*} \Pi_A(w_o)\Pi_B(w_o)\Pi_{C\cup\{a,b\}}(w_o) = \Pi_A(w)\Pi_B(w)h_o(w)$

$\xrightarrow[\Theta]{*} \Pi_A(w')\Pi_B(w')h_o(w')$

En effet :

- $\Pi_A(w)\Pi_B(w) = \Pi_A(w')\Pi_B(w')$,
- en utilisant la dernière partie du **Lemme 49** en remplaçant respectivement f_σ, f et g par f, F et h_o, nous avons

$$h_o(w') \in h_o \circ f(w) = F \circ h_o(w) \ ,$$

  donc $h_o(w) \xrightarrow[\Theta]{*} h_o(w')$.

$= \Pi_A(h(w'))\Pi_B(h(w'))\Pi_{C\cup\{a,b\}}(h(w'))$
$\xrightarrow[\theta']{*} h(w')$
$\xrightarrow{h^{-1}} w'$

Donc : $w' \in h^{-1} \circ f' \circ F \circ f' \circ h(w)$.
Donc : $f(w) \subset h^{-1} \circ f' \circ F \circ f' \circ h(w)$.

2. $h^{-1} \circ f' \circ F \circ f' \circ h(w) \subset f(w)$?
Soit w' tel que :

$$w \xrightarrow{h} w_o \xrightarrow{f'} w_1 \xrightarrow{F} w_2 \xrightarrow{f'} w_3 \xrightarrow{h^{-1}} w' \ .$$

Nous allons prouver que $w' \in f(w)$ avec le **Lemme de Projection** : nous prouverons que $\Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w')$ pour chaque x et y de X.

- Si $(\{x,y\} \subset A)$ ou $(\{x,y\} \subset B)$ ou $(\{x,y\} \subset C)$:
Alors $\Pi_{xy}(w) = \Pi_{xy}(w')$ car

$$\Pi_A(w) = \Pi_A(w'),\ \Pi_B(w) = \Pi_B(w') \text{ et } \Pi_C(w) = \Pi_C(w') \ .$$

Donc $\Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w')$.
- Si $(x,y) \in A \times B$:
On a :

$$w \xrightarrow{h} w_o \xrightarrow{f'} w_1 \xrightarrow{F} w_2 \xrightarrow{f'} w_3 \xrightarrow{h^{-1}} w' \ ,$$

et nous allons prouver :

$$\Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w') \ .$$

Nous définissons $F_{ab}(w_o)$:

$$F_{ab}(w_o) = \{u'x \in \mathrm{FG}(w_o) \mid u' \in (X \cup \{a,b\})^*, x \in \{a,b\} \cup C\} \ .$$

$F_{ab}(w_o)$ est l'ensemble des facteurs gauches de w_o ne se terminant pas par un élément de $A \cup B$. Cet ensemble va nous permettre de mettre en relation des facteurs gauches de w_o et les facteurs gauches de w : il est en effet évident qu'il existe une bijection entre $F_{ab}(w_o)$ et FG(w), et que cette bijection est réalisée par h :

$$h(\mathrm{FG}(w)) = F_{ab}(w_o) \text{ et } \forall u \in F_{ab}(w_o), \exists! v \in \mathrm{FG}(w) \text{ tel que } h(v) = u \ .$$

On définit de la même façon $F_{ab}(w_3)$, en bijection avec FG(w') par h :

$$h(\mathrm{FG}(w')) = F_{ab}(w_3) \text{ et } \forall u \in F_{ab}(w_3), \exists! v \in \mathrm{FG}(w') \text{ tel que } h(v) = u \ .$$

Il nous faut faire quelques remarques quant aux projections des différents mots w, w_o, w_1, w_2, w_3, w' :

- Comme on a :
  * $\Pi_{a+C}(w_o) = \Pi_{a+C}(w_1) = \Pi_{a+C}(w_2) = \Pi_{a+C}(w_3)$,
  * $\Pi_{a+C}(w_o) = h_o(\Pi_{A+C}(w_o))$,
  * $\Pi_{A+C}(w_o) = \Pi_{A+C}(w)$,
  * $\Pi_{a+C}(w_3) = h_o(\Pi_{A+C}(w_3))$,
  * $\Pi_{A+C}(w_3) = \Pi_{A+C}(w')$,

  on en arrive à

$$\Pi_{A+C}(w) = \Pi_{A+C}(w_o) = \Pi_{A+C}(w_3) = \Pi_{A+C}(w') \ .$$

- De la même façon on prouve que :

$$\Pi_{B+C}(w) = \Pi_{B+C}(w_o) = \Pi_{B+C}(w_3) = \Pi_{B+C}(w') \ .$$

Prenons alors deux mots quelconques U et V tels que :

- $U \in \mathrm{FG}(\Pi_{xy}(w))$,
- $V \in \mathrm{FG}(\Pi_{xy}(w'))$,
- $|U|_{xy} = |V|_{xy}$.

D'après les remarques ci-dessus, on peut trouver deux mots u et v tels que :

- $u \in \mathrm{FG}(w)$ et $\Pi_{xy}(u) = U$,
- $v \in \mathrm{FG}(w')$ et $\Pi_{xy}(v) = V$,
- $|u|_{A \cup B} = |v|_{A \cup B}$.

D'après les remarques faites au sujet de F_{ab}, on peut trouver u' et v' tels que :

- $u' \in F_{ab}(w_o)$ et $|u'|_{A \cup B} = |u|_{A \cup B}$,
- $v' \in F_{ab}(w_3)$ et $|v'|_{A \cup B} = |v|_{A \cup B}$.

On a alors $|u'|_{ab} = |v'|_{ab}$, puisque

$$|u'|_{ab} = |u'|_{A \cup B} = |u|_{A \cup B} = |v|_{A \cup B} = |v'|_{A \cup B} = |v'|_{ab} \ .$$

Comme f' ne modifie pas les projections sur ab il existe

$$u'' \in \mathrm{FG}(w_1) \text{ et } v'' \in \mathrm{FG}(w_2)$$

tels que :

$$\Pi_{ab}(u'') = \Pi_{ab}(u') \text{ et } \Pi_{ab}(v'') = \Pi_{ab}(v') \ . \tag{5.1}$$

On aura alors évidemment $|u''|_{ab} = |v''|_{ab}$.
D'après le **Lemme 16**, nous pouvons affirmer que $|u''|_a \leq |v''|_a$.
On en arrive alors à

$$|u'|_a \leq |v'|_a \ ,$$

grâce à (5.1), ce qui donne

$$|u'|_A \leq |v'|_A \ ,$$

car $u' \in F_{ab}(w_o)$ et $v' \in F_{ab}(w_3)$, puis

$$|u|_A \leq |v|_A \ ,$$

à cause des bijections existant entre $F_{ab}(w_o)$ et $\mathrm{FG}(w)$, ainsi qu'entre $F_{ab}(w_3)$ et $\mathrm{FG}(w')$.
On a enfin, puisque $\Pi_A(w) = \Pi_A(w')$,

$$|u|_x \leq |v|_x \ .$$

On a donc

$$|\Pi_{xy}(u)|_x \leq |\Pi_{xy}(v)|_x \ .$$

Les mots U et V étant quelconques, nous pouvons alors conclure à l'aide du **Lemme 16**, que

$$\Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w') \ .$$

- Si $(x,y) \in B \times A$:
  La démonstration est alors évidemment identique à la précédente.
- Si $(x,y) \in A \times C$:
  Nous avons :
  - $\Pi_A(w) = \Pi_A(w') = \Pi_A(w_o) = \Pi_A(w_1) = \Pi_A(w_2) = \Pi_A(w_3)$,
  - $\Pi_{a+C}(w_o) = \Pi_{a+C}(w_1) = \Pi_{a+C}(w_2) = \Pi_{a+C}(w_3)$.

  De plus : $\Pi_{A\cup C}(w) = \Pi_{A\cup C}(w_o) = \alpha_1\gamma_1 \ldots \alpha_n\gamma_n$ avec, pour $i \in [1..n]$,

$$\alpha_i \in A^* \text{ et } \gamma_i \in C^* \ .$$

  Nous avons aussi, pour $i \in [0..2]$:

$$\Pi_{a+C}(w_i) = a^{A_1}\gamma_1' \ldots a^{A_s}\gamma_s'$$

où $\{A_j\}_{j\in[1..s]} \subset \mathbb{N}$.
Parce que

$$\Pi_{A\cup C}(w_o) = \Pi_{a+C}(h(\Pi_{A\cup C}(w_o))) = \Pi_{a+C}(w_o) \ ,$$

nous devons avoir $s = n$, et, pour chaque $i \leq n$, $A_i = |\alpha_i|$ et $\gamma_i = \gamma_i'$.
Donc : $\Pi_{a+C}(w_i) = a^{|\alpha_1|}\gamma_1 \dots a^{|\alpha_n|}\gamma_n$.
De la même façon nous prouvons :

$$\Pi_{A\cup C}(w_3) = \Pi_{A\cup C}(w') = \alpha_1'\gamma_1'' \dots \alpha_t'\gamma_t''$$

et

$$\Pi_{a+C}(w_i) = a^{|\alpha_1'|}\gamma_1'' \dots a^{|\alpha_t'|}\gamma_t'' \ .$$

Avec ceci nous pouvons dire que $t = n$, $\gamma_i'' = \gamma_i$ pour chaque $i \leq n$, et $|\alpha_i| = |\alpha_i'|$ pour chaque $i \leq n$.
De plus $\Pi_A(w) = \alpha_1 \dots \alpha_n = \alpha_1' \dots \alpha_n'$ aussi $\alpha_i = \alpha_i'$ pour chaque $i \leq n$.
Donc $\Pi_{A\cup C}(w) = \Pi_{A\cup C}(w')$, donc :

$$\forall x \in A, \forall y \in C, \Pi_{xy}(w) = \Pi_{xy}(w') \ ,$$

donc :

$$\forall x \in A, \forall y \in C, \Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w') \ .$$

- Si $((x,y) \in C \times A)$ ou $((x,y) \in B \times C)$ ou $((x,y) \in C \times B)$:
  La démonstration est alors évidemment identique à la précédente.

Ainsi $\Pi_{xy}(w) \xrightarrow[\theta]{*} \Pi_{xy}(w')$ pour chaque x et y de X.

D'après le **Lemme de Projection** nous pouvons dire $w' \in f(w)$.

Donc $h^{-1} \circ f' \circ F \circ f' \circ h(w) \subset f(w)$.

Donc $f = h^{-1} \circ f' \circ F \circ f' \circ h$. □

5.4 Décomposition de semi commutations

Nous avons prouvé que chaque semi commutation peut être décomposée en semi commutations atomiques (**Proposition 42**). Nous avons prouvé dans la partie précédente que chaque semi commutation atomique peut être décomposée en homomorphismes non effaçants, homomorphismes inverses, et commutations partielles.

Nous obtenons donc immédiatement le dernier résultat :

Théorème 51. *Toute semi commutation est décomposable en homomorphismes non effaçants, homomorphismes inverses, et commutations partielles.*

Chapitre 6

Langages multicompteurs et semi commutations à compteurs

Dans ce chapitre nous donnons une caractérisation des semi commutations qui préservent la famille des langages multicompteurs.

Dans ce but nous définissons les semi commutations à compteurs comme étant celles qui contiennent $\{a\} \times (X \setminus \{a\})$ ou $(X \setminus \{b\}) \times \{b\}$ à chaque fois qu'elles contiennent un couple (a, b). Nous prouvons alors que les semi commutations à compteurs sont les semi commutations qui préservent les langages multicompteurs.

Nous pourrons alors remarquer que certains problèmes qui étaient indécidables dans la famille des langages rationnels en utilisant des semi commutations quelconques, deviennent décidables en utilisant des semi commutations à compteurs.

On pourra également remarquer que ces problèmes, toujours en utilisant des semi commutations à compteurs, restent décidables quand ils concernent la famille des langages multicompteurs, sur-ensemble de la famille des langages rationnels.

6.1 Langages multi-compteurs

$\mathcal{MC}$ est la famille des *langages multicompteurs*: c'est la plus petite famille fermée par intersection et transduction rationnelle qui contient $D_1'^*$, langage de Semi-Dyck. (Ils sont reconnus par des automates à compteurs, pas nécessairement déterministes, dans lesquels le vide des compteurs ne peut pas être testé, c'est-à-dire qu'il n'y a pas de symbole de fond de compteur.)

Exemple 4. Tout langage rationnel est dans $\mathcal{MC}$ car tout rationnel est reconnu par un automate d'états finis, qui n'est qu'un automate à compteurs particulier.

Exemple 5. $D_1^* \in \mathcal{MC}$:

Nous donnons l'automate à compteurs qui reconnait ce langage :

- quand nous écrirons $b/+\beta$, cela devra être compris comme une transition de cet automate : un 'b' est lu, et le compteur β est incrémenté $(+)$,
- quand nous écrivons $a/-\alpha$, un 'a' est lu, et le compteur α est décrémenté $(-)$,
- un mot sera reconnu par l'automate s'il permet d'atteindre un état final avec les compteurs remis à zéro.

Avec ces conventions D_1^* est reconnu par l'automate à compteurs de la figure 6.1 (automate à deux compteurs α et β).

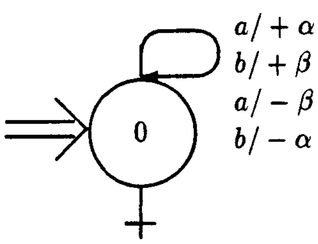


Figure 6.1 : Automate à compteurs reconnaisssant D_1^*

Exemple 6. $\{a^n b^n c^n \mid n \geq 1\} \in \mathcal{MC}$:

$\{a^n b^n c^n \mid n \geq 1\}$ est reconnu par l'automate à compteurs de la figure 6.2 (automate à deux compteurs α et β).

Exemple 7. $\{a^n b^n \mid n \geq 0\}^* \notin \mathcal{MC}$:

Intuitivement on peut voir qu'il faudrait un nombre infini de compteurs, car on ne peut pas tester le vide d'un compteur.

Plus précisement, le plus petit cône rationnel fermé par intersection qui contient

$$\{a^n b^n \mid n \geq 0\}^*$$

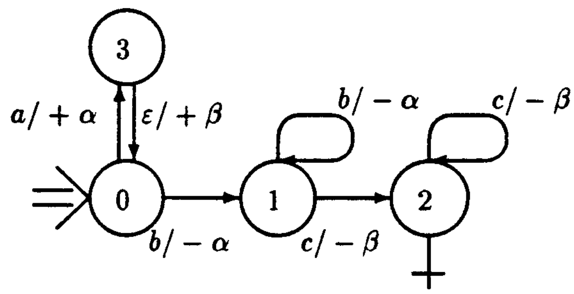


Figure 6.2 : Automate à compteurs reconnaissant $\{a^n b^n c^n \mid n \geq 1\}$

est l'ensemble des langages récursivement énumérables ([30]), et non $\mathcal{MC}$, alors que $\mathcal{MC}$ est lui-même un cône rationnel fermé par intersection ([20]) et que $\mathcal{MC}$ est strictement inclus dans cet ensemble.

Donc $\{a^n b^n \mid n \geq 0\}^* \notin \mathcal{MC}$.

6.2 Semi commutations à compteurs

Nous proposons une nouvelle sous-famille de semi commutations :

Définition 52. Une *semi commutation à compteurs* est une semi commutation θ telle que pour tout $(a,b) \in \theta$, on ait $\{a\} \times (X - \{a\}) \subset \theta$ ou $(X - \{b\}) \times \{b\} \subset \theta$.

Cette caractéristisation se conservant par union, on a immédiatement le lemme suivant :

Lemme 53. *La famille des semi commutations à compteurs est fermée par union.*

On peut également donner la caractérisation suivante :

Lemme 54. *Une semi commutation est une semi commutation à compteurs si et seulement si :*

$$\textit{pour tous } a \neq b \textit{ et tous } c \neq d, \textit{ si } (a,b) \in \bar{\theta} \textit{ et } (c,d) \in \bar{\theta} \textit{ alors } (a,d) \in \bar{\theta}\,.$$

Preuve. Immédiat par une simple réécriture de la définition.

□

Remarque. On aura dans ce cas bien sûr également $(c,b) \in \bar{\theta}$.

Exemple 8. Le graphe de la figure 6.3 définit :

$$\theta = (\{a\} \times (X\backslash\{a\})) \cup (\{b\} \times (X\backslash\{b\})) \cup ((X\backslash\{b\}) \times \{b\}) \ ,$$

avec $X = \{a, b, c, d\}$.

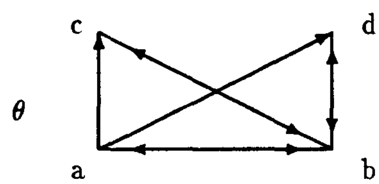


Figure 6.3 : Graphe de commutation d'une semi commutation à compteurs

6.3 Composition de semi commutations à compteurs

Théorème 55. *Deux semi commutations à compteurs quelconques agissant sur le même alphabet sont toujours composables, et leur composition est une semi commutation à compteurs.*

Preuve. 1. Toujours composables ?

Nous utiliserons le **Théorème 26** ; soit un cycle (voir figure 6.4).

Dans la première partie (droite) de ce cycle $(x_n, x_o, \ldots, x_j, x_{j+1})$ il existe x_{k_1} tel que :

$$(x_{k_1-1}, x_{k_1}) \in \bar{\theta}_2^{-1} \text{ et } (x_{k_1}, x_{k_1+1}) \in \bar{\theta}_1 \ .$$

Dans la deuxième partie (gauche) de ce cycle $(x_j, x_{j+1}, \ldots, x_n, x_o)$ il existe x_{k_2} tel que :

$$(x_{k_2-1}, x_{k_2}) \in \bar{\theta}_1 \text{ et } (x_{k_2}, x_{k_2+1}) \in \bar{\theta}_2^{-1} \ .$$

Donc on obtient $(x_{k_1}, x_{k_1+1}) \in \bar{\theta}_1$ et $(x_{k_2-1}, x_{k_2}) \in \bar{\theta}_1$.

D'après le **Lemme 54** nous pouvons dire que :

$$(x_{k_1}, x_{k_2}) \in \bar{\theta}_1 \ .$$

$$\begin{tikzcd}
 & x_n & x_o \arrow[l, "\theta_1 \cap \bar{\theta}_2"'] \arrow[d] & \\
 & x_{n-1} \arrow[u] & x_1 & \\
\bar{\theta}_1 \cup \bar{\theta}_2^{-1} & \vdots & \vdots & \bar{\theta}_1 \cup \bar{\theta}_2^{-1} \\
 & x_{j+2} & x_{j-1} \arrow[d] & \\
 & x_{j+1} \arrow[u] & x_j \arrow[l, "\bar{\theta}_1 \cap \theta_2"] &
\end{tikzcd}$$

Figure 6.4: Cycle de composabilité

Nous avons aussi $(x_{k_1}, x_{k_1-1}) \in \bar{\theta}_2$ et $(x_{k_2+1}, x_{k_2}) \in \bar{\theta}_2$.

De nouveau d'après le **Lemme 54** nous pouvons dire également que:

$$(x_{k_1}, x_{k_2}) \in \bar{\theta}_2 \ .$$

D'où $(x_{k_1}, x_{k_2}) \in \bar{\theta}_1 \cap \bar{\theta}_2$.

Nous en concluons que θ_1 and θ_2 vérifient les conditions du **Théorème 26**, et que θ_1 et θ_2 sont composables.

2. Leur composition est-elle une semi commutation à compteurs?

   Evident: si deux semi commutations sont composables alors leur composition est leur union, et la famille des semi commutations à compteurs est close par union.

□

Corollaire 56. *Toute semi commutation à compteurs peut être décomposée en semi commutations à compteurs atomiques.*

Preuve. Nous pouvons remarquer qu'une semi commutation à compteurs atomique est de la forme:

$$\{a\} \times (X \setminus \{a\}) \text{ ou } (X \setminus \{b\}) \times \{b\} \ .$$

Soient θ_1, θ_2,..., θ_n toutes les semi commutations à compteurs atomiques incluses dans une semi commutation à compteurs θ.

Il est évident que $\bigcup_{i \in [1..n]} \theta_i = \theta$.

Donc $f_\theta = f_{\theta_1} f_{\theta_2} \dots f_{\theta_n}$. □

Nous pouvons énoncer un résultat plus fort :

Corollaire 57. *Soit θ une semi commutation à compteurs. Soient θ_1, θ_2,...,θ_n des semi commutations à compteurs (atomiques ou non) telles que $\bigcup_{i \in [1..n]} \theta_i = \theta$.*

Alors $f_\theta = f_{\theta_1} f_{\theta_2} \dots f_{\theta_n}$.

Preuve. Même preuve. □

Nous pouvons aussi remarquer :

Corollaire 58. *La composition de semi commutations à compteurs est totalement commutative.*

Preuve. Soient θ_1, θ_2,..., θ_n des semi commutations à compteurs.

Soit $\theta = \theta_1 \cup \theta_2 \cup \dots \cup \theta_n$.

D'après le **Corollaire 57** $f_\theta = f_{\theta_1} f_{\theta_2} \dots f_{\theta_n}$.

Il est évident que tout ordre sur θ_1, θ_2, ..., θ_n donnera le même résultat. □

6.4 Langages et semi commutations

Théorème 59. *Soit θ une semi commutation. Alors*

$$f_\theta(\mathcal{MC}) \subset \mathcal{MC} \Leftrightarrow \theta \textit{ est une semi commutation à compteurs} .$$

Preuve. 1. $f_\theta(\mathcal{MC}) \subset \mathcal{MC} \Rightarrow \theta$ est une semi commutation à compteurs ?

Nous allons prouver :

$$\theta \text{ n'est pas une semi commutation à compteurs} \Rightarrow \exists L \in \mathcal{MC}, f_\theta(L) \notin \mathcal{MC} .$$

Supposons que θ n'est pas une semi commutation à compteurs. Alors il existe a, b, c et d tels que $(a,b) \in \theta$, $(a,c) \notin \theta$ et $(d,b) \notin \theta$.

(a) Si $(b,a) \in \theta$:
Soit $L = ((ba)^*cd)^*$. Alors $w \in f_\theta(L) \cap (a^+b^+cd)^*$ si et seulement si w est de la forme

$$w = u_o cd u_1 cd \dots u_n cd \ ,$$

avec :

- $\forall i \geq n, u_i \in a^+b^+$,
- $\forall i \geq n, |u_o u_1 \dots u_n|_a \geq |u_o u_1 \dots u_n|_b$.

Soit $\theta_o = \theta^{-1}$: $(a,b) \in \theta_o$, $(b,a) \in \theta_o$, $(c,a) \notin \theta_o$ et $(b,d) \notin \theta_o$.
Alors $w \in f_{\theta_o}(L) \cap (a^+b^+cd)^*$ si et seulement si w est de la forme

$$w = u'_o cd u'_1 cd \dots u'_n cd \ ,$$

avec :

- $\forall i \geq n, u'_i \in a^+b^+$,
- $\forall i \geq n, |u'_o u'_1 \dots u'_n|_a \leq |u'_o u'_1 \dots u'_n|_b$.

Donc $f_\theta(L) \cap f_{\theta_o}(L) \cap (a^+b^+cd)^* = \{a^n b^n cd \,|\, n \in \mathbb{N}^+\}^* \notin \mathcal{MC}$ car ce dernier langage est rationnellement équivalent à $\{a^n b^n \,|\, n \in \mathbb{N}\}^*$.
Donc $f_\theta(L) \notin \mathcal{MC}$.

(b) Si $(b,a) \notin \theta$:
Clairement dans ce cas $f_\theta(L) \cap (b^+a^+cd)^* = \{b^n a^n cd \,|\, n \in \mathbb{N}^+\}^* \notin \mathcal{MC}$ (car il est rationnellement équivalent à $\{a^n b^n \,|\, n \in \mathbb{N}\}^*$).
D'où $f_\theta(L) \notin \mathcal{MC}$.

Donc dans chacun de ces cas $f_\theta(L) \notin \mathcal{MC}$ avec $L \in \mathcal{MC}$.

2. $f_\theta(\mathcal{MC}) \subset \mathcal{MC} \Leftarrow \theta$ est une semi commutation à compteurs ?

Par le **Corollaire 56** nous savons que chaque semi commutation à compteurs peut être décomposée en semi commutations à compteurs atomiques. Nous allons d'abord proposer une preuve pour les semi commutations à compteurs atomiques, et nous étendrons facilement ce résultat à toutes les semi commutations à compteurs.

(a) Pour une semi commutation à compteurs atomique :
Soit une semi commutation à compteurs atomique :

$$\theta = \{a\} \times (X \setminus \{a\}) \ .$$

Soit $L \in \mathcal{MC}$. Nous devons prouver que $f_\theta(L) \in \mathcal{MC}$.
Pour cela nous prouverons que f_θ peut être vue comme composition de transductions rationnelles et intersections avec des langages multicompteurs. $\mathcal{MC}$ étant clos par transduction rationnelle et intersection, nous aurons $f_\theta(L) \in \mathcal{MC}$.
Soit $\bar{a} \notin X$.

- Soit le morphisme g défini par

$$\begin{array}{rcccc} g & : & (X \cup \{\bar{a}\})^* & \longrightarrow & X^* \\ & & \bar{a} & \longmapsto & \varepsilon \\ & & x \in X & \longmapsto & x \end{array}$$

  Soit $L_1 = g^{-1}(L)$.
  $L \in \mathcal{MC}$ et g est une transduction rationnelle, donc $L_1 \in \mathcal{MC}$.

  Remarque. $\bar{a}$ représentera les futures positions de a dans $f_\theta(L)$.

- Soit $L_2 = L_1 \cap (D_1'^* ⧢ (X \setminus \{a\})^*)$ où $D_1'^*$ est le langage de Semi-Dyck sur les lettres a et $\bar{a}$. Comme $(D_1'^* ⧢ (X \setminus \{a\})^*) \in \mathcal{MC}$, nous avons également $L_2 \in \mathcal{MC}$.

  Remarque. Cette intersection avec un langage de Semi-Dyck est utilisée pour s'assurer que chaque $\bar{a}$ est placé après le a qui lui corespond.

- Soit le morphisme f défini par

$$\begin{array}{rcccc} f & : & (X \cup \{\bar{a}\})^* & \longrightarrow & X^* \\ & & a & \longmapsto & \varepsilon \\ & & \bar{a} & \longmapsto & a \\ & & x \in X \setminus \{\bar{a}, a\} & \longmapsto & x \end{array}$$

  Soit $L_3 = f(L_2)$.
  $L_2 \in \mathcal{MC}$ et f est une transduction rationnelle, donc $L_3 \in \mathcal{MC}$.

  Remarque. f efface les a à leurs anciennes positions, et transforme les $\bar{a}$ en a aux nouvelles positions.

Si $\theta = (X \setminus \{b\}) \times \{b\}$ on obtient la même construction mais a est remplacé par b, $\bar{a}$ par $\bar{b}$; $D_1'^*$ est sur $\bar{b}$ et b. D'où $f_\theta(L) = L_3 \in \mathcal{MC}$. Chaque semi commutation à compteurs atomique préserve $\mathcal{MC}$.

(b) Pour toute semi commutation à compteurs :

Nous avons vu (**Corollaire 56**) que toute semi commutation à compteurs peut être décomposée en semi commutations à compteurs atomiques. Toute semi commutation atomique, comme nous venons de le prouver, préserve $\mathcal{MC}$. Leur composition le préservera donc aussi.

D'où on obtient l'équivalence désirée. □

6.5 Décidabilité

En conclusion à l'étude des semi commutations à compteurs et de $\mathcal{MC}$ nous allons donner quelques problèmes qui sont indécidables dans le cas général mais qui deviennent décidables dans le cas particulier des langages multi-compteurs.

6.5.1 Intersection avec un rationnel

D'après les travaux de S.R. Kosaraju ([20]) et E.W. Mayr ([24]) sur la décidabilité des problèmes d'accessibilité pour les systèmes d'addition de vecteurs il est clair que les langages de $\mathcal{MC}$ sont récursifs. Comme $\mathcal{MC}$ est clos par morphisme effaçant, il est facile de voir que le problème du vide y est décidable.

Dans le cas général il est indécidable de savoir si $f_\theta(L) \cap R = \emptyset$ où θ est une semi commutation ([6]), quand que L et R sont des langages rationnels. Cependant nous pouvons donner un corollaire du **Théorème 59**:

Corollaire 60. *Si θ est une semi commutation à compteurs, si L et R sont des langages multi-compteurs, alors il est décidable de savoir si $f_\theta(L) \cap R = \emptyset$.*

Preuve. Si θ est une semi commutation à compteurs, si L et R sont des langages multi-compteurs, alors $f_\theta(L) \cap R$ est un langage à compteurs, donc le problème de $f_\theta(L) \cap R = \emptyset$ devient décidable car le problème du vide est décidable dans $\mathcal{MC}$. □

On a imédiatement le corollaire suivant:

Corollaire 61. *Si θ est une semi commutation à compteurs, si L et R sont des langages rationnels, alors il est décidable de savoir si $f_\theta(L) \cap R = \emptyset$.*

6.5.2 Mots maximaux pour une semi commutation

En étudiant les semi commutations on est assez naturellement amené à considérer la notion de Max_θ:

Définition 62. Soit θ une semi commutation. Soit L un langage. $Max_\theta(L)$ est l'ensemble

$$\{u \in L \mid \forall v \in L, (v \xrightarrow[\theta]{*} u) \Rightarrow (u \xrightarrow[\theta]{*} v)\} .$$

K. Reinhardt a prouvé que la détermination de $Max_\theta(L) = L$ est un problème indécidable pour un langage rationnel défini sur un alphabet de quatre lettres ou plus ([37]). Cependant voici un autre résultat démontré à partir du **Théorème 59**:

Corollaire 63. *Le problème de $Max_\theta(L) = L$ est décidable quand θ est une semi commutation à compteurs et L un langage rationnel.*

Preuve. Il est facile de voir que $Max_\theta(L) = L$ est équivalent à $f_\theta \tau_\theta f_\theta(L) \cap L = \emptyset$, où τ_θ est la transduction rationnelle définie par :

$$\tau_\theta(u) = \{u_1 abu_2 \in X^* \mid u = u_1 bau_2, u_1, u_2 \in X^*, (a,b) \in \theta \cap \bar{\theta}^{-1}\} \ .$$

Comme L est un langage multi-compteurs, $f_\theta(L) \in \mathcal{MC}$, d'où $\tau_\theta f_\theta(L) \in \mathcal{MC}$.

D'après le **Corollaire 60** il est décidable de savoir si $f_\theta \tau_\theta f_\theta(L) \cap L = \emptyset$. □

On a immédiatement le corollaire suivant :

Corollaire 64. *Le problème de $Max_\theta(L) = L$ est décidable quand θ est une semi commutation à compteurs et L un langage rationnel.*

Chapitre 7

Décomposition en semi commutations atomiques maximales

La complexité de l'algorithme de décomposition d'une semi commutation en semi commutations atomiques peut être mesurée par le nombre de semi commutations obtenues. Un trop grand nombre de semi commutations atomiques handicaperait énormément son éventuelle implémentation. (Il est à noter qu'il existe une implémentation limitée de cet algorithme.)

Or, tel qu'il a été défini dans les chapitres précédents, cet algorithme semble *encombrant*: une étude rapide nous permet facilement d'évaluer le nombre maximum de semi commutations atomiques que l'on peut obtenir pour une semi commutation de n règles comme étant de 2^n. En effet la seule chose que garantisse l'algorithme est qu'en cas de décomposition de f_θ en $f_{\theta'} f_{\theta''}$, on a $\theta' \subsetneq \theta$ et $\theta'' \subsetneq \theta$. Dans le pire des cas on pourrait donc avoir:

$$\mathrm{Card}(\theta') = \mathrm{Card}(\theta'') = \mathrm{Card}(\theta) - 1 \ .$$

Il est donc clair que l'on peut construire un arbre de décomposition (voir figure 7.1) binaire équilibré, et dont les feuilles seront les semi commutations atomiques obtenues en décomposant θ. Sa hauteur serait, toujours dans le pire des cas, de n, car, à chaque étage de l'arbre, les semi commutations auraient une règle de moins que celles de l'étage précédent: les feuilles, qui correspondent au résultat de la décomposition, seront donc au nombre de 2^n.

Cependant l'étude des semi commutations à compteurs nous a amenés à nous pencher sur certaines semi commutations atomiques que nous avons appelées semi commutations atomiques maximales.

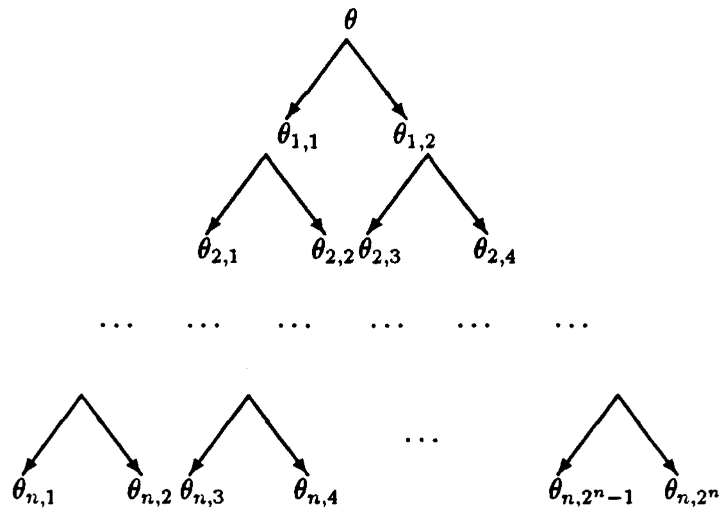


Figure 7.1 : Exemple de décomposition dans le pire des cas

Celles-ci semblaient permettre une décomposition de toute semi commutation en un nombre de semi commutations atomiques (maximales) égal au nombre de règles de la semi commutation de départ.

Mais, en fait, leur étude nous a permis de prouver que la complexité de l'algorithme est bien exponentielle par rapport au nombre de règles de la semi commutation à décomposer.

7.1 Définitions et résultats préliminaires

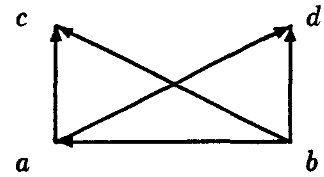
Commençons par la définition des semi commutations atomiques maximales :

Définition 65. Pour toute semi commutation θ on appellera *semi commutation atomique maximale* de θ, toute semi commutation θ' atomique telle qu'il n'existe aucune autre semi commutation θ'' atomique telle que

$$\theta' \subsetneq \theta'' \subset \theta \ .$$

Exemple 9. Pour la semi commutation θ de la figure 7.2 les semi commutations atomiques sont celles de la figure 7.3 ainsi que toutes les semi commutations réduites à une règle. Les seules *semi commutations atomiques maximales* sont θ_3 et θ_7.

L'idée de cet énoncé est de définir un type de semi commutations atomiques qui permettraient d'éliminer les semi commutations atomiques inutiles ; quoi de plus inutile, a



θ

Figure 7.2 : Une semi commutation

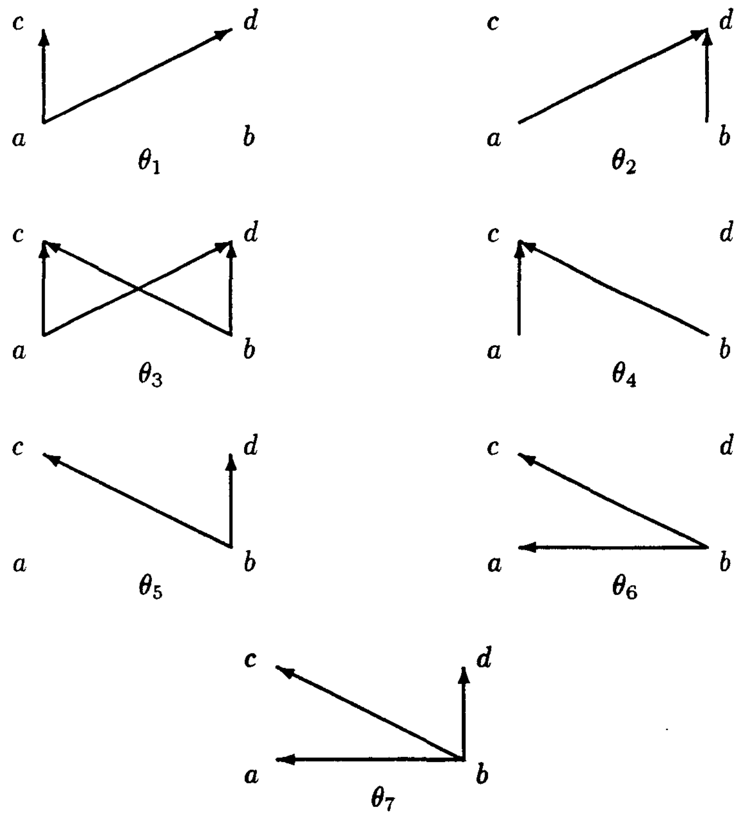


Figure 7.3 : Semi commutation atomiques maximales

priori, qu'une semi commutation dont l'action pourrait être effectuée par une autre (qui la contiendrait) ?

Définition 66. Pour toute semi commutation θ, on notera M_θ l'ensemble de toutes les semi commutations atomiques maximales de θ :

$$M_\theta = \{\theta' \subset \theta \mid (\theta' \text{ atomique}) \text{ et } (\nexists \theta'' \text{ atomique}, \theta' \subsetneq \theta'' \subset \theta)\} \ .$$

Grâce au **Lemme 35** nous allons pouvoir démontrer les deux lemmes suivants qui sont à la base de nos résultats ; ils affirment en effet qu'il est toujours possible de décomposer une semi commutation en n'utilisant que des semi commutations atomiques maximales.

Lemme 67. *Pour toute semi commutation θ, pour toute partition $\{M'_\theta, M''_\theta\}$ de M_θ on pose :*

$$\theta' = \bigcup_{\theta_i \in M'_\theta} \theta_i \ \text{ et } \ \theta'' = \bigcup_{\theta_i \in M''_\theta} \theta_i \ .$$

Les semi commutations θ' et θ'' peuvent alors être composées, et leur composition est θ.

Preuve. Il est évident que $\theta = \theta' \cup \theta''$. Il ne reste qu'à démontrer que θ' et θ'' sont composables. Pour cela nous utiliserons le **Théorème 26**. Soit un cycle (voir figure 7.4).

$$\begin{tikzcd}
 & x_n & & x_o \arrow[ll, "\theta' \cap \bar{\theta}''"'] \arrow[d] & \\
 & x_{n-1} \arrow[u] & & x_1 & \\
\bar{\theta}' \cup \bar{\theta}''^{-1} & \vdots & & \vdots & \bar{\theta}' \cup \bar{\theta}''^{-1} \\
 & x_{j+2} & & x_{j-1} \arrow[d] & \\
 & x_{j+1} \arrow[u] & & x_j \arrow[ll, "\bar{\theta}' \cap \theta''"] &
\end{tikzcd}$$

Figure 7.4 : Cycle de composabilité

Pour que θ' et θ'' soient composables il faut que pour tout tel cycle il existe un couple $(x_j, x_k) \in \bar{\theta}' \cap \bar{\theta}''$, où $j \in [0..i]$ et $k \in [i+1..n]$. Nous allons en faire la preuve par l'absurde. Supposons qu'il n'existe pas de tel couple.

- Tout couple (x_j, x_k) tel que $j \in [0..i]$ et $k \in [i+1..n]$ sera alors dans $\theta' \cup \theta''$. Il en est de même pour (x_o, x_{i+1}).

  Supposons que $(x_o, x_{i+1}) \in \theta'' \backslash \theta'$. On a $(x_o, x_n) \in \theta' \subset \theta$ et $(x_o, x_{i+1}) \in \theta'' \subset \theta$. Nous en concluons que $\theta_o = \{x_o\} \times \{x_n, x_{n+1}\}$ est une semi commutation atomique incluse dans θ : il existe une semi commutation atomique maximale θ_1 telle que $\theta_o \subset \theta_1$.

  Par conséquent on a soit $\theta_1 \in M'_\theta$, soit $\theta_1 \in M''_\theta$.

  Mais

  - $(x_o, x_{i+1}) \notin \theta'$ d'où $\theta_1 \notin M'_\theta$,
  - $(x_o, x_n) \notin \theta''$ d'où $\theta_1 \notin M''_\theta$.

  On en arrive a une impossibilité, nous devons en conclure que :

  $$(x_o, x_{i+1}) \in \theta' \ .$$

- De la même façon, on prouve que :

  $$(x_i, x_n) \in \theta'' \ .$$

- D'après les points précédents, on a :

  - $(x_o, x_n) \in \theta' \setminus \theta''$,
  - $(x_o, x_{i+1}) \in \theta'$,
  - $(x_i, x_n) \in \theta''$,
  - $(x_i, x_{i+1}) \in \theta'' \setminus \theta'$.

  De ce fait $\theta_2 = \{x_o, x_i\} \times \{x_{i+1}, x_n\}$ est une semi commutation atomique incluse dans θ. Il existe alors une semi commutation atomique maximale θ_3 telle que $\theta_2 \subset \theta_3$; par conséquent on a soit $\theta_3 \in M'_\theta$, soit $\theta_3 \in M''_\theta$.

  Mais

  - $(x_i, x_{i+1}) \notin \theta'$ d'où $\theta_3 \notin M'_\theta$.
  - $(x_o, x_n) \notin \theta''$ d'où $\theta_3 \notin M''_\theta$,

  On en arrive a une impossibilité.

Donc pour tout tel cycle il existe un couple $(x_j, x_k) \in \bar{\theta}' \cap \bar{\theta}''$, où $j \in [0..i]$ et $k \in [i+1..n]$.

Nous pouvons en conclure que θ' et θ'' sont composables. □

Le lemme précédent, bien qu'apportant une information non négligeable, n'est pas déterminant ; en effet, pouvoir décomposer θ en θ' et θ'' n'a d'intérêt que si $\theta' \subsetneq \theta$ et $\theta'' \subsetneq \theta$. Le lemme suivant nous apporte le résultat important.

Lemme 68. *Pour toute semi commutation θ non atomique, il existe une partition*

$$\{M'_\theta, M''_\theta\}$$

de M_θ telle que

$$\theta' \subsetneq \theta \text{ et } \theta'' \subsetneq \theta \ .$$

Preuve. La semi commutation θ n'étant pas atomique :

$$\exists(a,b) \in \theta, \exists(c,d) \in \theta \text{ tels que } (a,d) \notin \theta \ ,$$

(d'après le **Lemme 35**). Soit une semi commutation atomique maximale θ_o contenant (a,b) ; alors $(c,d) \notin \theta_o$ (sinon, d'après le **Lemme 35**, on aurait $(a,d) \in \theta_o \subset \theta$).

Soit alors

$$M'_\theta = \{\theta_i \in M_\theta \,|\, (a,b) \in \theta_i\} \ ,$$

et

$$M''_\theta = \{\theta_i \in M_\theta \,|\, (a,b) \notin \theta_i\} \ .$$

$\{M'_\theta, M''_\theta\}$ est bien une partition de M_θ et on a :

- $\theta' \subsetneq \theta$ car $(c,d) \notin \theta'$,
- $\theta'' \subsetneq \theta$ car $(a,b) \notin \theta''$.

□

Le dernier résultat préliminaire va nous permettre d'affirmer que les partitions utilisées précédemment ne créent pas de semi commutations atomiques maximales supplémentaires.

Lemme 69. *Avec les notations précédentes on a :*

$$M_{\theta'} = M'_\theta \text{ et } M_{\theta''} = M''_\theta \ .$$

Preuve. • $\theta' = \bigcup_{\theta_i \in M'_\theta} \theta_i$ donc toute θ_i est atomique car appartenant à M'_θ.

De plus il n'existe pas de θ_j atomique telle que

$$\theta_i \subsetneq \theta_j \subset \theta' \subset \theta$$

donc tous les θ_i de M'_θ sont dans $M_{\theta'}$:

$$M'_\theta \subset M_{\theta'} \ .$$

- De la même façon, on prouve que :

$$M''_\theta \subset M_{\theta''} \ .$$

- S'il existe $\theta_o \in M_{\theta'}$ telle que $\theta_o \notin M'_\theta$:

  θ_o étant atomique (puisque dans $M_{\theta'}$), on peut trouver θ_j telle que

$$\theta_o \subsetneq \theta_j \subset \theta \ .$$

  Comme $\theta_j \not\subset \theta'$ (sinon on aurait $\theta_o \subsetneq \theta_j \subset \theta'$, ce qui est impossible car θ_j est une semi commutation atomique maximale), on a $\theta_j \notin M'_\theta$.

  De la même façon on prouve que $\theta_j \notin M''_\theta$.

  On en arrive à une contradiction puisque $\theta_j \in M_\theta$.

  Nous pouvons en conclure que $M_{\theta'} \subset M'_\theta$.

- De la même façon on prouve que $M_{\theta''} \subset M''_\theta$.

Nous avons donc prouvé que

$$M_{\theta'} = M'_\theta \text{ et } M_{\theta''} = M''_\theta \ .$$

□

7.2 Décomposition en semi commutations atomiques maximales

Ces lemmes nous permettent de donner un résultat général de décomposition :

Proposition 70. *Pour toute semi commutation θ il existe une décomposition où ne figurent que des semi commutations atomiques maximales de M_θ, chacune au plus une fois.*

Preuve. Ce résultat est évident si on remarque que, grâce au **Lemme 68** et au **Lemme 69**, pour toute semi commutation θ, on peut partager M_θ de façon que chaque semi commutation atomique maximale n'apparaisse qu'une fois. □

Dans le cas des semi commutations antisymétriques nous avons un résultat complémentaire à la **Proposition 70**. En effet la proposition suivante prouve que non seulement on peut n'utiliser que des semi commutations atomiques maximales pour décomposer une semi commutation antisymétrique, mais on doit les utiliser toutes.

Proposition 71. *Pour toute semi commutation anti-symétrique θ, toute semi commutation atomique maximale doit apparaître au moins une fois dans une décomposition en semi commutations atomiques maximales.*

Preuve. Soit θ une semi commutation. D'après la **Proposition 70** il existe une décomposition de θ où ne figurent que les semi commutations atomiques maximales de M_θ, chacune une et une seule fois.

Pour prouver que les semi commutations atomiques maximales doivent apparaître dans la décomposition d'une semi commutation antisymétrique, nous allons prouver que pour toute décomposition en semi commutations atomiques maximales

$$f_\theta = f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}$$

n'utilisant pas une certaine semi commutation atomique maximale θ_o, il existe deux mots u et v tels que

$$v \in f_\theta(u) \setminus f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1}(u) \ .$$

La preuve va être faite par l'absurde : supposons qu'il existe des semi commutations atomiques maximales telles que

$$f_\theta = f_{\theta_n} f_{\theta_{n-1}} \cdots f_{\theta_2} f_{\theta_1} \ .$$

Pour tout $i \in [1..n]$, il existe $(a_i, b_i) \in \theta_o \setminus \theta_i$ (sinon $\theta_o \subset \theta_i$, ce qui est impossible pour deux semi commutations atomiques maximales).

Soit alors les mots

$$\alpha = a_n a_{n-1} \ldots a_2 a_1 \ , \ \beta = b_1 b_2 \ldots b_{n-1} b_n \ , \ u = \alpha\beta \text{ et } v = \beta\alpha \ .$$

Il est évident que $v \in f_{\theta_o}(u) \subset f_\theta(u)$.

D'après l'hypothèse :

$$u \xrightarrow[\theta_1]{*} u_1 \xrightarrow[\theta_2]{*} u_2 \xrightarrow[\theta_3]{*} \ldots \xrightarrow[\theta_{n-1}]{*} u_{n-1} \xrightarrow[\theta_n]{*} u_n = v \ .$$

On peut remarquer deux faits importants :

- $\{a_1, a_2, \ldots, a_n\} \times \{b_1, b_2, \ldots, b_n\} \subset \theta_o$ qui est atomique ; on a donc $\{a_1, a_2, \ldots, a_n\} \cap \{b_1, b_2, \ldots, b_n\} = \emptyset$.

- θ est antisymétrique ; toute dérivation $xy \xrightarrow[\theta]{} yx$ ne pourra être *défaite* une fois *faite*. Comme
$$\Pi_\alpha(u) = \Pi_\alpha(v) \text{ et } \Pi_\beta(u) = \Pi_\beta(v) \ ,$$
il est évident que, d'après la remarque précédente, nous ne pourrons avoir que
$$\forall i \in [1..n], \Pi_\alpha(u_i) = \Pi_\alpha(u) = \alpha \text{ et } \Pi_\beta(u_i) = \Pi_\beta(u) = \beta \ .$$

Nous pouvons maintenant affirmer :

- $f_{\theta_1}(u) = \{u\}$ car :
  - $(a_1, b_1) \in \theta_o \setminus \theta_1$,
  - si $u \xrightarrow[\theta_1]{*} u_1, \Pi_\alpha(u_1) = \alpha, \Pi_\beta(u_1) = \beta$.
- $f_{\theta_2} f_{\theta_1}(u) \subset (a_n \ldots a_3 a_2 b_2 b_3 \ldots b_n) \sqcup \text{com}(a_1 b_1)$ pour les mêmes raisons.
- etc...
- $f_{\theta_n} \ldots f_{\theta_2} f_{\theta_1}(u) \subset (a_n b_n) \sqcup \text{com}(a_{n-1} \ldots a_2 a_1 b_1 b_2 \ldots b_{n-1})$.

Or, d'après les remarques précédentes, il est impossible que $a_n b_n$ soit un sous-mot de v. Donc $v \notin f_{\theta_n} \ldots f_{\theta_2} f_{\theta_1}(u)$.

L'hypothèse choisie nous amène donc à une contradiction. □

C'est ce dernier résultat qui va nous permettre de conclure.

7.3 Complexité de l'algorithme de décomposition

Proposition 72. *La complexité de l'algorithme décomposition des semi commutations est exponentielle par rapport au nombre de règles de la semi commutation de départ.*

Preuve. Soit θ_a la semi commutation totale sur un alphabet de n lettres
$$X_a = \{a_1, a_2, \ldots, a_n\} \ .$$
(Cela signifie que $\theta = X_a \times X_a \setminus \Delta_{X_a}$.) Cette semi commutation possède $n \times (n-1)$ règles :
$$\begin{aligned} \theta_a \ = \ & \{(a_1,a_2),(a_1,a_3),\ldots,(a_1,a_n), \\ & (a_2,a_1),(a_2,a_3),\ldots,(a_2,a_n), \\ & \ldots \\ & (a_n,a_1),(a_n,a_2),\ldots,(a_n,a_{n-1})\} \ . \end{aligned}$$

Soit l'alphabet $X_b = \{b_1, b_2, \ldots, b_n\}$ disjoint de X_a.

On peut alors définir sur l'alphabet $X = X_a \cup X_b$ la semi commutation

$$\theta = \{(a_i, b_j) \,|\, (a_i, a_j) \in \theta_a\} \ .$$

Cette semi commutation est antisymétrique puisque X_a et X_b sont disjoints, et elle comporte évidemment autant de règles que θ_a, c'est-à-dire $n \times (n-1)$ règles.

Evaluons maintenant le nombre de semi commutations atomiques maximales de θ.

Toute semi commutation atomique incluse dans θ doit être une partie de $X_a \times X_b$. De plus elle doit correspondre à une semi commutation atomique incluse dans θ_a qui, elle, doit être de la forme $A \times B$, A et B étant deux sous-ensembles, disjoints, de X_a. On en conclut que toute semi commutation atomique incluse dans θ ne peut contenir que des couples (a_i, b_j) tels que $i \neq j$. Les semi commutations atomiques maximales de θ sont donc toutes les semi commutations atomiques de la forme $A \times B$ où A et B sont tels que :

- $A = \{a_i \,|\, i \in I\}$,
- $B = \{b_i \,|\, i \in J\}$,
- I et J forment une partition de $[1..n]$.

Il y aura alors :

- C_n^1 semi commutations atomiques maximales telles que $\text{Card}(A) = 1$,
- C_n^2 semi commutations atomiques maximales telles que $\text{Card}(A) = 2$,
- C_n^3 semi commutations atomiques maximales telles que $\text{Card}(A) = 3$,
- ...
- C_n^{n-1} semi commutations atomiques maximales telles que $\text{Card}(A) = n-1$.

Il y aura ainsi

$$C_n^1 + C_n^2 + C_n^3 + \ldots + C_n^{n-1} = 2^n - 2$$

semi commutations atomiques maximales dans θ.

En résumé : nous avons une semi commutation qui a N règles ($N = n \times (n-1)$), et qui possède un nombre de semi commutations atomiques maximales de l'ordre de $2^{\sqrt{N}}$.

Comme cette semi commutation est antisymétrique, d'après la **Proposition 71**, toute décomposition de cette semi commutation doit contenir ces $2^{\sqrt{N}}$ semi commutations atomiques maximales.

La complexité de l'algorithme est bien exponentielle, car nous avons vu plus haut que dans le pire des cas on pouvait trouver 2^N semi commutations atomiques maximales pour N règles. □

La valeur réelle de la complexité, exprimée en nombres de semi commutations atomiques obtenues, est donc comprise entre $2^{\sqrt{N}}$ et 2^N pour N étant le nombre de couples de la semi commutation de départ.

Chapitre 8

Fermeture d'un rationnel par une semi commutation

Les problèmes de fermeture sont classiques dans l'étude des langages et des opérations qui s'appliquent sur eux. Le but de ce chapitre est de fournir un algorithme simple qui permet de calculer, pour un langage rationnel R donné, la plus grande semi commutation θ telle que $f_\theta(R) = R$, et, par la-même, de reconnaître si un langage rationnel donné (dont on connaît l'automate réduit déterministe) est fermé par une semi commutation donnée.

Il est à noter que cet algorithme est à rapprocher de la construction de l'automate minimal d'Eilenberg ([17]) : cette dernière construit une relation d'équivalence, alors qu'ici il s'agit d'une relation d'ordre.

8.1 Etat meilleur

Soit un langage rationnel R dont on connaît l'automate réduit déterministe

$$(X, Q, \delta, I, F)$$

où X est l'alphabet, Q est l'ensemble des états, δ est une fonction de transition, I est l'ensemble des états initiaux (réduit à un élément Q_o, car l'automate est réduit déterministe), F est l'ensemble des états terminaux.

Définition 73. Pour $q \in Q$ nous définissons $R_q = \{u \mid \delta(q, u) \in F\}$.

R_q est l'ensemble des mots qui permettent d'atteindre un état terminal en partant de q.

Nous pouvons alors définir une relation d'ordre sur Q :

Définition 74. On notera $q \leq q'$ (et on dira que q' est *meilleur* que q) si et seulement si $R_q \subset R_{q'}$.

L'état q' sera en effet meilleur que q dans le sens qu'en partant de q', il y a un plus grand nombre de mots qui permettent d'atteindre un état terminal qu'en partant de q.

Nous pouvons alors énoncer le lemme suivant :

Lemme 75. *Soient q et q', deux états d'un automate déterministe. Si $q \not\leq q'$, $\delta(q_1, a) = q$ et $\delta(q'_1, a) = q'$ alors $q_1 \not\leq q'_1$.*

Preuve. Si $q \not\leq q'$ alors $R_q \not\subset R_{q'}$, donc il existe u tel que $\delta(q, u) \in F$ et $\delta(q', u) \notin F$.

On en conclut donc que $\delta(q_1, au) \in F$ et $\delta(q'_1, au) \notin F$; d'où $q_1 \not\leq q'_1$. □

Définissons maintenant la suite $\{H_i\}_{i \in \mathbb{N}}$:

- $H_o = F \times (Q \setminus F)$,
- pour tout i, $H_{i+1} = H_i \cup \{(q, q') \mid \exists a \in X, (\delta(q, a), \delta(q', a)) \in H_i\}$.

Proposition 76. *Il existe un rang N tel que $\forall k \geq N$, $H_{k+1} = H_k$.*

Preuve. La suite $\{H_i\}_{i \in \mathbb{N}}$ étant croissante (au sens de l'inclusion) et incluse dans $Q \times Q$ (qui est de cardinal fini) le résultat est évident. □

Soit alors

$$H = \bigcup_{i \geq 0} H_i \ .$$

Nous allons prouver que H est le complémentaire du graphe de la relation $\leq$.

Proposition 77. $H = \{(q, q') \mid q \not\leq q'\}$.

Preuve. • $H \subset \{(q, q') \mid q \not\leq q'\}$?

La démonstration sera faite par récurrence.

– Soit $(q_o, q'_o) \in H_o$. Par hypothèse :

$$q_o \in F \text{ et } q'_o \notin F \ ,$$

donc

$$\varepsilon \in R_{q_o} \text{ et } \varepsilon \notin R_{q'_o} \ ,$$

donc

$$R_{q_o} \not\subset R_{q'_o} \ ,$$

donc

$$q_o \not\leq q'_o \ .$$

– Supposons que pour tout $i < n$ on ait :

$$(q_i, q'_i) \in H_i \Rightarrow q_i \not\leq q'_i \text{ (hypothèse de récurrence).}$$

Cette relation est-elle encore vérifiée pour n ?
Soit $(q_1, q'_1) \in H_n$. Alors deux cas sont possibles :

$$(q_1, q'_1) \in H_{n-1} \text{ ou } (q_1, q'_1) \notin H_{n-1} \ .$$

* Si $(q_1, q'_1) \in H_{n-1}$, il suffit d'utiliser l'hypothèse de récurrence.
* Si $(q_1, q'_1) \notin H_{n-1}$ alors il existe une lettre a et deux états q_o et q'_o tels que

$$\delta(q_o, a) = q_1, \ \delta(q'_o, a) = q'_1 \text{ et } (q_o, q'_o) \in H_{n-1} \ .$$

On a donc

$$q_o \not\leq q'_o \ ,$$

donc

$$R_{q_o} \not\subset R_{q'_o} \ ,$$

donc il existe un mot $u \in R_{q_o} \setminus R_{q'_o}$, c'est-à-dire

$$\delta(q_o, u) \in F \text{ mais } \delta(q'_o, u) \notin F \ .$$

Donc

$$\delta(q_1, au) = \delta(q_o, u) \in F \text{ mais } \delta(q'_1, au) = \delta(q'_o, u) \notin F \ ,$$

donc

$$q_1 \not\leq q'_1 \ .$$

L'hypothèse de récurence est vérifiée pour tout n.

- $H \supset \{(q,q') \,|\, q \not\leq q'\}$?

  Soit $q' \not\leq q$. On a alors un mot $u = a_1 a_2 \dots a_n$, tel que $\delta(q,u) \in F$ et $\delta(q',u) \notin F$.

  Soient les suites d'états $\{q_i\}_{i\in[0..n]}$ et $\{q'_i\}_{i\in[0..n]}$ définies par

  $$q_i = \delta(q, a_1 a_2 \dots a_i) < \text{ et } q'_i = \delta(q', a_1 a_2 \dots a_i) \ .$$

  (On a donc $q_n = \delta(q,u)$ et $q'_n = \delta(q',u)$.)

  Comme $q_n \in F$ et $q'_n \in Q \setminus F$, $(q_n, q'_n) \in H_o$ par hypothèse.

  Or $\delta(q_{n-1}, a_n) = q_n$ et $\delta(q'_{n-1}, a_n) = q'_n$. Donc $(q_{n-1}, q'_{n-1}) \in H_1$.

  De la même manière, par récurrence, on montre que $(q,q') \in H$.

Donc $H = \{(q,q') \,|\, q \not\leq q'\}$. □

8.2 Plus grande semi commutation pour laquelle R est fermé

Proposition 78. *Soit θ une semi-commutation, soit R un langage rationnel reconnu par l'automate déterministe (X, Q, δ, I, F), alors:*

$$R = f_\theta(R) \text{ si et seulement si } \forall q \in Q, \forall (b,a) \in \theta, \delta(q,ba) \leq \delta(q,ab) \ .$$

Preuve. Il faut d'abord remarquer que l'on a

$$\delta(q,ba) \leq \delta(q,ab)$$

si et seulement si

$$\forall u \in X^*, \quad \delta(q, bau \in F) \Rightarrow \delta(q,abu) \in F \quad . \tag{8.1}$$

- $R = f_\theta(R) \Rightarrow \forall q \in Q, \forall (b,a) \in \theta, \delta(q,ba) \leq \delta(q,ab)$?

  Soit un état q. Soit v un mot permettant de l'atteindre:

  $$\delta(Q_o, v) = q \ .$$

  S'il n'existe pas de mot u tel que $\delta(q,bau) \in F$, il est alors évident que

  $$\delta(q,ba) \leq \delta(q,ab) \ .$$

  Sinon, pour tout mot u tel que $\delta(q,bau) \in F$, alors $vbau \in R$, et $vabu$, qui est dans $f_\theta(vbau)$, est aussi dans R. On en conclut alors que $\delta(q,abu) \in F$.

  On a bien $\delta(q,ba) \leq \delta(q,ab)$.

- $\forall q \in Q, \forall (b,a) \in \theta, \delta(q,ba) \leq \delta(q,ab) \Rightarrow R = f_\theta(R)$?

  Soit un mot $w = vbau$ de R tel que $(b,a) \in \theta$.

  Soit $w' = vabu$.

  Considérons alors l'état q_1 tel que $d(Q_o, v) = q_1$. (Cet état est unique puisque l'automate est déterministe.) Le mot w étant dans le langage R, on a $\delta(Q_o, vbau) \in F$ donc on a aussi $\delta(Q_o, bau) \in F$.

  D'après (8.1), on a aussi $\delta(q_1, abu) \in F$, ce qui entraine immédiatement que

  $$\delta(Q_o, vabu) \in F$$

  c'est-à-dire que $vabu$ est un mot de R.

  Si $w' \in f_\theta(w)$, il est évident que nous pourrions faire le même raisonnement pour chaque étape de la dérivation de w en w'

  $$w \underset{\theta}{\longrightarrow} w_1 \underset{\theta}{\longrightarrow} w_2 \underset{\theta}{\longrightarrow} \ldots \underset{\theta}{\longrightarrow} w_n = w'$$

  et que nous démontrerions ainsi, de proche en proche, que

  $$w_1 \in R \ ,$$

  puis que

  $$w_2 \in R \ , \ldots$$

  et enfin que

  $$w' \in R \ .$$

  On a bien $f_\theta(R) \subset R$, d'où $f_\theta(R) = R$.

En conclusion, on a

$$R = f_\theta(R) \text{ si et seulement si } \forall q \in Q, \forall (b,a) \in \theta, \delta(q,ba) \leq \delta(q,ab) \ .$$

□

Grâce à ce résultat, nous pouvons écrire:

Corollaire 79. *Pour un langage rationnel R donné, la plus grande semi commutation pour laquelle R est fermé est:*

$$\Theta_R = \{(b,a) \in X \times X \mid \forall q \in Q, \delta(q,ba) \leq \delta(q,ab)\} \ .$$

Preuve. Immédiat. □

Nous pouvons alors conclure :

Corollaire 80. *Soit un langage rationnel R. Soit une semi commutation θ. Alors :*

$$f_\theta(R) = R \text{ si et seulement si } \theta \subset \Theta_R \ .$$

Preuve. Immédiat. □

Bibliographie

[1] I.J. Aalbersberg et G. Rozenberg. Theory of traces. *Theoretical Computer Science*, 60:1–82, 1988.

[2] W. Aigner. Combinatorial theory. Springer Verlag.

[3] J. Berstel et J. Sakarovich. Recents results in the theory of rational sets. In J. Gruska, B. Rovan, and J. Wiedermann, editors, *Proceeding of the 13th Symposium on Mathematical Foundations of Computer Science (MFCS'86), Bratislava (Czechoslovakia) 1986*, number 233 in Lecture Notes in Computer Science, pages 15–28. Springer, Berlin-Heidelberg-New York, 1986.

[4] P. Cartier et D. Foata. *Problèmes combinatoires de commutation et réarrangements.* Lecture Notes in Computer Science. Springer, Berlin-Heidelberg-New York, 1969.

[5] C. Choffrut. Free partially commutative monoids. Technical report 86/20, LITP Université de Paris 7, 1986.

[6] M. Clerbout. *Commutations partielles et familles de langages.* Thèse, Université de Lille (France), 1984.

[7] M. Clerbout. Compositions de fonctions de commutation partielle. *R.A.I.R.O.-Informatique Théorique et Applications*, pages 395–424, 1986.

[8] M. Clerbout et D. Gonzalez. Decomposition of semi commutations. In B. Rovan, editor, *Proceedings of the 15th Symposium on Mathematical Foundations of Computer Science (MFCS'90), Banskà Bystrica (Czechoslovakia) 1990*, number 452 in Lecture Notes in Computer Science, pages 209–216, 1990. Extended version: *Atomic semi-commutations*, to appear in TCS.

[9] M. Clerbout et D. Gonzalez. Atomic semi commutations. Technical Report I. T. 216, Laboratoire d' Informatique fondamentale de Lille, UFR d'IEEA, Batiment M3, F-59655 Villeneuve d' Ascq CEDEX, 1991.

[10] M. Clerbout, D. Gonzalez, M. Latteux, E. Ochmański, Y. Roos, et P.A. Wacrenier. Recognizable morphisms on semi commutations. Technical Report I.T. 238, Laboratoire d'Informatique Fondamentale de Lille, 1992.

[11] M. Clerbout et M. Latteux. Partial commutations and faithful rational transductions. *Theoretical Computer Science*, 35:241–254, 1985.

[12] M. Clerbout et M. Latteux. Semi-Commutations. *Information and Computation*, 73:59–74, 1987.

[13] M. Clerbout, M. Latteux, et Y. Roos. Decomposition of partial commutations. Publication interne I.T. 173, Laboratoire d'Informatique Fondamentale de Lille, 1989.

[14] M. Clerbout, M. Latteux, Y. Roos, et W. Zielonka. Semi commutations and rational expressions. In W. Kuich, editor, *Proceedings of the 19th International Colloquium on Automata Languages and Programming (ICALP'92), Wien (Austria) 1992*, number 623 in Lecture Notes in Computer Science, pages 113–125. Springer, Berlin-Heidelberg-New York, 1992.

[15] R. Cori. Partially abelian monoids. In *STACS, Orsay*, 1986. Invited lecture.

[16] V. Diekert. *Combinatorics on Traces*. Number 454 in Lecture Notes in Computer Science. Springer, Berlin-Heidelberg-New York, 1990.

[17] S. Eilenberg. *Automata, Languages and Machines*, volume I. Academic Press, New York and London, 1974.

[18] D. Gonzalez. Semi-commutations et multi-compteurs. *Bulletin of the European Association for Theoretical Computer Science (EATCS)*, 47:240–242, Sept 1992.

[19] D. V. Hung et E. Knuth. Semi-commutations and Petri-nets. *Theoretical Computer Science*, 34:67–81, 1989.

[20] S.R. Kosaraju. Decidability of reachability in vector addition systems. In *14th Ann. Symp. on Theory Of Computing*, pages 267–281, 1982.

[21] J. Latteux et J. Leguy. On the composition of morphisms and inverse morphisms. In *Proceedings of the 10th International Colloquium on Automata Languages and Programming (ICALP'83)*, number 154 in Lecture Notes in Computer Science, pages 430–432, 1983.

[22] M. Latteux. Langages à un compteur. *Journal of Computer and System Sciences*, 26:37–54, 1983.

[23] M. Latteux. Some results on semi-commutations. In M. Ito, editor, *Proc. of the International Colloquium on Words, Languages and Combinatorics (Kyoto 1990)*. World Scientific, Singapore, 1992.

[24] E. Mayr. An algorithm for the general Petri net reachability problem. *SIAM Journal of Computing*, 13:441–459, 1984.

[25] A. Mazurkiewicz. Concurrent program schemes and their interpretations. DAIMI Rep. PB 78, Aarhus University, Aarhus, 1977.

[26] A. Mazurkiewicz. Traces, histories, graphs: Instances of a process monoid. In M.P. Chytil et al., editors, *Proceeding of the 11th Symposium on Mathematical Foundations of Computer Science (MFCS'84), Praha (CSFR) 1984*, number 176 in Lecture Notes in Computer Science, pages 115–133. Springer, Berlin-Heidelberg-New York, 1984.

[27] Y. Métivier. On recognizable subsets of free partially commutative monoids. In L. Kott, editor, *Proceedings of the 13th International Colloquium on Automata Languages and Programming (ICALP'86), Rennes (France) 1986*, number 226 in Lecture Notes in Computer Science, pages 254–264. Springer, Berlin-Heidelberg-New York, 1986.

[28] Y. Métivier. Une condition suffisante de reconnaissabilité dans un monoïde partiellement commutatif. *R.A.I.R.O.-Informatique Théorique et Applications*, 20:121–127, 1986.

[29] Y. Métivier et E. Ochmański. On lexicographic semi-commutations. *Information Processing Letters*, 26:55–59, 1987.

[30] Minsky. Recursive unsolvability of post's problem of tag and other topics in the theory of turing machines. *Annals of Mathematics*, 74:437–455, 1961.

[31] M. Nivat. Transduction des langages de chomsky. *Ann. de l'Inst. Fourier*, 18:339–456, 1968.

[32] E. Ochmański. Regular behaviour of concurrent systems. *Bulletin of the European Association for Theoretical Computer Science (EATCS)*, 27:56–67, Oct 1985.

[33] E. Ochmański. Semi-Commutation and Petri Nets. In V. Diekert, editor, *Proceedings of the ASMICS workshop Free Partially Commutative Monoids, Kochel am See 1989*, Report TUM-I9002, Technical University of Munich, pages 151–166, 1990.

[34] E. Ochmański. Semi-commutations and deterministic petri-nets. In B. Rovan, editor, *Proceedings of the 15th Symposium on Mathematical Foundations of Computer Science (MFCS'90), Banská Bystrica (Czechoslovakia) 1990*, number 452 in Lecture Notes in Computer Science, pages 430–438, 1990.

[35] E. Ochmański. Modelling concurrency with semi-commutations. In I. M. Havel and V. Koubek, editors, *Proceedings of the 17th Symposium on Mathematical Foundations of Computer Science (MFCS'92), Prague, (Czechoslovakia), 1992*, number 629 in Lecture Notes in Computer Science, pages 412–420. Springer, Berlin-Heidelberg-New York, 1992.

[36] D. Perrin. Words over a partially commutative alphabet. Report no. 84-59, LITP Université de Paris VII, 1984. Also appeared in A. Apostolico, editor, *Combinatorial Algorithms on Words*, Springer NATO-ASI Series, Vol. F12, p.329-340, 1986.

[37] K. Reinhardt. ASMICS meeting , Rambouillet (France), 1992.

[38] Y. Roos. *Contribution à l'étude des fonctions de commutations partielles.* PhD thesis, Laboratoire d'Informatique Fondamentale de Lille, Université des Sciences et Technologies de Lille, 1989.

[39] Y. Roos et P.A. Wacrenier. Composition of two semi commutations. In A. Tarlecki, editor, *Proceedings of the 16th Symposium on Mathematical Foundations of Computer Science (MFCS'91), Kazimierz Dolny (Poland) 1991*, number 520 in Lecture Notes in Computer Science, pages 406–414. Springer, Berlin-Heidelberg-New York, 1991. Also available as Report I.T. 193, University of Lille (France).

[40] P.A. Wacrenier. *Semi-commutations et reconnaissabilité.* PhD thesis, Laboratoire d'Informatique Fondamentale de Lille, Université des Sciences et Technologies de Lille, 1993.

[41] W. Zielonka. Notes on finite asynchronous automata. *R.A.I.R.O.-Informatique Théorique et Applications*, 21:99–135, 1987.

Table des figures

Index