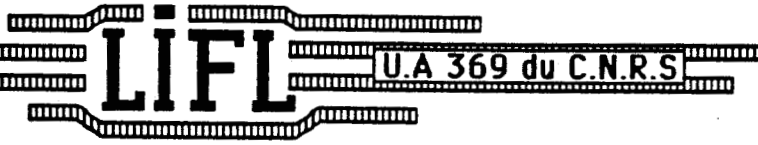


USTL
FLANDRES ARTOIS



50376
1992
51

LABORATOIRE D'INFORMATIQUE FONDAMENTALE DE LILLE

50376
1992
51

N° d'ordre : 847

THÈSE

Nouveau Régime

présentée à

L'UNIVERSITE DES SCIENCES ET TECHNIQUES DE LILLE FLANDRES ARTOIS

pour obtenir le titre de

DOCTEUR en INFORMATIQUE

par

Karine SLOWINSKI



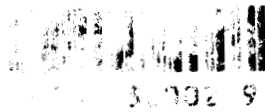
SYSTEMES DE REECRITURE ET LANGAGES DE MOTS DE FIGURE

Thèse soutenue le 20 Janvier 1992 devant la Commission d'Examen

Membres du jury

Président & Rapporteur
Rapporteur
Directeur de thèse
Examineurs

M. NIVAT
D. BEAUQUIER
M. LATTEUX
M. DAUCHET
M. MÉRIAUX
P. SÉÉBOLD



UNIVERSITE DES SCIENCES
ET TECHNIQUES DE LILLE
FLANDRES ARTOIS

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. PAREAU, J. LOMBARD, M. MIGEON, J. CORTOIS.

PRESIDENT DE L'UNIVERSITE DES SCIENCES ET TECHNIQUES
DE LILLE FLANDRES ARTOIS

M. A. DUBRULLE.

PROFESSEURS - CLASSE EXCEPTIONNELLE

M. CONSTANT Eugène	Electronique
M. FOURET René	Physique du solide
M. GABILLARD Robert	Electronique
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. BOSCOQ Denis	Probabilités
M. BOUGHON Pierre	Algèbre
M. BOURIQUET Robert	Biologie Végétale
M. BREZINSKI Claude	Analyse Numérique

M. BRIDOUX Michel
 M. CELET Paul
 M. CHAMLEY Hervé
 M. COEURE Gérard
 M. CORDONNIER Vincent
 M. DAUCHET Max
 M. DEBOURSE Jean-Pierre
 M. DHAINAUT André
 M. DOUKHAN Jean-Claude
 M. DYMENT Arthur
 M. ESCAIG Bertrand
 M. FAURE Robert
 M. FOCT Jacques
 M. FRONTIER Serge
 M. GRANELLE Jean-Jacques
 M. GRUSON Laurent
 M. GUILLAUME Jean
 M. HECTOR Joseph
 M. LABLACHE-COMBIER Alain
 M. LACOSTE Louis
 M. LAVEINE Jean-Pierre
 M. LEHMANN Daniel
 Mme LENOBLE Jacqueline
 M. LEROY Jean-Marie
 M. LHOMME Jean
 M. LOMBARD Jacques
 M. LOUCHEUX Claude
 M. LUCQUIN Michel
 M. MACKE Bruno
 M. MIGEON Michel
 M. PAQUET Jacques
 M. PETIT Francis
 M. POUZET Pierre
 M. PROUVOST Jean
 M. RACZY Ladislav
 M. SALMER Georges
 M. SCHAMPS Joel
 M. SEGUIER Guy
 M. SIMON Michel
 Mlle SPIK Geneviève
 M. STANKIEWICZ François
 M. TILLIEU Jacques
 M. TOULOTTE Jean-Marc
 M. VIDAL Pierre
 M. ZEYTOUNIAN Radyadour

2

Chimie-Physique
 Géologie Générale
 Géotechnique
 Analyse
 Informatique
 Informatique
 Gestion des Entreprises
 Biologie Animale
 Physique du Solide
 Mécanique
 Physique du Solide
 Mécanique
 Métallurgie
 Ecologie Numérique
 Sciences Economiques
 Algèbre
 Microbiologie
 Géométrie
 Chimie Organique
 Biologie Végétale
 Paléontologie
 Géométrie
 Physique Atomique et Moléculaire
 Spectrochimie
 Chimie Organique Biologique
 Sociologie
 Chimie Physique
 Chimie Physique
 Physique Moléculaire et Rayonnements Atmosph.
 E.U.D.I.L.
 Géologie Générale
 Chimie Organique
 Modélisation - calcul Scientifique
 Minéralogie
 Electronique
 Electronique
 Spectroscopie Moléculaire
 Electrotechnique
 Sociologie
 Biochimie
 Sciences Economiques
 Physique Théorique
 Automatique
 Automatique
 Mécanique

PROFESSEURS - 2ème CLASSE

M. ALLAMANDO Etienne
 M. ANDRIES Jean-Claude
 M. ANTOINE Philippe
 M. BART André
 M. BASSERY Louis

Composants Electroniques
 Biologie des organismes
 Analyse
 Biologie animale
 Génie des Procédés et Réactions Chimiques

Mme BATTIAU Yvonne	Géographie
M. BEGUIN Paul	Mécanique
M. BELLET Jean	Physique Atomique et Moléculaire
M. BERTRAND Hugues	Sciences Economiques et Sociales
M. BERZIN Robert	Analyse
M. BKOUICHE Rudolphe	Algèbre
M. BODARD Marcel	Biologie Végétale
M. BOIS Pierre	Mécanique
M. BOISSIER Daniel	Génie Civil
M. BOIVIN Jean-Claude	Spectroscopie
M. BOUQUELET Stéphane	Biologie Appliquée aux enzymes
M. BOUQUIN Henri	Gestion
M. BRASSELET Jean-Paul	Géométrie et Topologie
M. BRUYELLE Pierre	Géographie
M. CAPURON Alfred	Biologie Animale
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 Electroniques Optiques
M. COMYN Gérard	Informatique Théorique
M. COQUERY Jean-Marie	Psychophysiologie
M. CORIAT Benjamin	Sciences Economiques et Sociales
Mme CORSIN Paule	Paléontologie
M. CORTOIS Jean	Physique Nucléaire et Corpusculaire
M. COUTURIER Daniel	Chimie Organique
M. CRAMPON Norbert	Tectonique Géodynamique
M. CROSNIER Yves	Electronique
M. CURGY Jean-Jacques	Biologie
Melle DACHARRY Monique	Géographie
M. DEBRABANT Pierre	Géologie Appliquée
M. DEGAUQUE Pierre	Electronique
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
M. DENEL Jacques	Informatique
M. DE PARIS Jean Claude	Analyse
M. DEPREZ Gilbert	Physique du Solide - Cristallographie
M. DERIEUX Jean-Claude	Microbiologie
Melle DESSAUX Odile	Spectroscopie de la réactivité Chimique
M. DEVRAINNE Pierre	Chimie Minérale
Mme DHAINAUT Nicole	Biologie Animale
M. DHAMELINCOURT Paul	Chimie Physique
M. DORMARD Serge	Sciences Economiques
M. DUBOIS Henri	Spectroscopie Hertzienne
M. DUBRULLE Alain	Spectroscopie Hertzienne
M. DUBUS Jean-Paul	Spectrométrie des Solides
M. DUPONT Christophe	Vie de la firme (I.A.E.)
Mme EVRARD Micheline	Génie des procédés et réactions chimiques
M. FAKIR Sabah	Algèbre
M. FAUQUAMBERGUE Renaud	Composants électroniques

M. FONTAINE Hubert
 M. FOUQUART Yves
 M. FOURNET Bernard
 M. GAMBLIN André
 M. GLORIEUX Pierre
 M. GOBLOT Rémi
 M. GOSSELIN Gabriel
 M. GOUDMAND Pierre
 M. GOURIEROUX Christian
 M. GREGORY Pierre
 M. GREMY Jean-Paul
 M. GREVET Patrice
 M. GRIMBLOT Jean
 M. GUILBAULT Pierre
 M. HENRY Jean-Pierre
 M. HERMAN Maurice
 M. HOUDART René
 M. JACOB Gérard
 M. JACOB Pierre
 M. Jean Raymond
 M. JOFFRE Patrick
 M. JOURNEL Gérard
 M. KREMBEL Jean
 M. LANGRAND Claude
 M. LATTEUX Michel
 Mme LECLERCQ Ginette
 M. LEFEBVRE Jacques
 M. LEFEBVRE Christian
 Mlle LEGRAND Denise
 Mlle LEGRAND Solange
 M. LEGRAND Pierre
 Mme LEHMANN Josiane
 M. LEMAIRE Jean
 M. LE MAROIS Henri
 M. LEROY Yves
 M. LESENNE Jacques
 M. LHENAFF René
 M. LOCQUENEUX Robert
 M. LOSFELD Joseph
 M. LOUAGE Francis
 M. MAHIEU Jean-Marie
 M. MAIZIERES Christian
 M. MAURISSON Patrick
 M. MESMACQUE Gérard
 M. MESSELYN Jean
 M. MONTEL Marc
 M. MORCELLET Michel
 M. MORTREUX André
 Mme MOUNIER Yvonne
 Mme MOUYART-TASSIN Annie Françoise
 M. NICOLE Jacques
 M. NOTELET Francis
 M. PARSY Fernand

4

Dynamique des cristaux
 Optique atmosphérique
 Biochimie Sturcturale
 Géographie urbaine, industrielle et démog.
 Physique moléculaire et rayonnements Atmos.
 Algèbre
 Sociologie
 Chimie Physique
 Probabilités et Statistiques
 I.A.E.
 Sociologie
 Sciences Economiques
 Chimie Organique
 Physiologie animale
 Génie Mécanique
 Physique spatiale
 Physique atomique
 Informatique
 Probabilités et Statistiques
 Biologie des populations végétales
 Vie de la firme (I.A.E.)
 Spectroscopie hertzienne
 Biochimie
 Probabilités et statistiques
 Informatique
 Catalyse
 Physique
 Pétrologie
 Algèbre
 Algèbre
 Chimie
 Analyse
 Spectroscopie hertzienne
 Vie de la firme (I.A.E.)
 Composants électroniques
 Systèmes électroniques
 Géographie
 Physique théorique
 Informatique
 Electronique
 Optique-Physique atomique
 Automatique
 Sciences Economiques et Sociales
 Génie Mécanique
 Physique atomique et moléculaire
 Physique du solide
 Chimie Organique
 Chimie Organique
 Physiologie des structures contractiles
 Informatique
 Spectrochimie
 Systèmes électroniques
 Mécanique

M. PECQUE Marcel
M. PERROT Pierre
M. STEEN Jean-Pierre

5
Chimie organique
Chimie appliquée
Informatique

Je remercie Maurice Nivat de présider et de rapporter cette thèse.

Je remercie Danièle Beauquier d'avoir accepté d'être rapporteur.

Je remercie Max Dauchet, Michel Mériaux et Patrice Séébold de participer à ce jury.

Je remercie tout particulièrement Michel Latteux pour m'avoir encadrée et soutenue tout au long de ce travail. Je suis reconnaissante envers les différentes personnes avec lesquelles j'ai travaillé: leur collaboration m'a été très profitable et enrichissante.

Je remercie mes proches pour m'avoir encouragée à mener à terme mes études.

Je remercie Henri Glanc pour le tirage de cette thèse.

TABLE DES MATIERES

Table des matières

Introduction	1
Chapitre 1: Préliminaires	7
1.1 Notations	9
1.2 Notions de base	12
Chapitre 2: Figure tracée sans lever le crayon	19
2.1 Le système de réécriture S	21
2.1.1 Présentation du système S	22
2.1.2 Obtenir tous les mots décrivant la même figure	24
2.1.3 Obtenir un mot minimal	36
2.1.4 Parlons un peu des autres	40
2.1.5 Quelques mots concernant les figures de base	41
2.2 Le système de réécriture S_r -red	44
2.2.1 Historique	44
2.2.2 De fausses ressemblances	46
2.2.3 La preuve	48
Chapitre 3: Figure tracée avec possibilité de lever le crayon	55
3.1 Quelques définitions complémentaires	58
3.2 Le système de réécriture S'	60
3.2.1 Présentation du système S'	60
3.2.2 Le système S' est complet	63
3.2.3 Pas de système fini	65
3.2.4 Méthode de simplification	67
3.3 Le cas particulier des figures connexes	74
3.3.1 Quelques définitions	74
3.3.2 Une inégalité	74
3.3.3 Caractérisation d'un mot minimal de Π'^*	79
3.3.4 Relation entre la longueur d'un mot minimal et sa figure de base	83

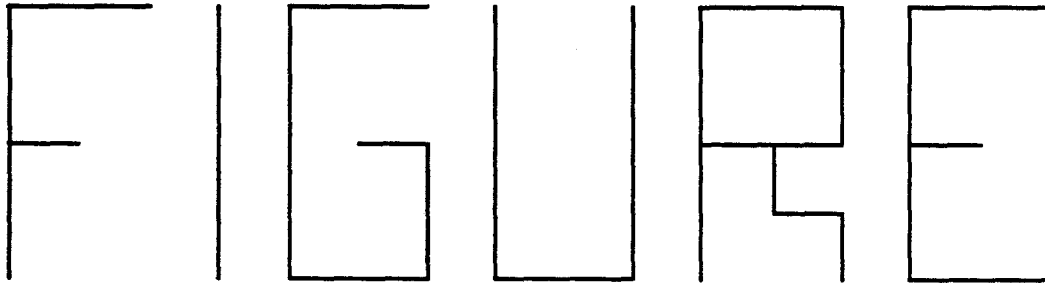
Chapitre 4: Figure particulière: Contour de polyomino	89
4.1 Définitions et propriétés	92
4.1.1 Les polyominos	92
4.1.2 Les mots de contour	94
4.2 Quel est le problème ?	98
4.3 La preuve	99
4.3.1 Idée de la preuve	99
4.3.2 Le système de réécriture R	99
4.3.3 Le système de réécriture T	103
4.4 Extension aux polyominos semi-convexes	110
4.4.1 Quelques définitions	110
4.4.2 Quel est le problème ?	113
4.4.3 La preuve	113
4.5 Autres extensions envisagées	119
Conclusion	121
Références	125

INTRODUCTION

Introduction

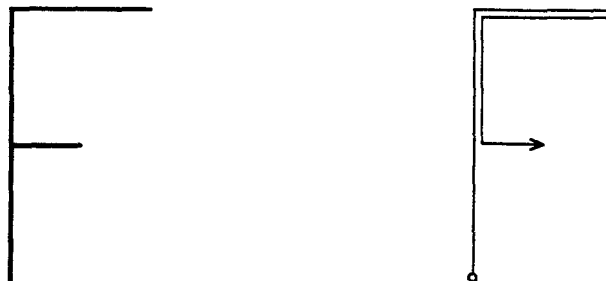
Une *figure* est un ensemble fini de segments unitaires horizontaux ou verticaux dont les extrémités coïncident avec les éléments du plan cartésien Z^2 . Elle peut être connexe ou ne pas l'être.

Exemple:



En s'inspirant du codage des figures introduit par H. Freeman [Fr] en 1961, chaque composante connexe peut être représentée par une séquence de symboles, chacun d'entre eux engendrant un segment unitaire tracé dans l'une des quatre directions: droite, gauche, haut ou bas. Si nous codons r (resp. l , u et d) l'action traçant un segment unitaire vers la droite (resp. gauche, haut et bas), le symbole "F" peut être décrit par la suite d'actions "uuurrllldr".

Exemple:



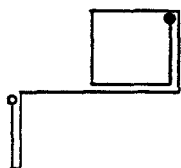
Cette approche a permis à H.A. Maurer, G. Rozenberg et E. Welzl [MRW] d'enrichir la théorie des langages d'un nouveau concept:

Les langages de mots de figure.

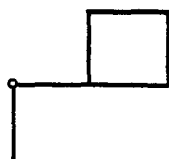
Un *mot de figure* est un mot défini sur l'alphabet $\Pi = \{ r, l, u, d \}$ où chaque lettre correspond à l'action élémentaire définie précédemment. Ainsi, nous pouvons associer à tout mot de figure w , un *unique tracé* obtenu en partant d'un point fixe (l'origine du repère notée \circ) et en exécutant les actions élémentaires dictées par la suite de lettres de ce mot. L'ensemble des segments parcourus au moins une fois par le tracé de w , caractérise *la figure* associée à w et si nous précisons le point où le tracé se termine (noté \bullet), nous parlerons de *figure décrite* par w .

Exemple:

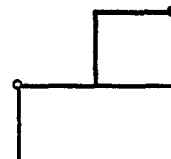
$w = \text{durruldr}$



Le tracé
de w



La figure
associée à w



La figure décrite
par w

Coder les figures par des mots est un modèle possible pour piloter le crayon d'un traceur. Nous pouvons très facilement modifier un mot (par un morphisme, par exemple) de façon à ce que la figure associée subisse certaines transformations: rotation, symétrie, affinité horizontale ou verticale, homothétie ...

Dans ce travail, nous nous sommes intéressés à différents systèmes de réécriture ayant tous un même principe: conserver certaines propriétés des figures associées au cours des dérivations d'un mot de figure par ce système.

. Le premier chapitre est consacré à la mise en place d'outils de travail: un rappel des notations couramment utilisées et des définitions du vocabulaire employé, propre aux langages de mots de figure.

. Les mots de Π^* représentent des figures connexes tracées "sans lever le crayon". Le second chapitre étudie deux systèmes de réécriture *conservant les figures décrites* dans ce contexte particulier. Ces travaux ont fait l'objet d'une collaboration avec Patrice Séebold [SS1, SS2, SS3].

Le premier système noté S , est *complet* dans le sens où il permet à partir de tout mot de Π^* de dériver tous les mots ayant la même figure décrite que le mot initial: il est en particulier possible d'obtenir tous les mots minimaux représentant la figure décrite de départ (un mot est *minimal* s'il n'existe pas de mot strictement plus court représentant la même figure décrite). Comme le système S contient des règles qui augmentent la longueur du mot, il n'est pas évident d'obtenir un mot minimal après un certain nombre d'étapes de dérivation si nous dérivons de manière quelconque: dans cette optique, nous avons établi une méthode pour obtenir un mot minimal. Celle-ci nous a amenés à distinguer des mots de figure particuliers, les mots réduits, permettant une meilleure utilisation de notre algorithme (un mot est *réduit* si son tracé parcourt au plus deux fois chaque segment). Des travaux récents et simultanés fournissent des résultats assez proches des nôtres: F.J. Brandenburg et J. Dassow [BD] ont présenté un système de réécriture permettant de dériver tout mot de Π^* en un mot réduit (mais pas en un mot minimal), puis ils ont décrit un algorithme de minimalisation relevant de la théorie des graphes et indépendant de tout système de réécriture; R. Gutbrod [Gu1, Gu2] a introduit un système de transformation et un algorithme pour la recherche d'un mot minimal similaires aux nôtres mais n'ayant pas la notion de mot réduit, il néglige certaines simplifications.

Ce système S permet de nombreuses manipulations sur les mots de figure. Cela nous donne de puissantes propriétés mais en contrepartie, le système S ne conserve pas la rationalité d'un langage. C'est pourquoi nous étudions maintenant un système de réécriture restreint $S_r\text{-red}$ permettant de réduire les mots de figure (mais pas forcément de les minimaliser) tout en conservant la rationalité du langage.

Le système $S_r\text{-red}$ est un système de réécriture introduit par F. Hinz [Hi] et dont le rôle est d'éliminer les aller-retours redondants d'un mot de figure (un *aller-retour redondant* est un mot de la forme $x\bar{x}$ avec $x \in \Pi^+$ et \bar{x} , le mot ayant un tracé inverse à celui de x). F. Hinz a montré qu'un tel système de réécriture conservait la rationalité d'un langage. Nous démontrons que ce système est confluent: le mot irréductible, dérivé d'un mot quelconque de Π^* , est unique (indépendant de l'ordre des dérivations).

Dans un troisième chapitre, nous élargissons notre façon de coder des figures en ajoutant à notre alphabet quatre lettres "blanches" permettant de se déplacer dans les quatre directions du plan (droite, gauche, haut et bas) d'une unité sans engendrer de trace visible. Nous pouvons ainsi représenter une figure non connexe par un mot de figure

défini sur ce nouvel alphabet Π' et améliorer la description minimale d'une figure connexe [S1].

Un système de réécriture S' est défini sur l'alphabet Π' : ce système conserve les figures décrites (l'ensemble des segments parcourus de manière visible et le point final du tracé) et permet de dériver à partir d'un mot quelconque, tous les mots représentant la même figure décrite. Nous énonçons également une méthode pour dériver tout mot de Π'^* en un mot plus court mais nous n'obtenons pas de description minimale dans le cas général: ceci est dû au fait que nous ne savons pas reconnaître un mot minimal de Π'^* .

Nous décidons alors de restreindre le problème aux figures connexes pour lesquelles Denis Robilliard [Ro] a trouvé une caractérisation d'un mot minimal de Π'^* . A partir de ce résultat, nous nous sommes intéressés aux différentes valeurs que pouvait prendre le rapport "longueur d'un mot minimal" / "nombre de segments de la figure". En particulier, nous nous penchons sur un problème posé par G. Paun [Pa]: Existe-t-il des figures connexes telles qu'en choisissant les "meilleurs" points de départ et d'arrivée pour son tracé, le rapport précédent soit toujours très proche de deux ?

De nombreuses personnes travaillent sur des problèmes de complexité et de décidabilité relatifs aux langages de mots de figure [BC, Bd, Da1, Da2, Da3, DH1, DH2, Hi1, Hi3, HW, Ki1, Ki2, KS1, KS2, SW]. Le dernier chapitre traite de la décidabilité de certaines questions.

Un mot de figure peut être également utilisé pour caractériser le contour d'un polyomino (un *polyomino* est une pièce sans trou composée d'un nombre fini de carrés unitaires juxtaposés). Le pavage du plan par une ou plusieurs pièces [GS] est un problème attrayant. Récemment, D. Beauquier et M. Nivat [BN] ont trouvé une condition simple sur les mots de contour d'un polyomino permettant de savoir si ce polyomino était apte à paver le plan. De là, des problèmes de décidabilité concernant les polyominos et leurs mots de contour ont été soulevés.

D. Beauquier [Bd] a démontré que nous ne pouvions pas décider si un langage rationnel $R \subset \Pi'^*$ contenait un mot de contour de polyomino. En limitant le problème à certaines classes de polyominos, nous montrons [BLS] que cette même question devient décidable. Ces résultats, concernant les polyominos convexes, horizontalement convexes et verticalement convexes, se déduisent de propriétés invariantes de deux systèmes de réécriture.

Chapitre 1:

PRELIMINAIRES

- 1 - Notations
- 2 - Notions de base

Chapitre 1:

Préliminaires

1.1 Notations

Nous supposons le lecteur familier avec la théorie des langages formels et nous précisons seulement quelques définitions. Si celles-ci lui semblent toutefois incomplètes, il peut se reporter aux ouvrages de M. Harrison [Ha], A. Salomaa [Sa], M. Jantzen [Ja] ou C. Berge [Bc].

Quelques notations:

On note N , l'ensemble des entiers naturels et N_+ , l'ensemble des entiers naturels non nuls. Si p et q sont deux éléments de N , $[p,q]$ est l'intervalle d'entiers défini par $[p,q]=\emptyset$ si $p>q$ et $[p,q]=\{p, p+1, \dots, q\}$ sinon. Z désigne l'ensemble des entiers relatifs.

A propos des langages:

Soit X , un ensemble fini appelé *alphabet*. Les éléments de X sont des *lettres* et les suites finies de lettres de X sont des *mots*. X^* désigne le monoïde libre engendré par l'alphabet X , ε le mot vide et $X^+=X^*-\{\varepsilon\}$. Si w et w' sont des mots de X^* , nous notons:

- . $|w|$, la *longueur* du mot w (en particulier, $|\varepsilon|=0$);
- . $|w|_a$, le *nombre d'occurrences* de la lettre a dans w ;
- . $F(w)=\{v \in X^* / \exists w_1, w_2 \in X^* w=w_1 v w_2\}$, l'ensemble des *facteurs* de w .
Un facteur est *propre* si $w_1 w_2 \neq \varepsilon$ et $w_1 w_2 \neq w$; $FG(w)=\{v \in X^* / \exists w' \in X^* w=v w'\}$, l'ensemble des *facteurs gauches* de w ;
 $FD(w)=\{v \in X^* / \exists w' \in X^* w=w' v\}$, l'ensemble des *facteurs droits* de w ;
- . $[w]=\{w_2 w_1 / w=w_1 w_2\}$, l'ensemble des *conjugés* du mot w .
- . $w \sqcup w'=\{u_1 v_1 u_2 v_2 \dots u_n v_n / \forall i \in [1,n], u_i, v_i \in X^*, w=u_1 u_2 \dots u_n \text{ et } w'=v_1 v_2 \dots v_n\}$, le *produit de mélange* (ou *shuffle*) des mots w et w' ;
- . $SM(w)=\{v \in X^* / \exists w' \in X^* w=v \sqcup w'\}$, l'ensemble des *sous-mots* de w .
- . Par extension, si L est un ensemble de mots (ou *langage*) de X^* et K , un opérateur défini sur X^* : $K(L) = \bigcup_{w \in L} K(w)$.

Soient L et L' , deux langages de X^* . Nous définissons:

- . $LL'=\{ww' / w \in L \text{ et } w' \in L'\}$, la *concaténation* de L et de L' ;
- . $L^0=\{\varepsilon\}$, $L^i=LL^{i-1}$ pour $i \geq 1$, $L^* = \bigcup_{i \geq 0} L^i$ et $L^+ = \bigcup_{i \geq 1} L^i$.

A propos des systèmes de réécriture:

Un système de réécriture σ sur un alphabet X est un sous-ensemble de $X^* \times X^*$. Chaque élément $(x,y) \in \sigma$ est une règle et peut être symbolisé par $x \rightarrow y$. Une étape de dérivation \Rightarrow_{σ} est la relation définie par:

$$(w \Rightarrow_{\sigma} w') \Leftrightarrow (\exists w_1, w_2, w'_1, w'_2 \in X^* / w = w_1 x w_2, w' = w'_1 y w'_2 \text{ et } (x,y) \in \sigma)$$

Dans certains cas, nous indiquons la règle de σ utilisée lors de la dérivation. Nous notons $\overset{*}{\Rightarrow}_{\sigma}$, la clôture transitive et réflexive de \Rightarrow_{σ} .

Si w et w' sont deux mots de X^* vérifiant $w \overset{*}{\Rightarrow}_{\sigma} w'$, nous disons que:

w se dérive en w' ou est un ascendant de w' par σ
 et w' est dérivé de w ou est un descendant de w par σ .

S'il n'y a pas d'ambiguïté sur le système de réécriture utilisé, nous pouvons omettre de le préciser et écrire $w \overset{*}{\Rightarrow} w'$.

Un système de réécriture σ sur un alphabet X est symétrique si pour tout descendant w' d'un mot w quelconque de X^* , w' se dérive en w :

$$(\sigma \text{ est symétrique}) \Leftrightarrow (\forall w, w' \in X^* (w \overset{*}{\Rightarrow}_{\sigma} w') \Rightarrow (w' \overset{*}{\Rightarrow}_{\sigma} w))$$

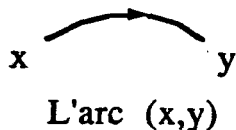
A propos des graphes:

Un graphe $G(X, \Gamma)$ est un couple constitué par un ensemble X et une relation Γ . Un élément de X est appelé un sommet du graphe et un couple $(x,y) \in X^2$ avec $y \in \Gamma(x)$ est appelé un arc du graphe. Par la suite, l'ensemble des arcs du graphe sera désigné par U .

Représentation: Les éléments de l'ensemble X sont symbolisés par des points du plan et si deux points x et y sont tels que $y \in \Gamma(x)$ (ou $(x,y) \in U$), un trait continu muni d'une flèche allant de x à y sera tracé.

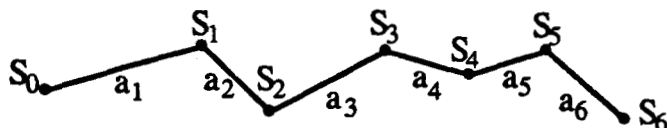
Dans ce travail, la notion d'orientation n'est pas nécessaire. Nous employerons plutôt le vocabulaire suivant: Dans un graphe $G(X,U)$, une arête est un ensemble de deux sommets x et y tels que $(x,y) \in U$ ou $(y,x) \in U$. L'arête qui a pour extrémités les sommets x et y se note $[x,y]$.

Exemple:



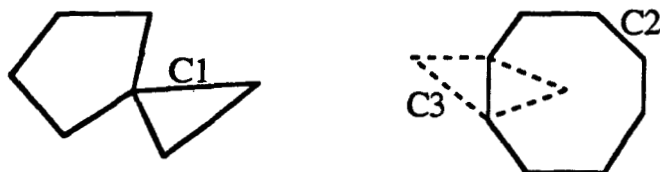
Une *chaîne* est une séquence d'arêtes (a_1, a_2, \dots, a_n) où chaque arête a_k ($k \in [2, n-1]$) est rattaché à a_{k-1} par une de ses extrémités, et à a_{k+1} par l'autre. A toute chaîne est associée une unique *suite de sommets* (S_0, S_1, \dots, S_n) telle que $a_k = [S_{k-1}, S_k]$, ($k \in [1, n]$). La *longueur* d'une chaîne est le nombre d'arêtes qui la composent.

Exemple:



Un *cycle* est une chaîne qui part d'un sommet x et aboutit au même sommet x . Un cycle est *simple* si les arêtes utilisées sont toutes différentes. Deux cycles simples sont *disjoints* s'ils n'ont aucune arête commune (mais ils peuvent avoir des sommets communs).

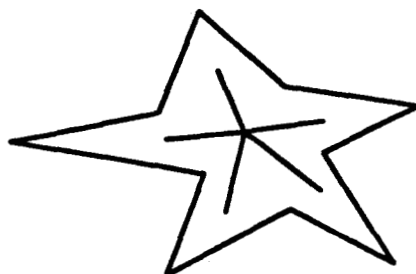
Exemple: C1, C2 et C3 sont des cycles simples disjoints.



Dans la suite, le terme "cycle" désignera implicitement un cycle simple.

Un graphe est *connexe* si pour tout couple de sommets distincts, il existe une chaîne allant de l'un à l'autre. Un sommet x étant donné, désignons par C_x l'ensemble des sommets pouvant être reliés à x par une chaîne, augmenté de x . Une *composante connexe* d'un graphe $G(X, U)$ est un sous-graphe de la forme (C_x, U_x) avec $x \in X$ et $U_x = U \cap C_x^2$.

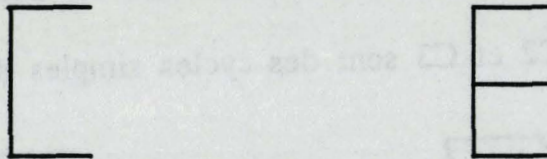
Exemple: Ce graphe n'est pas connexe. Il possède 2 composantes connexes.



1.2 Notions de base

Une figure est un ensemble fini de segments unitaires horizontaux ou verticaux dont les extrémités coïncident avec des éléments du plan cartésien Z^2 . Notre travail utilise la description (ou le codage) de telles figures par des mots.

En 1961, H. Freeman [Fr] a présenté un codage des figures connexes. Ce code est constitué de huit symboles, chacun d'entre eux engendrant un segment pointé vers le premier point de Z^2 rencontré dans l'une des directions: Nord, Nord-Est, Est, Sud-Est En s'inspirant de ce codage et en se limitant aux quatre déplacements horizontaux et verticaux, H.A. Maurer, G. Rozenberg et E. Welzl [MRW] ont formalisé la notion de *langages de mots de figure*.



Une méthode pour tracer le symbole qui représente la lettre "C" est de:

- . partir du coin supérieur droit;
- . tracer un trait vers la gauche;
- . tracer un trait deux fois plus long vers le bas;
- . puis tracer un trait vers la droite.

Maintenant pour tracer la lettre "E" à partir du symbole précédent, deux cas sont possibles:

- nous ne savons pas lever le crayon, il faut alors:
 - . tracer un trait vers la gauche, puis vers le haut en prenant soin de repasser exactement sur le tracé existant et de s'arrêter au milieu du trait vertical;
 - . tracer un trait vers la droite;
- ou nous savons lever le crayon, il suffit alors de:
 - . soulever le crayon et venir se positionner au milieu du trait vertical;
 - . tracer un trait vers la droite.

Pour qu'une machine puisse tracer des figures quelconques, il suffit de lui apprendre quelques actions élémentaires et de lui envoyer, par la suite, la séquence des actions à effectuer pour obtenir la figure désirée.

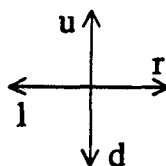
Ces actions élémentaires peuvent être:

- . tracer un segment unitaire vers la gauche (droite, haut et bas);
- . déplacer le crayon d'une unité (sans trace) vers la gauche (droite, haut et bas);
- ou
- . déplacer le crayon d'une unité vers la gauche (droite, haut et bas);
- . lever le crayon;
- . baisser le crayon;
- ou
- ...

Le codage que nous avons retenu est le premier.

De façon plus formelle, nous définissons les outils que nous utiliserons dans ce travail. Nous nous limitons, dans un premier temps, aux actions qui engendrent des segments unitaires visibles (le crayon baissé). Le déplacement du crayon en mode "levé" est tout à fait similaire et sera précisé ultérieurement.

- . Un *mot de figure* est un mot défini sur $\Pi = \{ r, l, u, d \}$, un alphabet de quatre lettres. Chaque lettre de Π correspond à l'action élémentaire suivante:



- r trace un segment unitaire vers la droite (right);
- l ou \bar{r} trace un segment unitaire vers la gauche (left);
- u trace un segment unitaire vers le haut (up);
- d ou \bar{u} trace un segment unitaire vers le bas (down).

Comme l symbolise une action opposée à celle engendrée par r, nous utiliserons plus souvent le symbole \bar{r} à la place de l (de même, \bar{u} à la place de d).

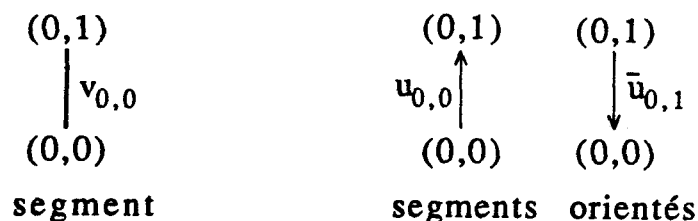
Nous pouvons définir un morphisme e (non injectif) qui associe à tout mot de Π^* , un couple d'entiers représentant le vecteur de déplacement dans \mathbb{Z}^2 :

$$\begin{array}{rcl}
 e: \Pi^* & \rightarrow & \mathbb{Z}^2 \\
 r & \rightarrow & (1, 0) \\
 \bar{r} & \rightarrow & (-1, 0) \\
 u & \rightarrow & (0, 1) \\
 \bar{u} & \rightarrow & (0, -1)
 \end{array}$$

- . Un *noeud* est un élément du plan cartésien \mathbb{Z}^2 .
- . Un *segment horizontal* $h_{i,j}$ (resp. *vertical* $v_{i,j}$) est la portion de droite comprise entre les noeuds N et N' de coordonnées (i,j) et $(i,j)+e(r)$ (resp. (i,j) et $(i,j)+e(u)$), noté $[N,N']$.
- . Une *figure* F est un ensemble fini de segments horizontaux et verticaux.
- . Un *segment orienté* $t_{i,j}$ est la portion de droite orientée du noeud N au noeud N' de coordonnées (i,j) et $(i,j)+e(t)$ ($t \in \Pi$), noté (N,N') .

Par conséquent, à chaque segment (horizontal ou vertical) $s_{i,j}$ correspondent deux segments orientés $t_{i,j}$ et $\bar{t}_{m,n}$ avec ($t=r$ si $s=h$ et $t=u$ si $s=v$) et $(m,n)=(i,j)+e(t)$.

Exemple:



Certaines analogies nous permettent de comparer une figure à un graphe: les segments de la figure et leurs extrémités seront respectivement les arêtes et les sommets du graphe associé. Sans difficulté, nous étendons les définitions de la théorie des graphes à la notion de figure.

Nous associons à tout mot w de Π^* , un *unique tracé* défini de la façon suivante: le point de départ est l'origine du plan de coordonnées $(0,0)$; le mot est lu, lettre par lettre, de gauche à droite; pour chaque lettre lue, nous traçons le segment orienté correspondant, le point d'arrivée servant de point de départ pour le tracé du segment suivant.

. Le *tracé du mot* w , $\text{tr}(w)$, est la suite des segments orientés parcourus de cette façon. Plus formellement, nous définissons l'opérateur tr sur Π^* ainsi:

- . $\text{tr}(\epsilon) = ()$ (suite vide) et
- . $\forall w \in \Pi^*, \forall a \in \Pi, \text{tr}(wa) = \text{tr}(w) + (N, N')$ avec les noeuds N et N' de coordonnées $e(w)$ et $e(wa)$; le symbole "+" indiquant que le segment orienté (N, N') est juxtaposé à la fin de la suite $\text{tr}(w)$.

Nous dirons qu'un segment s est *parcouru par le tracé de* w si au moins l'un des deux segments orientés correspondant à s se trouve dans la suite $\text{tr}(w)$.

. La *figure orientée associée au mot* w , $\vec{\text{pic}}(w)$, est l'ensemble des segments orientés parcourus lors du tracé de w . Plus formellement, nous définissons l'opérateur $\vec{\text{pic}}$ sur Π^* ainsi:

- . $\vec{\text{pic}}(\epsilon) = \emptyset$ et
- . $\forall w \in \Pi^*, \forall a \in \Pi, \vec{\text{pic}}(wa) = \vec{\text{pic}}(w) \cup \{ (N, N') \}$ avec les noeuds N et N' de coordonnées $e(w)$ et $e(wa)$;

. La *figure associée au mot* w , $\text{pic}(w)$ (picture), est l'ensemble des segments parcourus lors du tracé de w . Plus formellement, nous définissons l'opérateur pic sur Π^* ainsi:

- . $\text{pic}(\epsilon) = \emptyset$ et
- . $\forall w \in \Pi^*, \forall a \in \Pi, \text{pic}(wa) = \text{pic}(w) \cup \{ [N, N'] \}$ avec les noeuds N et N' de coordonnées $e(w)$ et $e(wa)$;

Remarque:

- . Le point d'arrivée du tracé du mot w est le point de coordonnées $e(w) = (|w|_r - |w|_f, |w|_u - |w|_d)$.
- . Une telle figure est dite tracée "sans lever le crayon": elle est nécessairement connexe.

Exemple: Le tracé et la figure du mot $w=urur\bar{u}\bar{u}$:
 (\circ désigne le point de départ et \bullet , le point d'arrivée)



avec $e(w)=(2,0)$ et

$$\text{tr}(w) = (u_{0,0}, r_{0,1}, u_{1,1}, r_{1,2}, \bar{u}_{2,2}, \bar{u}_{2,1})$$

$$\vec{\text{pic}}(w) = \{ u_{0,0}, r_{0,1}, u_{1,1}, r_{1,2}, \bar{u}_{2,2}, \bar{u}_{2,1} \}$$

$$\text{pic}(w) = \{ v_{0,0}, h_{0,1}, v_{1,1}, h_{1,2}, v_{2,1}, v_{2,0} \}$$

Nous nous intéresserons davantage à l'aspect d'une figure indépendamment de son tracé. Il peut être utile, dans certains cas, de spécifier le point d'arrivée de la figure (si on envisage, par exemple, de concaténer des figures). C'est pourquoi nous définissons deux nouvelles notions de figure: Pour tout mot w de Π^* ,

- La figure décrite par w , $\text{dpic}(w)$ (drawn picture), est caractérisée par le couple:

$$\text{dpic}(w) = (\text{pic}(w), e(w)).$$

- La figure de base de w , $\text{bpic}(w)$ (basic picture), est la classe d'équivalence de figures définie ainsi: si t_v est la translation de vecteur $v \in \mathbb{Z}^2$:

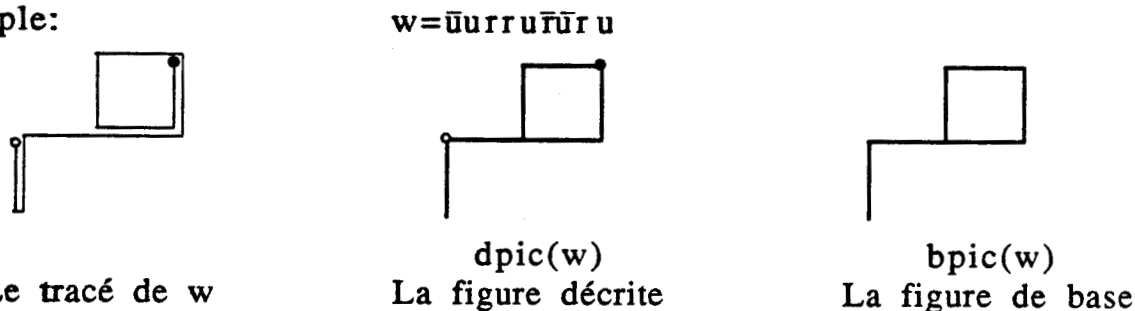
$$\text{bpic}(w) = \{ t_v(\text{pic}(w)) / v \in \mathbb{Z}^2 \}.$$

Remarque:

- Pour tous $w, w' \in \Pi^*$: $(\text{dpic}(w) = \text{dpic}(w')) \Rightarrow (\text{bpic}(w) = \text{bpic}(w'))$.
Naturellement, la réciproque est fausse.
- Dans les expressions suivantes, la définition implicite de figure sera la figure décrite. Nous dirons que le mot w décrit la figure $\text{dpic}(w)$, que le tracé de w est une *description* de la figure $\text{dpic}(w)$ et qu'un segment est *parcouru par un mot* w s'il est parcouru par son tracé.

Lors du tracé de $w \in \Pi^*$, il est possible que plusieurs passages s'effectuent sur le même segment. Ceux-ci seront représentés par des traces parallèles rapprochées mais ce segment n'apparaîtra qu'une seule fois dans $\text{pic}(w)$.

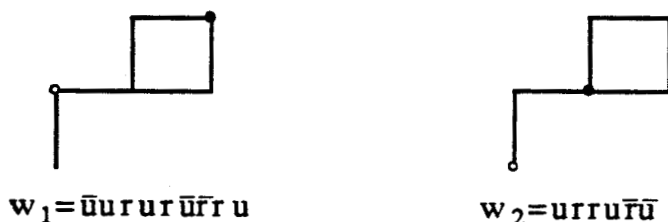
Exemple:



Notons qu'une figure donnée (non vide) est associée à une infinité de mots de Π^* . Par exemple, nous avons:

$$\text{dpic}(w) = \text{dpic}(\bar{u}urru\bar{r}\bar{u}ru) = \text{dpic}(\bar{u}urru(\bar{u}\bar{r}ru)^n), n > 0.$$

Exemple:



Les mots w_1 et w_2 ont la même figure de base mais les figures décrites sont différentes.

L'inverse d'un mot w de Π^* , $\text{inv}(w)$, est le mot ayant le même tracé que w mais "en sens inverse". Il est défini par:

$$\begin{aligned} \text{inv}(\epsilon) &= \epsilon & \text{inv}(r) &= \bar{r} & \text{inv}(\bar{r}) &= r & \text{inv}(u) &= \bar{u} & \text{inv}(\bar{u}) &= u \\ \text{et } \text{inv}(w) &= \text{inv}(a)\text{inv}(w') & \text{si } w &= w'a & \text{avec } w' \in \Pi^*, & a \in \Pi. \end{aligned}$$

Remarque:

$$\forall w_1, w_2 \in \Pi^*, \quad \text{inv}(w_1 w_2) = \text{inv}(w_2) \text{inv}(w_1).$$

Dans la suite, nous utiliserons la notation \bar{w} à la place de $\text{inv}(w)$.

Exemple:



Chapitre 2:

FIGURE TRACEE SANS LEVER LE CRAYON

- 1 - Le système de réécriture S
 - 1.1 Présentation du système S
 - 1.2 Obtenir tous les mots décrivant la même figure
 - 1.3 Obtenir un mot minimal
 - 1.4 Parlons un peu des autres
 - 1.5 Quelques mots concernant les figures de base
- 2 - Le système de réécriture Sr-red
 - 2.1 Historique
 - 2.2 De fausses ressemblances
 - 2.3 La preuve

Chapitre 2:

Figure tracée sans lever le crayon

Ce chapitre est consacré aux mots de figure définis sur l'alphabet $\Pi = \{ r, \bar{r}, u, \bar{u} \}$ dont les lettres n'engendrent que des déplacements visibles (sans lever le crayon).

Notre but est de pouvoir transformer (réécrire) un mot de figure donné en un mot plus court *tout en conservant la figure décrite*. Pour cela, nous nous sommes intéressés à deux systèmes de réécriture de $\Pi^* \times \Pi^*$ tels que la figure décrite soit conservée à chaque étape de dérivation. Ces systèmes sont étudiés dans les deux parties suivantes.

2.1 Le système de réécriture S

Ce système peut être considéré comme *complet* dans le sens où il permet, à partir de tout mot w de Π^* , de dériver *tous* les mots décrivant la même figure. En particulier, il est possible d'obtenir tous les mots réduits et tous les mots minimaux.

Les résultats que nous établissons dans cette section pour démontrer cette propriété du système S, nous permettent en outre, d'énoncer une méthode pour dériver un mot de figure quelconque en un *mot minimal*.

2.1.1 Présentation du système S

Le système S est caractérisé par 4 ensembles de règles de réécriture:

$$S = R1 \cup R2 \cup R2' \cup R3$$

avec

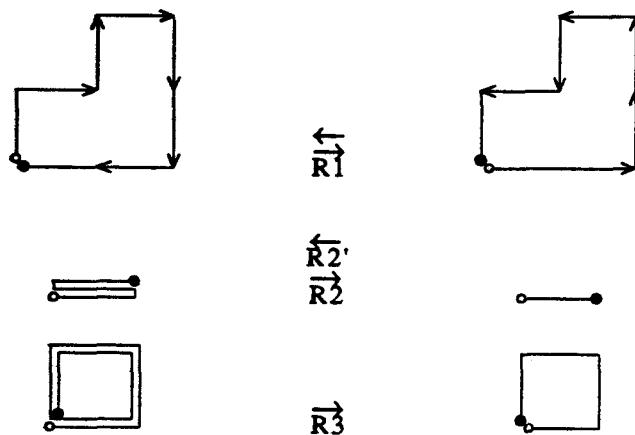
$$R1 = \{ b \rightarrow \bar{b} \ / \ b \in \mathbb{B} \}$$

$$R2 = \{ a\bar{a}a \rightarrow a \ / \ a \in \Pi \}$$

$$R2' = \{ a \rightarrow a\bar{a}a \ / \ a \in \Pi \}$$

$$R3 = \{ bb \rightarrow b \ / \ b \in \mathbb{B} \}$$

Illustrons ces transformations à l'aide d'exemples:



Remarques:

- Toutes les règles de S conservent les figures décrites: chaque règle de S peut être appliquée à un facteur d'un mot w de Π^* sans se préoccuper du contexte, le mot dérivé décrira la même figure que w .
- Le système S contient un nombre infini de règles: les ensembles $R1$ et $R3$ ne sont pas bornés. Nous montrerons par la suite qu'il n'existe pas de système de réécriture complet (permettant d'obtenir tous les mots décrivant la même figure) ayant un nombre fini de règles.
- L'ensemble des mots dérivés de $w \in \Pi^*$ par S, $Im_S(w)$, contient une infinité de mots: les règles de $R2'$ permettent d'augmenter indéfiniment la longueur des mots dérivés. Ceci est cohérent avec ce que nous annonçons dans le chapitre précédent: une figure donnée (non vide) est décrite par une infinité de mots de Π^* .

Notation:

- Dans la suite, l'expression "la règle $R1$ " signifiera "une règle de l'ensemble de règles $R1$ ".

Les règles suivantes peuvent être obtenues en composant des règles de S:

$$R4 = \{ bb' \rightarrow b'b \ / b, b' \in \mathbb{B} \}$$

$$R5 = \{ x\bar{x}x \rightarrow x \ / x \in \Pi^+ \}$$

$$R6 = \{ x\bar{b}x \rightarrow b'xb \ / b, b' \in \mathbb{B}, x \in \Pi^+ \}$$

$$R7 = \{ b \rightarrow bb \ / b \in \mathbb{B} \}$$

Preuve:

$$R4: bb' \xRightarrow{R1} \bar{b}\bar{b}' = \bar{b}'\bar{b} \xRightarrow{R1} b'b \xRightarrow{R1} b'b.$$

R5: Démonstration par récurrence sur $|x|$.

Si $|x|=1$, il s'agit de la règle R2;

Sinon, supposons que $x\bar{x}x \stackrel{*}{\Rightarrow} x$ pour $x \in \Pi^+$, $|x| \leq n$, $n > 0$.

Soit $x \in \Pi^+$ tel que $|x|=n+1$. Posons $x=ya$ avec $|y|=n$ et $a \in \Pi$. Nous avons $x\bar{x}x=ya\bar{y}a$ avec $\bar{y}y \in \mathbb{B}$, et $a\bar{a} \in \mathbb{B}$. Nous pouvons en déduire:

$$x\bar{x}x=ya\bar{y}a \xRightarrow{R4} y\bar{y}y a\bar{a} \xRightarrow{R2} y\bar{y}y \stackrel{*}{\Rightarrow} ya=x.$$

R6: Nous avons $\bar{x}b'x \in \mathbb{B}$ et $x\bar{x} \in \mathbb{B}$, par conséquent:

$$x\bar{x}b'x \xRightarrow{R4} x\bar{x}b'xb \xRightarrow{R4} b'x\bar{x}xb \xRightarrow{R5} b'xb.$$

R7: Soit $b \in \mathbb{B}$. Supposons $|b|=n$ et posons $b=a_1a_2 \dots a_n$. Nous définissons les suites de mots b_i et b'_i de la façon suivante: $b_i=a_1a_2 \dots a_i$ et $b'_i=a_{i+1}a_{i+2} \dots a_n$ avec $i \in [1, n]$.

Nous avons $b_i\bar{b}_i b_i b'_i \stackrel{*}{\Rightarrow} b_{i+1}\bar{b}_{i+1} b_{i+1} b'_{i+1}$ avec $i \in [1, n-1]$: en effet,

$$b_i\bar{b}_i b_i b'_i = b_i\bar{b}_i b_i a_{i+1} b'_{i+1} \xRightarrow{R2} b_i\bar{b}_i b_i a_{i+1} \bar{a}_{i+1} a_{i+1} b'_{i+1}$$

$$b_i\bar{b}_i b_i a_{i+1} \bar{a}_{i+1} a_{i+1} b'_{i+1} \xRightarrow{R4} b_i a_{i+1} \bar{a}_{i+1} \bar{b}_i b_i a_{i+1} b'_{i+1} = b_{i+1} \bar{b}_{i+1} b_{i+1} b'_{i+1}.$$

Par conséquent, $b_1\bar{b}_1 b_1 b'_1 \stackrel{*}{\Rightarrow} b_n\bar{b}_n b_n b'_n = b_n\bar{b}_n b_n$.

Nous pouvons en déduire:

$$b \xRightarrow{R2} b_1\bar{b}_1 b_1 b'_1 \stackrel{*}{\Rightarrow} b_n\bar{b}_n b_n = b\bar{b}b \xRightarrow{R1} bbb \xRightarrow{R3} bb.$$

Remarque:

Le système S est *symétrique*: toute règle de S a son symétrique dans S ou dans les règles déduites de S (R1 est symétrique, R2 et R2' sont symétriques l'une de l'autre, R3 et R7 aussi).

2.1.2 Obtenir tous les mots décrivant la même figure

Nous montrons que le système S est *complet*: il permet, à partir d'un mot quelconque de Π^* , de dériver tous les mots décrivant la même figure.

Théorème 2.1.1:

Pour tout mot $w \in \Pi^$, l'ensemble des mots dérivés de w par S est l'ensemble des mots de Π^* décrivant la même figure que w .*

$$Im_s(w) = \{ v \in \Pi^* \mid dpic(v) = dpic(w) \}$$

Pour prouver ce théorème, nous établissons les résultats intermédiaires suivants:

Lemme 2.1.2:

Soit $w \in \Pi^$. Pour tout segment $s \in pic(w)$ tel que $mult_w(s) \geq 3$, il existe un mot w' , dérivé de w par S , vérifiant $|w'| = |w| - 2$ tel que:*

- . $mult_{w'}(s) = mult_w(s) - 2$;*
- . $mult_{w'}(s') = mult_w(s')$ pour tout $s' \in pic(w) - \{s\}$.*

Preuve:

Soient $w \in \Pi^*$ et $s \in pic(w)$, un segment tel que $mult_w(s) \geq 3$: cela signifie qu'il existe au moins trois passages sur le segment s lors du tracé de w .

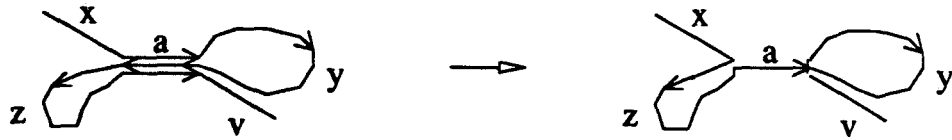
Posons $w = xayaza''v$ avec $x, y, z, v \in \Pi^*$, $a \in \Pi$ et $a', a'' \in \{a, \bar{a}\}$ où a , a' et a'' représentent dans cet ordre, les trois premiers passages sur s . Quatre cas sont possibles en fonction des valeurs de a' et de a'' .

Le but est de modifier le sens de parcours de certaines portions de façon à faire apparaître trois passages consécutifs sur s . Il suffit alors d'appliquer la règle R2 pour obtenir le résultat désiré.

Cas 1: $a' = \bar{a}$ et $a'' = a$.

Nous obtenons $w = xay\bar{a}zav$ avec $y, z \in \mathbb{B}$. En notant que $a\bar{a} \in \mathbb{B}$ et $\bar{a}za \in \mathbb{B}$, nous dérivons w de la façon suivante:

$$w = xay\bar{a}zav \xRightarrow{R4} xa\bar{a}zayv \xRightarrow{R4} xza\bar{a}ayv \xRightarrow{R2} xzayv = w'$$



La règle R4 se déduit uniquement de la règle R1, qui n'affecte pas la multiplicité des segments. Puisque la règle R2 est utilisée une fois, seule la multiplicité du segment s diminue de 2.

Cas 2: $a'=\bar{a}$ et $a''=\bar{a}$.

Nous obtenons $w=xay\bar{a}z\bar{a}v$ avec $y, z\bar{a}\in\mathbb{B}$. Nous pouvons nous reporter au cas précédent après avoir dérivé w ainsi:

$$w=xay\bar{a}z\bar{a}v \xRightarrow{R_1} xay\bar{a}z\bar{a}v$$

Cas 3&4: $a'=a$ et $a''\in\{a,\bar{a}\}$.

Nous obtenons $w=xayaza''v$ avec $ya\in\mathbb{B}$. Nous pouvons nous reporter à l'un des deux cas précédents après avoir dérivé w ainsi:

$$w=xayaza''v \xRightarrow{R_1} xa\bar{a}yza''v \quad \blacklozenge$$

Les corollaires suivants sont des conséquences directes de ce lemme:

Corollaire 2.1.3:

Tout mot minimal est réduit.

Preuve:

Supposons que w soit un mot minimal non réduit: il existe un segment $s\in\text{pic}(w)$ tel que $\text{mult}_w(s)\geq 3$. Appliquons le lemme 2.1.2, nous obtenons un mot strictement plus court que w décrivant la même figure. Contradiction. \blacklozenge

Corollaire 2.1.4:

Pour tout mot $w\in\Pi^$, il existe un mot réduit p , dérivé de w par S .*

Preuve:

Soit $w\in\Pi^*$. Tant qu'il existe un segment de multiplicité supérieure ou égale à 3, nous appliquons le lemme 2.1.2. Nous obtenons ainsi un mot p (décrivant la même figure que w) parcourant chaque segment au plus deux fois: p est réduit. \blacklozenge

Le résultat suivant n'intervient pas dans la preuve du théorème. Nous ouvrons cependant une parenthèse car les notions introduites dans cette section nous permettent une nouvelle approche de la preuve de la propriété suivante: *Un graphe connexe peut être entièrement parcouru en passant au plus une fois dans chaque sens sur chaque arête.* Nous nous limitons ici à des graphes particuliers: les arêtes doivent être horizontales ou verticales et de longueur 1 (l'extension à des graphes quelconques se conçoit facilement en grossissant l'alphabet de base). Un tel graphe connexe peut être entièrement parcouru par un mot $w \in \Pi^*$. Comme nous savons dériver, par S, ce mot en un mot réduit p [corollaire 2.1.4], la proposition suivante justifie cette propriété.

Proposition 2.1.5:

Pour tout mot réduit $p \in \Pi^$, il existe un mot q , dérivé de p par R1, vérifiant: $\forall t \in \vec{\Gamma}(q), \text{mult}_q(t) \leq 1$.*

Preuve:

La preuve s'effectue par récurrence sur $n = |p|$. Nous construisons une suite finie (p_j) de mots de Π^* vérifiant (*):

- . $p_0 = p$;
- . $\forall j \in [0, n], p_j = q_j q'_j$ avec $|q_j| = j$ et $\forall t \in \vec{\Gamma}(q_j), \text{mult}_{p_j}(t) \leq 1$;
- . $\forall j \in [1, n], p_{j-1} \xrightarrow[R1]{*} p_j$;

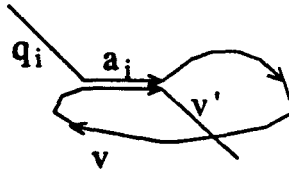
de la façon suivante:

- . Posons $p_0 = p$: ce mot vérifie bien (*);
- . Supposons que p_i soit un mot de cette suite (avec $i \in [0, n-1]$) et construisons p_{i+1} . Soit a_i , la première lettre de q'_i (comme p_i est dérivé de p_0 par R1, $|p_i| = |p_0| = n$ et $|q'_i| > 0$).
 - Si a_i engendre le premier passage sur un segment (non orienté) de multiplicité égale à 1 relativement à p_i , ou le deuxième passage sur un segment, nous posons $p_{i+1} = p_i$.
 - Si a_i engendre le premier passage sur un segment de multiplicité égale à 2 relativement à p_i , nous dérivons p_i de façon à ce que le second passage sur ce segment s'effectue nécessairement en sens inverse du premier passage.

Nous notons s_i (resp. t_i) le segment (resp. orienté) engendré par a_i .

a) $\text{mult}_w(t_i)=2$: les deux passages s'effectuent dans le même sens.

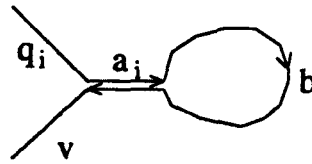
Nous avons $p_i=q_i a_i v a_i v'$ avec $q_i, v' \in \Pi^*$ et $v a_i \in \mathbb{B}$.



Dérivons p_i de la façon suivante: $p_i=q_i a_i v a_i v' \xRightarrow{R_1} p_{i+1}=q_i a_i \bar{a}_i \bar{v} v'$.

b) $\text{mult}_w(t_i)=1$: les deux passages s'effectuent en sens inverse.

Nous avons $p_i=q_i a_i b \bar{a}_i v$ avec $q_i, v \in \Pi^*$ et $b \in \mathbb{B}$.



Posons $v'=b \bar{a}_i v$. Deux possibilités:

. Soit il existe une boucle dans v' contenant la lettre \bar{a}_i .

Nous avons $v'=b_1 b_2 \bar{a}_i v_1 v_2$ avec $b=b_1 b_2$, $v=v_1 v_2$, et $b_2 \bar{a}_i v_1 \in \mathbb{B}$.

Dérivons p_i de la façon suivante:

$$p_i=q_i a_i b_1 b_2 \bar{a}_i v_1 v_2 \xRightarrow{R_1} p_i'=v a_i b_1 \bar{v}_1 a_i \bar{b}_2 v_2$$

de façon à ce que les deux passages sur s_i s'effectuent dans le même sens. Il suffit maintenant de se reporter au cas précédent pour obtenir les deux passages sur s_i consécutifs et de sens opposés.

. Soit il n'existe aucune boucle dans v' contenant la lettre \bar{a}_i : dans ce cas, il sera impossible de faire apparaître a_i pour décrire le deuxième passage sur s_i dans les prochains mots de la suite (p_j) .

Posons $p_{i+1}=p_i$.

Nous vérifions facilement que, dans tous les cas, p_{i+1} satisfait (*).

fin de la construction de la suite (p_j) .

En posant $q=p_n$, nous obtenons le résultat désiré. ♦

Pour tout mot w de Π^* ,

$P(w)$ est le *polynôme de segments* associé au mot w , défini par:

$$P(w) = \sum_{s \in \text{pic}(w)} \text{mult}_w(s) \cdot s$$

A chaque segment de la figure est associée sa multiplicité.

Lemme 2.1.6:

Deux mots de Π^ , q et q' , ont le même polynôme de segments si et seulement si q se dérive en q' par $R1$.*

Preuve:

Puisque $R1$ ne modifie pas la multiplicité des segments, nous avons:

$$(q \xrightarrow{*}_{R1} q') \Rightarrow (P(q)=P(q')).$$

La réciproque se démontre par récurrence sur $n=|q|=|q'|$ (comme q et q' ont le même polynôme de segments, ils contiennent nécessairement le même nombre de lettres). Montrons que:

$$(P(q)=P(q')) \Rightarrow (q \xrightarrow{*}_{R1} q').$$

- Si $n=0$ ou $n=1$, nous avons: $(P(q)=P(q')) \Rightarrow (q=q') \Rightarrow (q \xrightarrow{*}_{R1} q')$.

- Supposons la propriété vraie pour tout $i \in \mathbb{N}$, $i \geq 1$ et montrons qu'elle est encore vérifiée pour $n=i+1$.

Deux cas sont possibles:

. Si q et q' ont la même première lettre: posons $q=aq_1$ et $q'=aq'_1$ avec $a \in \Pi$ et $q_1, q'_1 \in \Pi^*$. D'après l'hypothèse de récurrence, nous avons:

$$q=aq_1 \xrightarrow{*}_{R1} q'=aq'_1.$$

. Sinon, posons $q=aq_1$ et $q'=a'q'_1$ avec $a, a' \in \Pi$, $a \neq a'$ et $q_1, q'_1 \in \Pi^*$. Notons s , le segment engendré par a' lors du tracé de q' . Ce segment est également parcouru par q : soit $q=v_1cv_2$, $c \in \Pi$ où c engendre le premier passage sur s . Puisque l'une des extrémités de s est le point de départ de la figure, nous avons: soit v_1 est une boucle, soit v_1c est une boucle.

- Si v_1 est une boucle alors $c=a'$ et il existe une boucle contenant c (dans q). Posons $q=u_1u_2a'u_3u_4$ avec $u_1, u_2, u_3, u_4 \in \Pi^*$ tels que $v_1=u_1u_2 \in \mathbb{B}$, $v_2=u_3u_4$ et $u_2a'u_3 \in \mathbb{B}$. En particulier, nous avons $u_2a'u_3 \in \mathbb{B}$ et $u_1\bar{u}_3\bar{a}' \in \mathbb{B}$ (en effet, $e(u_1\bar{u}_3\bar{a}') = e(u_1u_2\bar{u}_3\bar{a}'\bar{u}_2) = e(u_1u_2) + e(\bar{u}_3\bar{a}'\bar{u}_2) = (0,0)$). Dérivons q de la façon suivante:

$$q = u_1u_2a'u_3u_4 \xRightarrow{R_1} u_1\bar{u}_3\bar{a}'\bar{u}_2u_4 \xRightarrow{R_1} a'u_3\bar{u}_1\bar{u}_2u_4 = q''.$$

- Si v_1c est une boucle alors $c=\bar{a}'$ et nous dérivons q de la façon suivante:

$$q = v_1\bar{a}'v_2 \xRightarrow{R_1} a'\bar{v}_1v_2 = q''.$$

Dans ces deux cas, nous pouvons dériver q en $a'q_0$ par R_1 , $q_0 \in \Pi^*$, et par hypothèse de récurrence, nous avons:

$$q \xRightarrow{R_1} a'q_0 \xRightarrow{R_1} a'q'_1 = q'. \blacklozenge$$

Corollaire 2.1.7:

Si deux mots de Π^ , q et q' , ont le même polynôme de segments alors $e(q) = e(q')$.*

Remarque: Dire que q et q' ont le même polynôme de segments signifie qu'en partant d'un point fixé, l'origine du repère, les tracés de q et de q' parcourent les mêmes segments, le même nombre de fois. A priori, aucune information n'est fournie sur leur point d'arrivée.

preuve:

Soient $q, q' \in \Pi^*$ tels que $P(q) = P(q')$. D'après le lemme 2.1.6, q se dérive en q' par R_1 . Puisque R_1 conserve les figures décrites, nous avons $dpic(q) = dpic(q')$ et par conséquent $e(q) = e(q')$. \blacklozenge

Dans ce qui suit, une figure est assimilée à un graphe: les arêtes et les sommets sont les segments et les noeuds de la figure. Nous introduisons quelques définitions supplémentaires:

- . Un mot $w \in \Pi^*$ décrit une chaîne C si pour tout facteur gauche w' de w , nous avons $N_0 + e(w') = N_k$ avec $k = |w'|$ et (N_0, N_1, \dots, N_n) , la suite de noeuds associée à C .
 - Etant donné une chaîne C , le mot w décrivant cette chaîne est unique.
 - Si C est un cycle, alors $w \in B$.
 - Si C est un cycle simple, alors $\forall s \in \text{pic}(w) \quad \text{mult}_w(s) = 1$.
- . Le degré d'un noeud N de coordonnées (x, y) relatif à un mot $w \in \Pi^*$, $d_w(N)$, est la somme des multiplicités des segments de $\text{pic}(w)$ ayant N comme extrémité: c'est le nombre d'occurrences des segments ayant N comme extrémité dans la suite $\text{tr}(w)$.

$$d_w(N) = \sum_{s \in \text{seg}_N} \text{mult}_w(s) \quad \text{avec } \text{seg}_N = \{ h_{x,y}, v_{x,y}, h_{x-1,y}, v_{x,y-1} \}$$

Notons que, lors du tracé de w :

- Pour tout noeud N (exceptés les points de départ et d'arrivée), le tracé arrive en N autant de fois qu'il en repart. Par conséquent, $d_w(N)$ est pair;
- Si le point de départ est différent du point d'arrivée, les degrés de ces noeuds sont impairs sinon ils sont pairs.

D'une manière générale, étant donné une figure décrite, la parité du degré de chaque noeud est parfaitement déterminée.

- . Pour tous mots $p, q \in \Pi^*$, $\text{smd}(p, q)$ est l'ensemble des segments de $\text{pic}(p) \cup \text{pic}(q)$ dont les multiplicités relatives à p et à q sont différentes.
- . Une chaîne C est issue d'un ensemble de segments E si elle est constituée de segments de E . Un segment appartient à C s'il appartient à la suite de segments caractérisant C .

Lemme 2.1.8:

Soient p et q , deux mots réduits de Π^ décrivant la même figure. Nous pouvons regrouper les segments de $\text{smd}(p,q)$ en cycles, disjoints deux à deux.*

Preuve:

Considérons le graphe composé des segments de $\text{smd}(p,q)$. Il est à noter que *tout sommet de ce graphe est l'extrémité d'un nombre pair d'arêtes*: en effet, les mots p et q étant réduits, les multiplicités relatives à p et à q d'un segment de $\text{smd}(p,q)$ diffèrent d'une unité. Comme en chaque noeud N de la figure initiale, la parité du degré doit être conservée, il existe un nombre pair de segments de $\text{smd}(p,q)$ contenant N .

D'après le théorème d'Euler [Bc], chaque composante connexe de ce graphe admet un cycle eulérien (qui utilise chaque arête exactement une fois). De cette façon, nous pouvons regrouper les segments de $\text{smd}(p,q)$ en cycles, disjoints deux à deux.♦

Lemme 2.1.9:

Pour tout mot réduit $p \in \Pi^$ et pour tout cycle C issu de $\text{pic}(p)$, il existe un mot réduit q , dérivé de p par S , vérifiant:*

$$\begin{aligned} \forall s \in C, \text{mult}_q(s) &= (\text{mult}_p(s) \bmod 2) + 1 \\ \forall s \notin C, \text{mult}_q(s) &= \text{mult}_p(s). \end{aligned}$$

Preuve:

Soient p , un mot réduit de Π^* et C , un cycle issu de $\text{pic}(p)$.

Le but est de dériver p de façon à faire apparaître un facteur décrivant exactement le cycle C : la multiplicité des segments de C augmente d'une unité et pour les segments dont la multiplicité atteint trois, nous appliquons le lemme 2.1.2 pour la réduire à un. Ce procédé nous permet de dériver p en un mot pour lequel la multiplicité des segments de C est inversée (deux en un, et vice versa) et la multiplicité des autres segments inchangée.

Posons $C=(s_1, s_2, \dots, s_n)$ et (N_0, N_1, \dots, N_n) , la suite de noeuds associée au cycle C . Décomposons le mot p de la façon suivante:

$p=x_1a_1x_2a_2\dots x_na_nx_{n+1}$ avec $\forall i \in [1, n+1]$, $x_i \in \Pi^*$ et $\forall i \in [1, n]$, $a_i \in \Pi$, tel que a_i engendre le premier passage sur un segment de C (tous les segments de C sont ainsi concernés).

Supposons que s_1 soit engendré pour la première fois par a_k , $k \in [1, n]$.

Nous allons construire une suite finie (p_j) de mots de Π^* vérifiant (*):

- $p_0=p$;
- $\forall j \in [1, n]$, $p_j=x_1a_1x_2a_2\dots x_ky_j\bar{y}_ja_k\dots x_na_nx_{n+1}$ avec $y_j=a_{i_1}a_{i_2}\dots a_{i_j}$ décrit la chaîne (s_1, s_2, \dots, s_j) ;
- $\forall j \in [0, n-1]$, $p_j \xrightarrow{*} p_{j+1}$;

Cette suite est définie récursivement par:

- $p_0=p$: ce mot vérifie bien (*);
- $\forall j \in [0, n-1]$, p_j se dérive en p_{j+1} de la façon suivante:

Par définition, il existe $m \in [1, n] - \{i_1, i_2, \dots, i_j\}$ tel que a_m engendre le premier passage sur s_{j+1} . Supposons $m < k$ (le raisonnement est semblable si $m > k$).

$$p_j = x_1a_1x_2a_2\dots x_ma_mx_{m+1}\dots x_ky_j\bar{y}_ja_k\dots x_{n+1}.$$

Deux cas sont possibles:

- Soit a_m décrit s_{j+1} de N_j vers N_{j+1} :

$$\begin{array}{ccc} s_j & \xrightarrow{\quad} & s_{j+1} \\ & \searrow & \nearrow \\ & N_j & N_{j+1} \end{array}$$

$$e(x_1a_1x_2a_2\dots x_m) = e(x_1a_1x_2a_2\dots x_ma_mx_{m+1}\dots x_ky_j) = (x_{N_j}, y_{N_j})$$

(x_{N_j}, y_{N_j}) représentent les coordonnées de N_j .

Nous obtenons $a_mx_{m+1}\dots x_ky_j \in \mathbb{B}$ et nous posons $a_{ij+1} = a_m$.

- Soit a_m décrit s_{j+1} de N_{j+1} vers N_j :

$$\begin{array}{ccc} s_j & \xleftarrow{\quad} & s_{j+1} \\ & \swarrow & \searrow \\ & N_j & N_{j+1} \end{array}$$

$$e(x_1a_1x_2a_2\dots x_ma_m) = e(x_1a_1x_2a_2\dots x_ma_mx_{m+1}\dots x_ky_j) = (x_{N_j}, y_{N_j})$$

Nous obtenons $x_{m+1}\dots x_ky_j \in \mathbb{B}$ et nous posons $a_{ij+1} = \bar{a}_m$.

Dans les deux cas, $y_j a_{ij+1} = a_{i1} a_{i2} \dots a_{ij+1}$ décrit la chaîne $(s_1, s_2, \dots, s_{j+1})$ et nous dérivons p_j en p_{j+1} de la façon suivante:

$$\begin{aligned} p_j = x_1 \dots x_m a_m x_{m+1} \dots y_j \bar{y}_j \dots x_{n+1} &\xrightarrow{\text{R2}} x_1 \dots x_m a_m \bar{a}_m a_m x_{m+1} \dots y_j \bar{y}_j \dots x_{n+1} \\ x_1 \dots x_m a_m \bar{a}_m a_m x_{m+1} \dots y_j \bar{y}_j \dots x_{n+1} &\xrightarrow{\text{R4}} x_1 \dots x_m a_m x_{m+1} \dots y_j a_{ij+1} \bar{a}_{ij+1} \bar{y}_j \dots x_{n+1} = p_{j+1}. \end{aligned}$$

Nous vérifions facilement que p_{j+1} satisfait (*).

fin de la construction de la suite (p_j) .

En particulier, nous obtenons: $p \xrightarrow{*} p_n = x_1 a_1 x_2 a_2 \dots x_k y_n \bar{y}_n a_k x_{k+1} \dots a_n x_{n+1}$ avec y_n décrit la chaîne (s_1, s_2, \dots, s_n) .

Comme (s_1, s_2, \dots, s_n) est un cycle, $y_n \in \mathbb{B}$. Nous pouvons dériver p_n ainsi:

$$\begin{aligned} p_n = x_1 a_1 x_2 a_2 \dots x_k y_n \bar{y}_n a_k x_{k+1} \dots a_n x_{n+1} &\xrightarrow{\text{R1}} x_1 a_1 x_2 a_2 \dots x_k y_n y_n a_k x_{k+1} \dots a_n x_{n+1} \\ x_1 a_1 x_2 a_2 \dots x_k y_n y_n a_k x_{k+1} \dots a_n x_{n+1} &\xrightarrow{\text{R3}} x_1 a_1 x_2 a_2 \dots x_k y_n a_k x_{k+1} \dots a_n x_{n+1} = p'. \end{aligned}$$

Par construction:

$$\forall s \in C \quad \text{mult}_{p'}(s) = \text{mult}_p(s) + 1$$

$$\forall s \notin C \quad \text{mult}_{p'}(s) = \text{mult}_p(s).$$

Dérivons maintenant p' : pour chaque segment $s \in C$ tel que $\text{mult}_{p'}(s) = 3$, nous appliquons le lemme 2.1.2. Nous obtenons le mot q vérifiant:

$$\begin{aligned} \forall s \in C \quad &\text{si } \text{mult}_p(s) = 1 \text{ alors } \text{mult}_q(s) = 2 \\ &\text{si } \text{mult}_p(s) = 2 \text{ alors } \text{mult}_q(s) = 1; \end{aligned}$$

$$\forall s \notin C \quad \text{mult}_p(s) = \text{mult}_q(s). \blacklozenge$$

Corollaire 2.1.10:

Si deux mots réduits de Π^ , p et q , décrivent la même figure alors p se dérive en q par S .*

Preuve:

- Si p et q sont deux mots réduits décrivant la même figure, nous pouvons regrouper les segments de $\text{smd}(p, q)$ en cycles, disjoints deux à deux [lemme 2.1.8].

- En appliquant le lemme 2.1.9 au mot p pour chacun de ces cycles, nous obtenons un mot p' , dérivé de p , vérifiant:

$$\forall s \in \text{smd}(p,q), \quad \text{mult}_{p'}(s) = (\text{mult}_p(s) \bmod 2) + 1$$

$$\forall s \notin \text{smd}(p,q), \quad \text{mult}_{p'}(s) = \text{mult}_p(s).$$

Par conséquent:

$$\forall s \in \text{pic}(q), \quad \text{mult}_{p'}(s) = \text{mult}_q(s).$$

- D'après le lemme 2.1.6, q peut être dérivé de p' par R1.

Nous en déduisons que: $p \stackrel{*}{\Rightarrow} p' \stackrel{*}{\Rightarrow} q. \blacklozenge$

Preuve du théorème 2.1.1:

- Puisque le système S conserve les figures décrites, nous avons:

$$\text{Im}_S(w) \subset \{ v \in \Pi^* / \text{dpic}(v) = \text{dpic}(w) \}.$$

- Démontrons la réciproque:

Soient deux mots de Π^* , v et w , décrivant la même figure.

D'après le corollaire 2.1.4, il existe deux mots réduits v' et w' , dérivés de v et de w par S . Comme le système est symétrique, v peut également être dérivé de v' par S .

Puisque le système S conserve les figures décrites, les mots réduits v' et w' décrivent la même figure: nous savons dériver w' en v' par S [corollaire 2.1.10].

Par conséquent, nous pouvons dériver w de la façon suivante:

$$w \stackrel{*}{\Rightarrow} w' \stackrel{*}{\Rightarrow} v' \stackrel{*}{\Rightarrow} v. \blacklozenge$$

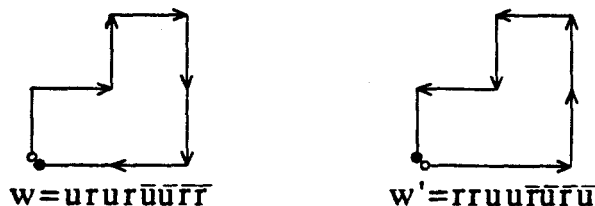
Remarque: Le système S est complet dans le sens où il permet de dériver à partir de tout mot $w \in \Pi^*$, tous les mots décrivant la même figure. Notons que pour satisfaire cette propriété,

Les quatre familles de règles de S sont nécessaires.

Aucune d'entre elles ne peut être simulée par les trois autres: elles sont indépendantes.

Nous justifions cette affirmation à l'aide d'exemples simples pour lesquels les mots w et w' décrivent la même figure.

Exemple 1:



Il est impossible de dériver w en w' sans utiliser la règle R1: en effet, les règles de $R2 \cup R2' \cup R3$ ne peuvent pas modifier la première lettre du mot à dériver.

Exemple 2:



Il est impossible de dériver w en w' sans utiliser la règle R2: Raisonnons par l'absurde et supposons que cela soit possible. La dernière étape de la dérivation utilise nécessairement une règle de R3 (car la longueur doit diminuer). Ceci est impossible car aucun facteur de w' n'est une boucle non vide. Contradiction.

Exemple 3:



Il est impossible de dériver w en w' sans utiliser la règle R2': en effet, seule la règle R2' permet d'augmenter la longueur du mot dérivé.

Exemple 4:



Il est impossible de dériver w en w' sans utiliser la règle R3: en effet, seule la règle R3 modifie la parité de la multiplicité des segments engendrés.

2.1.3 Obtenir un mot minimal

Le système S étant complet, nous savons qu'à partir de tout mot w de Π^* , il est possible de dériver tous les mots minimaux décrivant la même figure. Malheureusement, le système S n'étant pas noethérien (il comporte des règles qui augmentent indéfiniment la longueur du mot dérivé), nous avons besoin d'un algorithme pour être sûr d'obtenir un mot minimal après un certain nombre de dérivations.

Les différentes étapes de la preuve du théorème précédent nous fournissent cet algorithme de recherche d'un mot minimal. Mais d'abord, caractérisons les mots minimaux de Π^* .

- Un cycle C est *majoritaire à 2* (pour $w \in \Pi^*$) si les segments de C de multiplicité égale à 2 (relativement à w) sont strictement majoritaires.

Lemme 2.1.11:

Pour tout mot $m \in \Pi^$,*

m est minimal

\Leftrightarrow

m est réduit

et il n'existe pas de cycle majoritaire à 2, issu de $\text{pic}(m)$.

Preuve:

Soit $m \in \Pi^*$.

- Si m est minimal alors m est nécessairement réduit [corollaire 2.1.3] et il n'existe pas de cycle majoritaire à 2 issu de $\text{pic}(m)$ (sinon il serait possible de dériver m de façon à inverser la multiplicité des segments de ce cycle [lemme 2.1.9] et nous obtiendrions un mot plus court que m).
- Le mot m est réduit et il n'existe pas de cycle majoritaire à 2, issu de $\text{pic}(m)$. Supposons que m ne soit pas minimal. Il existe un mot m' , minimal (et réduit [corollaire 2.1.3]) décrivant la même figure que m . D'après le lemme 2.1.8, les segments de $\text{smd}(m, m')$ peuvent être assemblés en cycles, disjoints deux à deux, et au moins l'un d'entre eux est majoritaire à 2 pour m (sinon m serait minimal). Contradiction. ♦

Corollaire 2.1.12:

Pour tout mot $w \in \Pi^$ de longueur n , le mot w se dérive en un mot minimal par S , avec $O(n^2)$ étapes de dérivation.*

Preuve:

Soit $w \in \Pi^*$ et posons $n = |w|$. Il suffit d'appliquer l'algorithme suivant:

Algorithme:

- 1^{ère} étape: Nous dérivons le mot w en un mot réduit p [corollaire 2.1.4].
- 2^{ème} étape: Nous cherchons un cycle majoritaire à 2 pour p , issu de $\text{pic}(w)$. Si un tel cycle existe, nous dérivons p en p' de façon à inverser la multiplicité des segments de ce cycle [lemme 2.1.9]. Nous obtenons un mot réduit p' vérifiant $|p'| < |p|$. Nous recommençons cette opération en remplaçant p par p' . Puisque $|p|$ décroît strictement à chaque itération, ce traitement s'arrête et le mot obtenu est minimal [lemme 2.1.11].

Notons que le nombre de segments de la figure $\text{pic}(w)$ est borné par n .

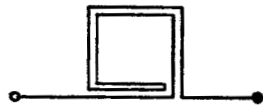
Estimation du nombre d'étapes de dérivation:

- 1^{ère} étape: Pour diminuer de deux la multiplicité d'un segment, nous appliquons au maximum trois règles de S [lemme 2.1.2]. Un mot réduit p est obtenu après au plus $f_1(n) = 3n$ étapes de dérivation.
- 2^{ème} étape: Pour inverser la multiplicité des segments d'un cycle, nous appliquons deux règles par segment de ce cycle, puis une règle fixe: ce traitement nécessite au plus $f_2(n) = 2n + 1$ étapes de dérivation. Comme la longueur du mot dérivé diminue strictement à chaque itération, nous effectuons au plus n fois cette opération.

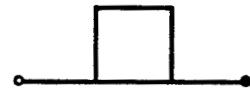
Par conséquent nous obtenons un mot minimal après au plus $f(n) = f_1(n) + nf_2(n) = 2n^2 + 4n$ étapes de dérivation.

Exemple:

$$w = rru\bar{r}ur\bar{r}ur\bar{r}$$

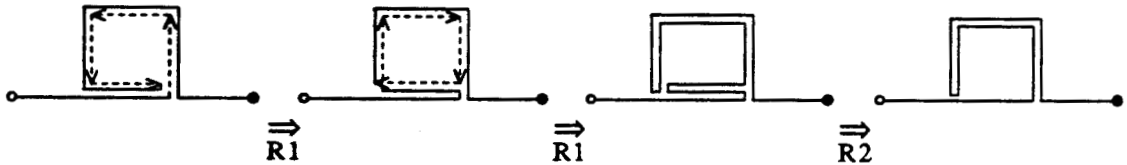


Le tracé de w



La figure décrite

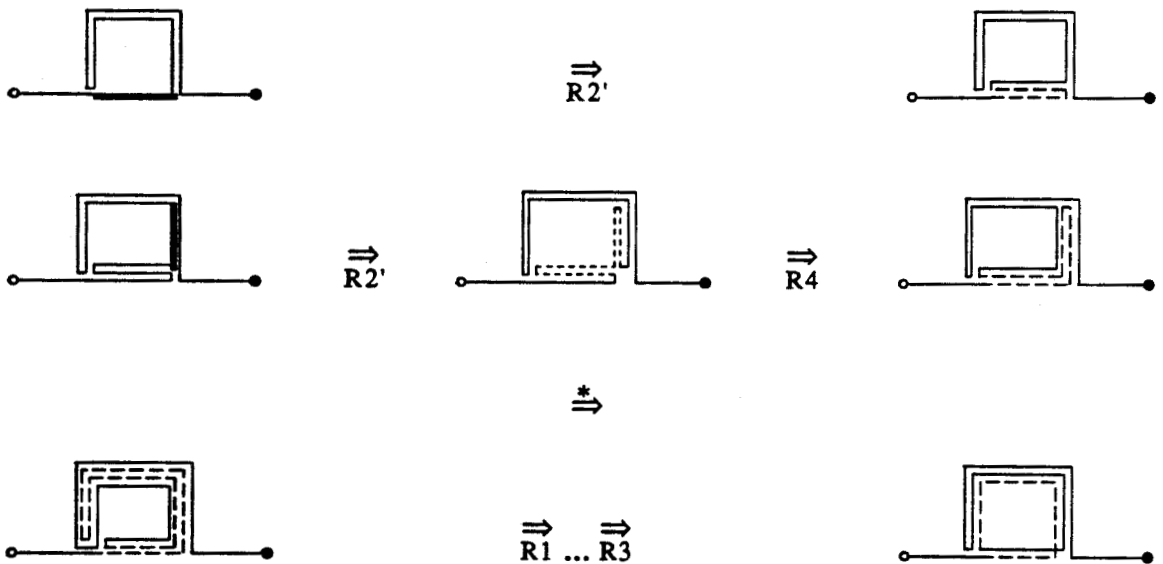
• Obtenir un mot réduit:



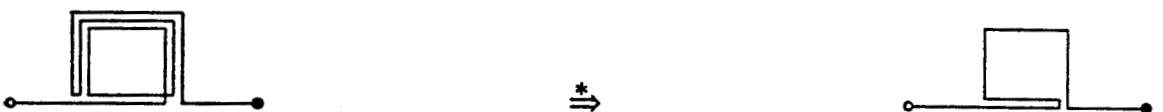
• Rechercher un cycle majoritaire à 2:



• Insérer une boucle décrivant ce cycle:



• Dériver en un mot réduit:



Complexité de l'algorithme:

La difficulté est d'estimer la recherche d'un cycle majoritaire à 2:

Pour cela, considérons le graphe formé des segments de $\text{pic}(w)$ et pondérons ses arêtes de la façon suivante: à tout segment de multiplicité égale à 1 (relativement à p), le poids sera (1) et à tout segment de multiplicité égale à 2 (relativement à p), le poids sera (-1).

Trouver un cycle majoritaire à 2 (pour p) est équivalent à trouver un cycle de somme négative dans ce graphe pondéré. D'après R.M. Karp [GM], nous pouvons trouver un cycle de moyenne minimale (la moyenne d'un cycle est la moyenne des poids de ce cycle et un cycle est de moyenne minimale si sa moyenne est la plus basse possible) en temps $O(\text{nombre de sommets} * \text{nombre d'arêtes}) = O(n^2)$. Tant que cette moyenne est négative, il s'agit d'un cycle à traiter sinon le mot obtenu est minimal.

Cette recherche s'effectuant au plus n fois, l'algorithme énoncé ci-dessus est en $O(n^3)$. ♦

Le corollaire suivant est une variante de la proposition 2.1.5:

Corollaire 2.1.13:

Pour tout mot $w \in \Pi^$, il existe un mot minimal m' , dérivé de w par S ,
vérifiant: $\forall t \in \vec{\text{pic}}(m'), \text{mult}_m(t) \leq 1$.*

Preuve:

- . Pour tout mot $w \in \Pi^*$, il existe un mot minimal m , dérivé de w par S [théorème 2.1.1]. En particulier, m est un mot réduit [corollaire 2.1.3].
- . D'après la proposition 2.1.5, nous pouvons dériver m en un mot m' vérifiant: $\forall t \in \vec{\text{pic}}(m'), \text{mult}_m(t) \leq 1$. Par définition, m' est réduit.
- . Pour dériver m en m' , nous pouvons utiliser des règles de $R1$ [lemme 2.1.6]. Comme $R1$ ne modifie pas la longueur du mot, m' est minimal. ♦

2.1.4 Parlons un peu des autres

Des travaux récents et simultanés ont abouti à des résultats similaires aux nôtres. Citons en particulier les travaux de J. F. Brandenburg & J. Dassow et de R. Gutbrod.

- J. F. Brandenburg & J. Dassow [BD] ont introduit le système de réécriture R suivant:

$$R=R1 \cup R2 \cup R3 \cup R4$$

avec

$$\begin{aligned} R1 &= \{ ac'aca \rightarrow ac'\bar{c} \ / a \in \Pi, ca, c'a \in \mathbb{B} \} \\ R2 &= \{ acab\bar{a} \rightarrow \bar{c}b\bar{a} \ / a \in \Pi, ca, b \in \mathbb{B} \} \\ R3 &= \{ ab\bar{a}b'a \rightarrow \bar{b}'ab \ / a \in \Pi, b, b' \in \mathbb{B} \} \\ R4 &= \{ ab\bar{a}d\bar{a} \rightarrow ab\bar{d} \ / a \in \Pi, b, d, d\bar{a} \in \mathbb{B} \} \end{aligned}$$

Il s'agit d'un système contenant une infinité de règles dont le but est de pouvoir transformer tout mot en un mot réduit (mais pas forcément en un mot minimal). Puis, ils annoncent qu'un mot minimal peut être calculé en temps polynômial à l'aide de l'algorithme du *Plus court tour du postier chinois* de Edmonds et Johnson [EJ]. Cet algorithme relève de la théorie des graphes et ne correspond pas à notre approche: dériver un mot minimal à partir d'un mot quelconque à l'aide d'un système de réécriture.

- Le système défini par R. Gutbrod [Gu1, Gu2] est présenté comme un système de transformations T de Π^* . Plusieurs systèmes équivalents sont donnés, ne citons que l'un d'entre eux: soit $w, w_1, w_2, w_3 \in \Pi^*$ tels que $w = w_1 w_2 w_3$.

$$T=INC \cup DEC \cup INV \cup DUP$$

avec

$$\begin{aligned} INC &= \{ w_1 w_2 w_3 \rightarrow w_1 w_2 \bar{w}_2 w_2 w_3 \} \\ DEC &= \{ w_1 w_2 \bar{w}_2 w_2 w_3 \rightarrow w_1 w_2 w_3 \} \\ INV &= \{ w_1 w_2 w_3 \rightarrow w_1 \bar{w}_2 w_3 \ / w_2 \in \mathbb{B} \} \\ DUP &= \{ w_1 w_2 w_3 \rightarrow w_1 w_2 w_2 w_3 \ / w_2 \in \mathbb{B} \} \end{aligned}$$

Ce système ressemble beaucoup à notre système de réécriture S et R. Gutbrod établit un certain nombre de résultats identiques aux nôtres: il définit un algorithme pour rechercher une description minimale. Cependant n'ayant pas introduit la notion de mot réduit, l'algorithme s'attaque directement aux cycles à "améliorer" sur des mots quelconques: ce procédé est plus coûteux.

Dans la suite, nous nous intéressons en particulier à deux transformations introduites par R. Gutbrod:

$$T6 = \{ w_1 w_2 \rightarrow w_1 w_2 \bar{w}_2 \}$$

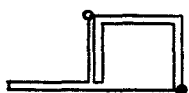
$$T7 = \{ w_1 w_2 \rightarrow \bar{w}_1 w_1 w_2 \}$$

Ces transformations ne conservent plus les figures décrites mais nous permettent d'obtenir des mots ayant les mêmes figures de base que le mot initial (indépendamment des points de départ et d'arrivée).

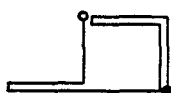
2.1.5 Quelques mots concernant les figures de base

Notre étude est essentiellement axée sur les figures décrites avec les points de départ et d'arrivée fixés. Il peut être cependant intéressant de signaler que nous pouvons obtenir une description encore plus courte que la précédente si nous nous autorisons le choix des points de départ et d'arrivée de la figure.

Exemple:



un mot quelconque



un mot minimal
(figure décrite)



un mot minimal
(figure de base)

Les mots de figure sont maintenant encadrés par le symbole "#" pour restreindre l'usage des règles que nous ajoutons, S6 et S7, aux seuls facteurs droits et gauches du mot à dériver (ces règles sont inspirées des transformations T6 et T7 de R. Gutbrod):

$$S6 = \{ x\# \rightarrow x\bar{x}\# \quad / x \in \Pi^+ \}$$

$$S7 = \{ \#x \rightarrow \#\bar{x}x \quad / x \in \Pi^+ \}$$

Remarque:

- Les règles S6 et S7 conservent les figures de base.

En considérant le système de réécriture $S_{bas} = S \cup S6 \cup S7$, nous pouvons étendre certains résultats établis pour les figures décrites aux figures de base.

Théorème 2.1.14:

Pour tout mot $w \in \# \Pi^ \#$, l'ensemble des mots dérivés de w par S_{bas} est l'ensemble des mots de $\# \Pi^* \#$ ayant la même figure de base que w .*

$$Im_{S_{bas}}(w) = \{ v \in \# \Pi^* \# \mid bpic(v) = bpic(w) \}$$

Preuve:

- Puisque le système S_{bas} conserve les figures de base (le système S conserve les figures décrites, il conserve à fortiori les figures de base), nous avons:

$$Im_{S_{bas}}(w) \subset \{ v \in \# \Pi^* \# \mid bpic(v) = bpic(w) \}.$$

- Considérons l'ensemble de toutes les figures décrites correspondant à la même figure de base $bpic(w)$. Chacune de ces figures peut être décrite par un mot dérivé de w à l'aide des règles $S6$ et $S7$: le théorème 2.1.1 nous permet de conclure. ♦

Corollaire 2.1.15:

Pour tout mot $w \in \# \Pi^ \#$ de longueur n , le mot w se dérive en un mot minimal au sens des figures de base par S , avec $O(n^4)$ étapes de dérivation.*

Preuve:

Il suffit d'appliquer l'algorithme suivant:

Posons $n = |w|$. Pour chaque mot obtenu en faisant varier les points de départ et d'arrivée du tracé à l'aide des règles $S6$ et $S7$ ($(n-1)^2$ mots possibles), nous calculons un mot minimal comme au corollaire 2.1.13 et nous retenons l'un des plus courts. ♦

Remarque 2.1.16:

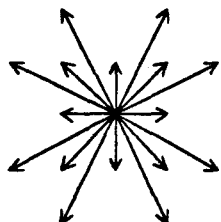
Dans le système S_{bas} , la règle $R2'$ n'est plus nécessaire: elle peut être simulée par les autres règles de S_{bas} .

Preuve:

Soit $w = \#w_1aw_2\#$ avec $w_1, w_2 \in \Pi^*$ et $a \in \Pi$. Nous pouvons remplacer la dérivation $w = \#w_1aw_2\# \xRightarrow{R_2'} \#w_1a\bar{a}aw_2\#$ par la suite des dérivations suivantes:

$$\begin{aligned}
 \#w\# = \#w_1aw_2\# &\xRightarrow{S_7} \# \bar{a}\bar{w}_1w_1aw_2\# &&\xRightarrow{S_6} \# \bar{a}\bar{w}_1w_1aw_2\bar{w}_2\bar{a}\# \\
 \# \bar{a}\bar{w}_1w_1aw_2\bar{w}_2\bar{a}\# &\xRightarrow{R_1} \#w_2\bar{w}_2\bar{a}\bar{w}_1w_1a\bar{a}\# &&\xRightarrow{R_4} \#w_2\bar{w}_2\bar{a}\bar{a}\bar{a}\bar{w}_1w_1\# \\
 \#w_2\bar{w}_2\bar{a}\bar{a}\bar{a}\bar{w}_1w_1\# &\xRightarrow{S_7} \# \bar{w}_2w_2\bar{w}_2\bar{a}\bar{a}\bar{a}\bar{w}_1w_1\# &&\xRightarrow{S_6} \# \bar{w}_2w_2\bar{w}_2\bar{a}\bar{a}\bar{a}\bar{w}_1w_1\bar{w}_1\# \\
 \# \bar{w}_2w_2\bar{w}_2\bar{a}\bar{a}\bar{a}\bar{w}_1w_1\bar{w}_1\# &\xRightarrow{R_5^*} \# \bar{w}_2\bar{a}\bar{a}\bar{w}_1\# &&\xRightarrow{S_7} \#w_1a\bar{a}aw_2\bar{w}_2\bar{a}\bar{a}\bar{w}_1\# \\
 \#w_1a\bar{a}aw_2\bar{w}_2\bar{a}\bar{a}\bar{w}_1\# &\xRightarrow{S_6} \#w_1a\bar{a}aw_2\bar{w}_2\bar{a}\bar{a}\bar{w}_1w_1a\bar{a}aw_2\# &&\xRightarrow{R_5} \#w_1a\bar{a}aw_2\#
 \end{aligned}$$

◆



Nous venons d'étudier un système de réécriture S dont le but est de conserver les figures décrites. Ce système est puissant dans le sens où il permet d'obtenir tous les mots décrivant la même figure et en particulier, tous les mots minimaux. Nous pouvons facilement nous convaincre que tous ces résultats sont encore valables si nous ajoutons d'autres déplacements élémentaires du plan et la généralisation à la troisième dimension ne semble pas poser de problèmes.

En contrepartie, le fait que le système S permette de nombreuses manipulations sur les mots peut modifier certaines propriétés. En effet, nous nous intéresserons davantage à des langages de mots de figure qu'à des mots isolés et malheureusement, le système S ne conserve pas la rationalité du langage. En effet, soit $Im_S(L)$, l'ensemble des mots dérivés par S du langage rationnel $L = r^*$. Nous avons: $Im_S(L) \cap r^*\bar{r}^*r^* = \{ r^i\bar{r}^j r^k / j \leq i \text{ et } j \leq k \}$. Cette intersection n'est pas un langage algébrique, par conséquent $Im_S(L)$ n'est ni algébrique, ni rationnel.

Etudions une situation intermédiaire: un système de réécriture restreint permettant de diminuer la longueur des mots de figure (mais pas forcément de les minimaliser) tout en conservant la rationalité du langage.

2.2 Le système de réécriture Sr-red

Ce système permet de dériver un mot de Π^* de façon à éliminer les aller-retours redondants de celui-ci: après chaque dérivation, le mot obtenu est strictement plus court et décrit la même figure que le mot initial. Nous montrons que ce système est confluent: deux mots dérivés d'un même mot ont un descendant commun. Dans ce cas, le mot irréductible, dérivé d'un mot w de Π^* quelconque, est *unique* (indépendant de l'ordre des dérivations).

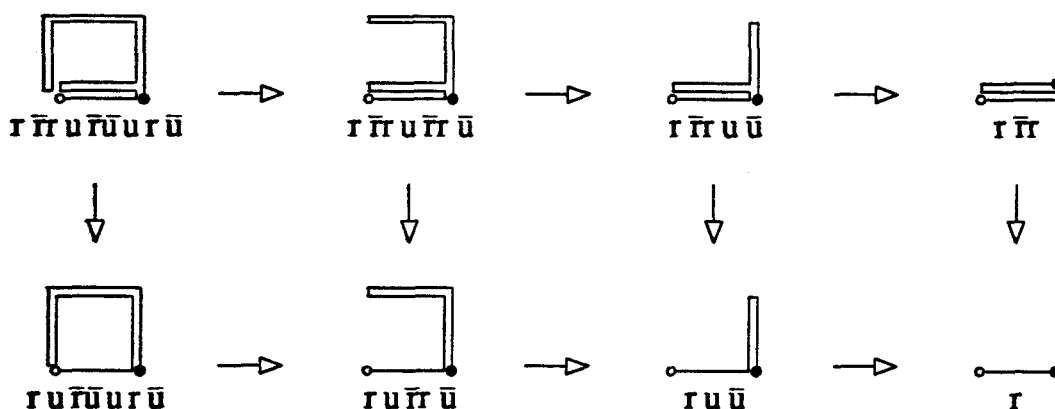
2.2.1 Historique

• Le système de réécriture $S_{red} = \{ a\bar{a} \rightarrow \epsilon / a \in \Pi \}$ a été introduit par H.A. Maurer, G. Rozenberg & E. Welzl [MRW]: il efface deux lettres consécutives d'un mot $w \in \Pi^*$ qui représentent un aller-retour. (Un *aller-retour* est un mot de la forme $a\bar{a}$ avec $a \in \Pi$).

Exemple:

$$w = r\bar{r}u\bar{r}\bar{u}u\bar{r}\bar{u}$$

Les mots dérivés de w sont:



Les figures associées sont:



Remarque: Ce système ne conserve pas les figures: le point d'arrivée de la figure est inchangé au cours des dérivations mais certains segments peuvent ne plus être engendrés. Cette réduction peut être utilisée dans le cas particulier où seul le point d'arrivée de la figure nous intéresse, indépendamment du parcours.

Rappelons les deux résultats bien connus suivants [Bm, MRW]:

Le système S_{red} conserve la rationalité:

Pour tout langage rationnel $L \subset \Pi^$, $Im_{S_{red}}(L)$ est un langage rationnel.*

Le système S_{red} est confluent:

Pour tout mot $w \in \Pi^$, $Im_{S_{red}}(w)$ contient un unique mot sans aller-retour.*

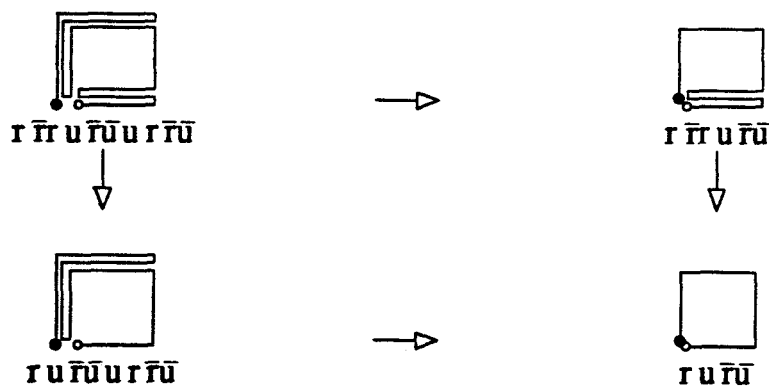
Ces résultats nous intéresseraient davantage si le système S_{red} conservait les figures. Dans cette optique, F. Hinz a défini un système de réécriture qui ne modifie que les facteurs qui induisent des aller-retours redondants.

- Le système de réécriture $S_{r-red} = \{ x\bar{x}x \rightarrow x \mid x \in \Pi^+ \}$ a été défini par F. Hinz [Hi2]: il remplace un aller-retour redondant en un aller simple. (Un *aller-retour redondant* est un mot de la forme $x\bar{x}x$ avec $x \in \Pi^+$, l'*aller simple* correspondant est le mot x).

Exemple:

$$w = r\bar{r}u\bar{r}\bar{u}u\bar{r}\bar{u}$$

Les mots dérivés de w sont:



La figure associée est:



Remarque: Contrairement au système précédent, seules les lettres engendrant des passages superflus sur des segments peuvent être effacées: le système S_{r-red} conserve les figures décrites.

F. Hinz [Hi2] a démontré à l'aide des automates boustrophédons que le système $Sr\text{-red}$ conservait la rationalité:

Pour tout langage rationnel $L \subset \Pi^$, $Im_{Sr\text{-red}}(L)$ est un langage rationnel.*

Nous complétons son étude en montrant que:

Pour tout mot $w \in \Pi^$,
 $Im_{Sr\text{-red}}(w)$ contient un unique mot sans aller-retour redondant.*

2.2.2 De fausses ressemblances

A première vue, ce problème d'unicité ressemble à des résultats déjà connus. Il est vrai que certains énoncés sont proches mais aucun d'entre eux ne permet de conclure positivement (au moins parmi ceux que nous avons étudiés). Le résultat que nous établissons n'est pas spécifique aux langages de mots de figure et s'énonce plus généralement ainsi:

Soit A , un alphabet et $\bar{A} = \{\bar{a} / a \in A\}$, une copie de A telle que $A \cap \bar{A} = \emptyset$.

Nous désignons par $w \rightarrow \bar{w}$, l'anti-isomorphisme défini par:

$$\begin{aligned} \bar{x} &= \bar{a} & \text{si } x &= a \in A, \\ \bar{x} &= a & \text{si } x &= \bar{a} \in \bar{A}, \\ \bar{x} &= \epsilon & \text{si } x &= \epsilon, \end{aligned}$$

$$\text{et } \bar{w}a = \bar{a}\bar{w} \text{ si } a \in (A \cup \bar{A}) \text{ et } w \in (A \cup \bar{A})^*;$$

par α , la congruence engendrée par la relation $\{(x\bar{x}x, x) / x \in (A \cup \bar{A})^+\}$;

et par $[w]_\alpha$, l'ensemble des mots congrus à w modulo α .

Théorème 2.2.1:

Pour tout mot $w \in (A \cup \bar{A})^$, il existe un et un seul mot $\alpha(w) \in [w]_\alpha$, qui ne contienne aucun facteur de type $x\bar{x}$ avec $x \in (A \cup \bar{A})^+$.*

Remarque: Quelle que soit la façon dont nous appliquons la relation $\{(x\bar{x}x, x) / x \in (A \cup \bar{A})^+\} \cup \{(x, x\bar{x}x) / x \in (A \cup \bar{A})^+\}$ à un mot $w \in (A \cup \bar{A})^*$ pour obtenir un mot sans facteur de type $x\bar{x}$ avec $x \in (A \cup \bar{A})^+$, ce mot sera toujours le même. Nous dirons que ce mot est *irréductible* pour α et nous le noterons $\alpha(w)$.

Quelques comparaisons avec des résultats connus:

- Pécuchet a montré le résultat suivant ([Pé], lemme 4.1):

Soit σ la congruence engendrée par la relation $\{ (x\bar{x}x, x) / x \in A^+ \cup \bar{A}^+ \}$.

Pour tout mot $w \in (A \cup \bar{A})^*$, il existe un et un seul mot $\sigma(w) \in [w]_\sigma$, qui ne contienne aucun facteur de type $x\bar{x}x$ avec $x \in A^+ \cup \bar{A}^+$.

Dans cet énoncé, le facteur x est formé soit de lettres de A , soit de lettres de \bar{A} . Pouvons nous en déduire le théorème?

• Soient a_1 et a_2 , deux éléments de A , et $w = a_1 \bar{a}_2 a_2 \bar{a}_1 a_1 \bar{a}_2$, un mot de A^* . Le mot w est irréductible pour σ mais ne l'est pas pour α (en choisissant $x = a_1 \bar{a}_2$). Le lemme de Pécuchet ne peut pas s'appliquer directement à notre problème: le facteur x doit pouvoir contenir des lettres de A et des lettres de \bar{A} .

• Considérons maintenant l'alphabet $B = A \cup \bar{A}$ et appliquons le lemme de Pécuchet aux alphabets B et \bar{B} , sa copie disjointe. Pour faire la jonction avec le résultat que nous cherchons à établir, nous définissons un morphisme ϕ de $B \cup \bar{B}$ vers B de la façon suivante: pour tout $a \in B$, $\phi(a) = a$ et $\phi(\bar{a}) = \bar{a}$. Ainsi, pour tout mot $w \in B^*$, $\phi^{-1}(w)$ contient $2^{|w|}$ mots de $(B \cup \bar{B})^*$: en effet, pour chaque lettre a de w , $\phi^{-1}(a)$ peut prendre deux valeurs différentes. D'après le lemme de Pécuchet, pour chaque mot $v \in \phi^{-1}(w)$, il existe un unique mot $\sigma(v) \in [v]_\sigma$ ne contenant pas de facteur du type $x\bar{x}x$ avec $x \in B^+ \cup \bar{B}^+$. Malheureusement, les mots de $\phi^{-1}(w)$ ne fournissent pas tous le même mot irréductible et par conséquent, notre problème n'est pas résolu.

Illustrons ces propos par l'exemple suivant: Soit $B = \{r, u\} \cup \{l, d\}$, un alphabet (par souci de clarté, nous notons $\bar{A} = \{l, d\}$ la copie disjointe de $A = \{r, u\}$) et $w = udulrdudul$, un mot de B^* . L'ensemble $\phi^{-1}(w)$ contient en particulier les mots $v_1 = u\bar{u}lrdudul$ et $v_2 = udul\bar{l}\bar{u}\bar{d}dul$ qui nous donnent de façon unique les mots $\sigma(v_1) = ulrdudul$ et $\sigma(v_2) = udul$ ne contenant pas de facteur de type $x\bar{x}x$ avec $x \in B^+ \cup \bar{B}^+$. Comment en déduire que le mot irréductible pour α est $\alpha(w) = ul$?

- Le monoïde inversif libre sur un alphabet A utilise une congruence τ dont la définition est proche de la congruence α . Pouvons nous utiliser des résultats de cette théorie?

Le monoïde inversif libre sur A est le monoïde quotient $FI_A = (A \cup \bar{A})^* / \tau$ avec τ , la congruence engendrée sur $(A \cup \bar{A})^*$ par la relation:

$$\{ (x\bar{x}x, x) / x \in (A \cup \bar{A})^+ \} \cup \{ (x\bar{x}y\bar{y}, y\bar{y}x\bar{x}) / x, y \in (A \cup \bar{A})^+ \}.$$

Malheureusement, nous nous apercevons que les modifications qu'engendrent la relation $\{ (x\bar{x}y\bar{y}, y\bar{y}x\bar{x}) / x, y \in (A \cup \bar{A})^+ \}$, permettent de transformer un mot irréductible pour α en un mot qui ne l'est plus. Par conséquent, pour tout mot $w \in (A \cup \bar{A})^*$, $[w]_\tau$ peut contenir plusieurs mots irréductibles pour α .

Prenons par exemple l'alphabet $A = \{r, u\}$ et le mot $w = uu\bar{u}\bar{u}r\bar{r}\bar{u}$. Ce mot est irréductible pour α mais ne l'est pas pour τ : en appliquant la relation $(\bar{u}r\bar{r}, r\bar{r}\bar{u})$ à w , nous obtenons le mot $uu\bar{r}\bar{r}\bar{u}\bar{u}$ qui n'est plus irréductible pour α .

Comme nous n'avons pas pu déduire le théorème de résultats existants, donnons-en une preuve directe.

2.2.3 La preuve

Pour prouver ce théorème, il suffit de montrer que le système $S_{r\text{-red}} = \{ x\bar{x}x \rightarrow x / x \in (A \cup \bar{A})^+ \}$ est confluent. En effet, un cas tout à fait similaire a été étudié dans [Bj] et voici le raisonnement:

Si $S_{r\text{-red}}$ est un système confluent, nous savons que ([Bj]) :

$$\forall x, y \in (A \cup \bar{A})^* (x \in [y]_\alpha \Leftrightarrow \exists w \in (A \cup \bar{A})^* / w \in \text{Im}_{S_{r\text{-red}}}(x) \text{ et } w \in \text{Im}_{S_{r\text{-red}}}(y)).$$

Supposons que pour $w \in (A \cup \bar{A})^*$, il existe deux mots irréductibles w_1 et w_2 appartenant à $[w]_\alpha$. En particulier, nous avons $w_1 \in [w_2]_\alpha$ et nous pouvons en déduire qu'il existe $w' \in (A \cup \bar{A})^*$ tel que $w' \in \text{Im}_{S_{r\text{-red}}}(w_1)$ et $w' \in \text{Im}_{S_{r\text{-red}}}(w_2)$. Comme w_1 et w_2 sont irréductibles, nous en concluons que $w' = w_1$ et $w' = w_2$: par conséquent, $w_1 = w_2$.

La preuve de la confluence n'est pas compliquée mais elle nécessite de distinguer de nombreux cas et d'effectuer beaucoup de calculs. Les définitions et les propriétés concernant les systèmes de réécriture sont issues de [Ja].

Un système de réécriture est *confluent* s'il est *noethérien* (pas de dérivation infinie) et *localement confluent* (deux mots dérivés d'un même mot par une dérivation de longueur un, ont un descendant commun). Dans ce cas, la forme normale d'un mot quelconque par ce système est unique.

- Le système S_{r-red} est noethérien: la longueur du mot décroît strictement à chaque étape de dérivation.
- Le système S_{r-red} est localement confluent:

Preuve:

Soit $w \in (A \cup \bar{A})^*$ et supposons que w contienne deux aller-retours redondants. Appelons w_1 , le mot dérivé de w obtenu en détruisant l'un des aller-retours redondants et w_2 , le mot dérivé de w en détruisant l'autre. Nous devons montrer qu'il existe un mot $m \in (A \cup \bar{A})^*$ que nous pouvons dériver indépendamment de w_1 ou de w_2 ($m \in \text{Im}_{S_{r-red}}(w_1) \cap \text{Im}_{S_{r-red}}(w_2)$). Pour cela, traitons les différents cas de chevauchement possibles entre les deux aller-retours redondants.

Nous pouvons supposer, sans restreindre le problème, que les deux aller-retours redondants, notés $x\bar{x}$ et $z\bar{z}$ avec $x, z \in (A \cup \bar{A})^+$ vérifient $|x| \geq |z|$. Posons $w = u_1 x \bar{x} u_2 = v_1 z \bar{z} v_2$ avec $u_1, u_2, v_1, v_2 \in (A \cup \bar{A})^*$, $w_1 = u_1 x u_2 \in \text{Im}_{S_{r-red}}(w)$, $w_2 = v_1 z v_2 \in \text{Im}_{S_{r-red}}(w)$ et montrons qu'il existe toujours un mot $m \in \text{Im}_{S_{r-red}}(w_1) \cap \text{Im}_{S_{r-red}}(w_2)$.

Quelques propriétés et quelques notations utilisées dans la suite:

Lemme 2.2.2:

Soient $v, v_1, z \in (A \cup \bar{A})^*$.

- (1) Si $v_1 \in FG(v)$ alors $v \in \text{Im}_{S_{r-red}}(v_1 \bar{v}_1 v)$ et $\bar{v} \in \text{Im}_{S_{r-red}}(v \bar{v}_1 \bar{v}_1)$
- (2) Si $v \in FG[(z\bar{z})^*]$ alors $z\bar{z} \in \text{Im}_{S_{r-red}}(z\bar{z} v \bar{v})$ et $z\bar{z} \in \text{Im}_{S_{r-red}}(v \bar{v} z\bar{z})$
- (3) Si $v = \bar{v}$ alors il existe $v' \in (A \cup \bar{A})^*$ tel que $v \bar{v}' = v$

Notations:

- Nous décomposons les facteurs x et z ainsi:
 $x = x_1 x_2$, $z = z_1 z_2 = z_3 z_4 = z_5 z_6$ avec $x_1, x_2, z_1, z_2, z_3, z_4, z_5, z_6 \in (A \cup \bar{A})^*$.
- Nous indiquons par $w \xrightarrow{(i)} w'$, le fait qu'en appliquant la propriété (i) du lemme précédent à w , nous en déduisons que $w' \in \text{Im}_{S_{r-red}}(w)$.
- Un facteur d'un mot est en caractère gras s'il est remplacé par un facteur équivalent.

Les différents cas:

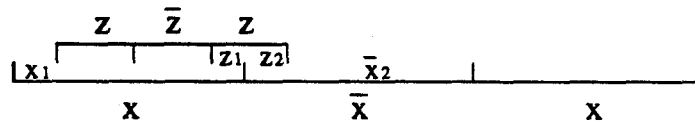
- . Les deux aller-retours redondants sont disjoints: nous avons $w = u_1 x \bar{x} y z \bar{z} z v_2$ avec $y \in (A \cup \bar{A})^*$ et nous en déduisons facilement que $m = u_1 x y z v_2 \in \text{Im}_{S_{r\text{-red}}}(w_1) \cap \text{Im}_{S_{r\text{-red}}}(w_2)$ (le cas $w = v_1 z \bar{z} y x \bar{x} u_2$ est similaire).
- . L'aller-retour redondant $z \bar{z} z$ est un facteur de x (ou \bar{x}): nous avons $w = u_1 x_1 z' \bar{z}' z' x_2 \bar{x}_2 \bar{z}' z' \bar{z}' \bar{x}_1 x_1 z' \bar{z}' z' x_2 u_2$ avec $z' = z$ (ou $z' = \bar{z}$) et nous en déduisons facilement que $m = u_1 x_1 z' x_2 u_2 \in \text{Im}_{S_{r\text{-red}}}(w_1) \cap \text{Im}_{S_{r\text{-red}}}(w_2)$.
- . Pour les autres cas, nous distinguons deux catégories de chevauchement: $z \bar{z} z$ est un facteur de $x \bar{x}$ ou il ne l'est pas.

a) $z \bar{z} z \in F(x \bar{x})$.

Dans ce cas, nous montrons un résultat plus fort: nous pouvons toujours dériver w_2 en w_1 , c'est-à-dire nous pouvons choisir $m = w_1$.

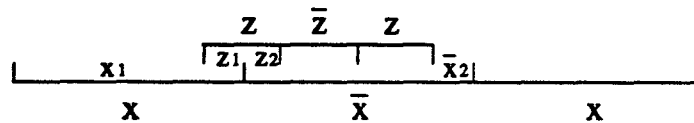
Nous supposons que le facteur $z \bar{z} z$ se trouve dans la partie gauche du facteur $x \bar{x}$. La preuve est exactement la même pour les cas symétriques.

Cas a.1: $z \bar{z} z \in F(x)$ ou $\bar{z} z \in F(\bar{x})$



Dans ce cas, $x = x_1 z \bar{z} z_1 = x_2 \bar{z}_2$.

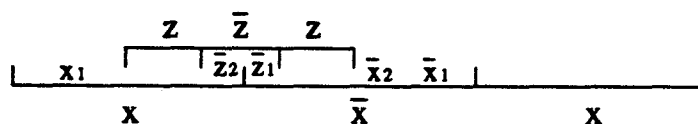
Nous avons $w = u_1 x \bar{x} u_2 = u_1 x_1 z \bar{z} z_1 \bar{x} u_2$ et $w_2 = u_1 x_1 z_1 \bar{x} u_2 = u_1 x_1 z_1 \bar{z}_1 z \bar{x}_1 u_2$, nous en déduisons: $w_2 = u_1 x_1 z_1 \bar{z}_1 z \bar{x}_1 u_2 \xrightarrow{(1)} u_1 x_1 z \bar{x}_1 u_2 \xrightarrow{(1)} u_1 x u_2 = w_1$.



Dans ce cas, $x = x_1 z_1 = x_2 \bar{z} z \bar{z}_2$.

Nous avons $w = u_1 x \bar{x} u_2 = u_1 x z_2 \bar{z} z \bar{x}_2 u_2$ et $w_2 = u_1 x z_2 \bar{x}_2 u_2 = u_1 x_2 \bar{z} z \bar{z}_2 z_2 \bar{x}_2 u_2$, nous en déduisons: $w_2 = u_1 x_2 \bar{z} z \bar{z}_2 z_2 \bar{x}_2 u_2 \xrightarrow{(1)} u_1 x_2 \bar{z} z \bar{x}_2 u_2 \xrightarrow{(1)} u_1 x u_2 = w_1$.

Cas a.2: $z\bar{z} \in F(x\bar{x})$ (excepté le cas a.1)



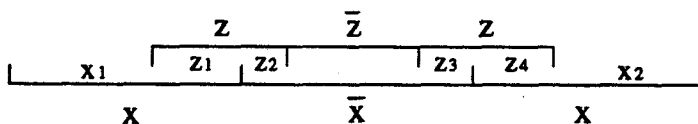
Dans ce cas, $x = x_1 z \bar{z}_2 = x_1 x_2 \bar{z}_1$.

Supposons $|z_1| \leq |z_2|$ (l'autre cas est similaire) et posons $\bar{z}_2 = \bar{c}z_1$, $\bar{c} \in (A \cup \bar{A})^*$. Puisque $x_1 z \bar{z}_2 = x_1 x_2 \bar{z}_1$ et $\bar{z}_2 = \bar{c}z_1$, nous avons $z\bar{c} = x_2 \bar{z}$ et ainsi $|c| = |x_2|$. Sachant que $|c| \leq |z_2| \leq |z|$, nous avons $x_2 \in FG(z)$ et $\bar{c} \in FG(z)$: par conséquent, $x_2 = c$. De plus, $x_2 \in FG(z) \Rightarrow x_2 \in FG(z_1 z_2) \Rightarrow x_2 \in FG(z_1 \bar{x}_1 x_2) \Rightarrow x_2 \in FG((z_1 \bar{z}_1)^*)$.

Nous avons $w = u_1 x \bar{x} u_2 = u_1 x_1 z \bar{z}_2 \bar{z}_1 z \bar{x}_2 \bar{x}_1 x u_2$ et $w_2 = u_1 x_1 z \bar{x}_2 \bar{x}_1 x u_2 = u_1 x_1 z_1 \bar{z}_1 x_2 \bar{x}_2 \bar{x}_1 x u_2$, nous en déduisons: $w_2 = u_1 x_1 z_1 \bar{z}_1 x_2 \bar{x}_2 \bar{x}_1 x u_2 \xrightarrow{(2)} u_1 x_1 z_1 \bar{z}_1 \bar{x}_1 x u_2 \xrightarrow{(1)} u_1 x u_2 = w_1$.

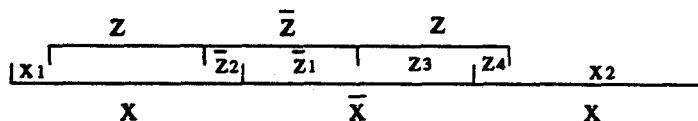
Case a.3: $z\bar{z} \in F(x\bar{x}x)$ (exceptés les cas a.1 et a.2)

Deux chevauchements sont possibles:



Dans ce cas, $x = x_1 z_1 = \bar{z}_3 z \bar{z}_2 = z_4 x_2$.

Nous avons $w = u_1 x \bar{x} u_2 = u_1 x z_2 \bar{z}_3 z_4 x_2 u_2$ et $w_2 = u_1 x z_2 x_2 u_2 = u_1 \bar{z}_3 z \bar{z}_2 z_4 x_2 u_2$, nous en déduisons: $w_2 = u_1 \bar{z}_3 z \bar{z}_2 z_4 x_2 u_2 \xrightarrow{(1)} u_1 \bar{z}_3 z x_2 u_2 = u_1 \bar{z}_3 z_3 z_4 x_2 u_2 = u_1 \bar{z}_3 z_3 x u_2 \xrightarrow{(1)} u_1 x u_2 = w_1$.



Dans ce cas, $x = x_1 z \bar{z}_2 = \bar{z}_3 z_1 = z_4 x_2$.

Nous avons $x = x_1 z_1 z_2 \bar{z}_2 = \bar{z}_3 z_1$: par conséquent $|\bar{z}_3| \geq |x_1|$ et nous posons $\bar{z}_3 = x_1 d$, $d \in (A \cup \bar{A})^*$. Puisque $x = x_1 z \bar{z}_2 = x_1 z_3 z_4 \bar{z}_2 = x_1 d \bar{x}_1 z_4 \bar{z}_2$ et $x = \bar{z}_3 z_1 = x_1 d z_1$, nous avons $d = \bar{d}$ et en appliquant la propriété (3) du lemme 2.2.2, nous savons qu'il existe $d' \in (A \cup \bar{A})^*$ tel que $d = d' \bar{d}'$ (notons que $x_1 d' \in FG(x)$).

Nous avons $w = u_1 x \bar{x} u_2 = u_1 x_1 z \bar{z}_2 \bar{z}_1 z_3 x u_2$ et $w_2 = u_1 x_1 z_3 x u_2 = u_1 x_1 d' \bar{d}' \bar{x}_1 x u_2$, nous en déduisons: $w_2 = u_1 x_1 d' \bar{d}' \bar{x}_1 x u_2 \xrightarrow{(1)} u_1 x u_2 = w_1$.

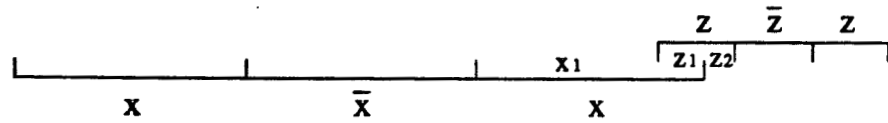
$$b) \quad z\bar{z}z \in F(x\bar{x}x).$$

Nous supposons que le facteur $z\bar{z}z$ chevauche la partie droite du facteur $x\bar{x}x$. La preuve est exactement la même pour les cas symétriques.

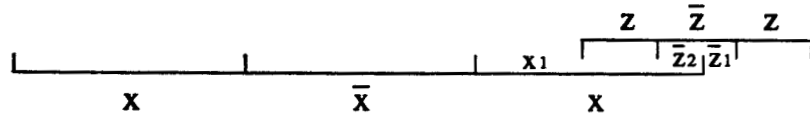
Ainsi, le facteur commun aux deux aller-retours redondants, noté f_{com} , vérifie $f_{com} \in FD(x\bar{x}x) \cap FG(z\bar{z}z)$. Pour chaque chevauchement possible, nous trouvons un mot $m \in \text{Im}_{S_{r-red}}(w_1) \cap \text{Im}_{S_{r-red}}(w_2)$.

Case b.1: $|f_{com}| < |x|$

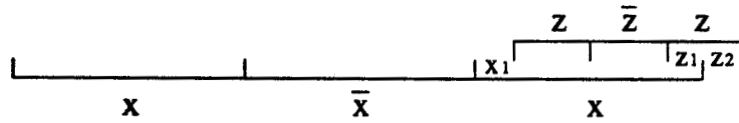
Trois chevauchements sont possibles:



(1) Dans ce cas, $x = x_1 z_1$.



(2) Dans ce cas, $x = x_1 z \bar{z}_2$.



(3) Dans ce cas, $x = x_1 z \bar{z}_1$.

Nous avons $w = u_1 x \bar{x} x u_2 = v_1 z \bar{z} z v_2 = u_1 x \bar{x} x_1 z \bar{z} z v_2$ et nous obtenons $w_1 = u_1 x_1 z \bar{z} z v_2$ and $w_2 = u_1 x \bar{x} x_1 z v_2$. Posons $m = u_1 x_1 z v_2$.

Nous avons $w_1 = u_1 x_1 z \bar{z} z v_2 \stackrel{(1)}{\rightarrow} u_1 x_1 z v_2 = m$.

Donc, $m \in \text{Im}_{S_{r-red}}(w_1)$.

Pour montrer que $m \in \text{Im}_{S_{r-red}}(w_2)$, nous devons distinguer les trois cas:

(1) Nous avons $w_2 = u_1 x \bar{x} x_1 z v_2 = u_1 x \bar{x} x_1 z_1 z_2 v_2 = u_1 x \bar{x} x z_2 v_2$ et nous en déduisons: $w_2 = u_1 x \bar{x} x z_2 v_2 \stackrel{(1)}{\rightarrow} u_1 x z_2 v_2 = u_1 x_1 z_1 z_2 v_2 = u_1 x_1 z v_2 = m$.

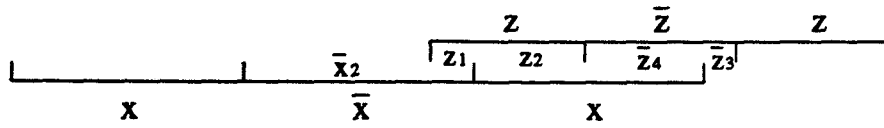
(2) Nous avons $w_2 = u_1 x \bar{x} x_1 z v_2 = u_1 x_1 z \bar{z}_2 z_2 \bar{z} \bar{x}_1 x_1 z v_2$ et nous en déduisons: $w_2 = u_1 x_1 z \bar{z}_2 z_2 \bar{z} \bar{x}_1 x_1 z v_2 \stackrel{(1)}{\rightarrow} u_1 x_1 z \bar{z} \bar{x}_1 x_1 z v_2 \stackrel{(1)}{\rightarrow} u_1 x_1 z v_2 = m$.

(3) Nous avons $w_2 = u_1 x \bar{x} x_1 z v_2 = u_1 x_1 z \bar{z}_1 z_1 \bar{z} \bar{x}_1 x_1 z v_2$ et nous en déduisons: $w_2 = u_1 x_1 z \bar{z}_1 z_1 \bar{z} \bar{x}_1 x_1 z v_2 \stackrel{(1)}{\rightarrow} u_1 x_1 z \bar{z} \bar{x}_1 x_1 z v_2 \stackrel{(1)}{\rightarrow} u_1 x_1 z v_2 = m$.

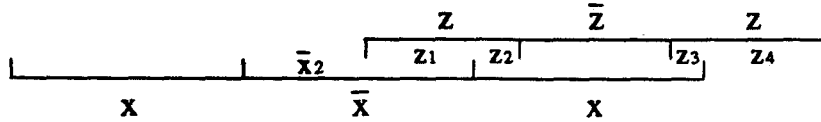
Donc, $m \in \text{Im}_{S_{r-red}}(w_2)$.

Case b.2: $|x| \leq |fcom| \leq 2|x|$

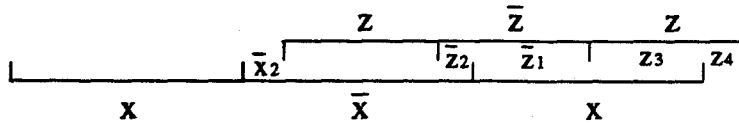
Trois chevauchements sont possibles:



(1) Dans ce cas, $x = \bar{z}_1 x_2 = z_2 \bar{z}_4$.



(2) Dans ce cas, $x = \bar{z}_1 x_2 = z_2 \bar{z} z_3$.



(3) Dans ce cas, $x = z_2 \bar{z} x_2 = \bar{z}_1 z_3$.

Nous avons $w = u_1 x \bar{x} u_2 = v_1 z \bar{z} v_2 = u_1 x \bar{x}_2 z \bar{z} v_2$.

Pour (1) et (2), nous avons $w = u_1 x \bar{x}_2 z \bar{z} v_2 = u_1 x \bar{x}_2 z_1 z_2 \bar{z} v_2 = u_1 x \bar{x}_2 z_2 \bar{z} v_2$ et nous obtenons $w_1 = u_1 z_2 \bar{z} v_2$ et $w_2 = u_1 x \bar{x} z_2 v_2$.

(1) Soit $m = u_1 z_2 \bar{z}_2 z_2 v_2$. Pour prouver que $m \in \text{Im}_{\text{Sr-red}}(w_1) \cap \text{Im}_{\text{Sr-red}}(w_2)$, nous montrons que $\bar{z}_4 \in \text{FG}[(\bar{z}_2 z_2)^*]$ et $\bar{z}_1 \in \text{FG}[(z_2 \bar{z}_2)^*]$.

Si $|z_4| > |z_2|$, nous posons $\bar{z}_4 = \bar{z}_2 \bar{c}$ avec $\bar{c} \in (A \cup \bar{A})^*$. Puisque $z_1 z_2 = z_3 z_4$ et $\bar{z}_4 = \bar{z}_2 \bar{c}$, nous obtenons $z_1 = z_3 \bar{c}$. Comme $\bar{z}_1 x_2 = z_2 \bar{z}_4$, $z_1 = z_3 \bar{c}$ et $\bar{z}_4 = \bar{z}_2 \bar{c}$, nous avons $\bar{c} z_3 x_2 = z_2 \bar{z}_2 \bar{c}$ et nous en déduisons: $\bar{c} \in \text{FG}(z_2 \bar{z}_2 \bar{c}) \Rightarrow \bar{c} \in \text{FG}[(z_2 \bar{z}_2)^*] \Rightarrow \bar{z}_4 \in \text{FG}[(\bar{z}_2 z_2)^*]$. Si $|z_4| \leq |z_2|$, nous avons évidemment $\bar{z}_4 \in \text{FG}(\bar{z}_2)$.

De là, nous concluons que $x = z_2 \bar{z}_4 \in \text{FG}[(z_2 \bar{z}_2)^*]$ et puisque $\bar{z}_1 \in \text{FG}(x)$, nous obtenons $\bar{z}_1 \in \text{FG}[(z_2 \bar{z}_2)^*]$.

Nous avons $w_1 = u_1 z_2 \bar{z} v_2 = u_1 z_2 \bar{z}_2 \bar{z}_1 z_1 z_2 v_2 \xrightarrow{(2)} u_1 z_2 \bar{z}_2 z_2 v_2 = m$ et $w_2 = u_1 x \bar{x} z_2 v_2 = u_1 z_2 \bar{z}_4 z_4 \bar{z}_2 z_2 v_2 \xrightarrow{(2)} u_1 z_2 \bar{z}_2 z_2 v_2 = m$.

Donc, $m \in \text{Im}_{\text{Sr-red}}(w_1) \cap \text{Im}_{\text{Sr-red}}(w_2)$.

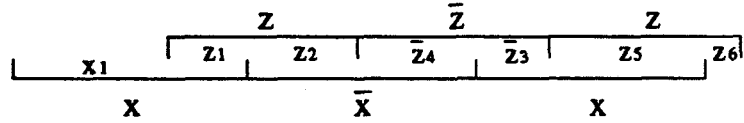
(2) Soit $m = w_1 = u_1 z_2 \bar{z} v_2$. Nous montrons seulement que $m \in \text{Im}_{\text{Sr-red}}(w_2)$.

Nous avons $w_2 = u_1 x \bar{x} z_2 v_2 = u_1 z_2 \bar{z} z_3 \bar{z}_3 z_2 z_2 v_2 \xrightarrow{(1)} u_1 z_2 \bar{z} z_2 z_2 v_2 \xrightarrow{(1)} u_1 z_2 \bar{z} z v_2 = m$.
Donc, $m = w_1 \in \text{Im}_{\text{Sr-red}}(w_2)$.

Pour (3), nous avons $w = u_1 x \bar{x}_2 z \bar{z} z v_2 = u_1 x \bar{x}_2 z \bar{z}_2 \bar{z}_1 z v_2 = u_1 x \bar{x} \bar{z}_1 z v_2$ et nous obtenons $w_1 = u_1 \bar{z}_1 z v_2$ et $w_2 = u_1 x \bar{x}_2 z v_2$.

(3) Soit $m = w_1 = u_1 \bar{z}_1 z v_2$. Nous montrons seulement que $m \in \text{Im}_{\text{Sr-red}}(w_2)$. Puisque $z_2 \bar{x}_2 = \bar{z}_1 z_3$ et $|z_1| \leq |z_2 z|$, posons $z_3 = c x_2$ avec $c \in (A \cup \bar{A})^*$. Nous obtenons $z_2 \bar{z} x_2 = \bar{z}_1 c x_2 \Rightarrow z_2 \bar{z}_4 \bar{z}_3 x_2 = \bar{z}_1 c x_2 \Rightarrow z_2 \bar{z}_4 \bar{x}_2 \bar{c} = \bar{z}_1 c$. Par conséquent, nous avons $c = \bar{c}$ et en appliquant la propriété (3) du lemme 2.2.2, nous savons qu'il existe $c' \in (A \cup \bar{A})^*$ tel que $c = c'$ (notons que $\bar{x}_2 c' \in \text{FG}(\bar{x})$). Nous avons $w_2 = u_1 x \bar{x}_2 z v_2 = u_1 x \bar{x}_2 z_3 z_4 v_2 = u_1 x \bar{x}_2 c' x_2 z_4 v_2 \xrightarrow{(1)} u_1 x z_4 v_2$ et $u_1 x z_4 v_2 = u_1 \bar{z}_1 z_3 z_4 v_2 = u_1 \bar{z}_1 z v_2 = m$. Donc, $m = w_1 \in \text{Im}_{\text{Sr-red}}(w_2)$.

Case b.3: $|f_{\text{com}}| > 2|x|$



Dans ce cas, $x = x_1 z_1 = z_4 \bar{z}_2 = \bar{z}_3 z_5$.

Nous avons $w = u_1 x \bar{x} u_2 = v_1 z \bar{z} z v_2 = u_1 x \bar{x} z_6 v_2 = u_1 x_1 z \bar{z} z v_2$ et nous obtenons $w_1 = u_1 x z_6 v_2$ et $w_2 = u_1 x_1 z v_2$. Posons $m = u_1 z_4 v_2$.

Si $|z_3| > |z_4|$, nous posons $\bar{z}_3 = z_4 c$ avec $c \in (A \cup \bar{A})^*$. Puisque $z_4 \bar{z}_2 = \bar{z}_3 z_5$ et $\bar{z}_3 = z_4 c$, nous obtenons $\bar{z}_2 = c z_5$. Comme $z_1 z_2 = z_3 z_4$, $\bar{z}_2 = c z_5$ et $\bar{z}_3 = z_4 c$, nous avons $z_1 \bar{z}_5 \bar{c} = \bar{c} \bar{z}_4 z_4$. Par conséquent:

$$z_1 \bar{z}_5 \bar{c} = \bar{c} \bar{z}_4 z_4 \Rightarrow c z_5 \bar{z}_1 = \bar{z}_4 z_4 c \Rightarrow c \in \text{FG}(\bar{z}_4 z_4 c) \Rightarrow c \in \text{FG}[(\bar{z}_4 z_4)^*] \Rightarrow \bar{z}_3 \in \text{FG}[(z_4 \bar{z}_4)^*].$$

Si $|z_3| \leq |z_4|$, nous avons évidemment $\bar{z}_3 \in \text{FG}(z_4)$.

De là, nous concluons que $\bar{z} = \bar{z}_4 \bar{z}_3 \in \text{FG}[(\bar{z}_4 z_4)^*]$ et puisque $\bar{z}_2 \in \text{FG}(\bar{z})$, nous obtenons $\bar{z}_2 \in \text{FG}[(\bar{z}_4 z_4)^*]$.

Nous avons $w_1 = u_1 x z_6 v_2 = u_1 \bar{z}_3 z_5 z_6 v_2 = u_1 \bar{z}_3 z_3 z_4 v_2$.

Si $|z_3| \leq |z_4|$ alors $w_1 = u_1 \bar{z}_3 z_3 z_4 v_2 \xrightarrow{(1)} u_1 z_4 v_2 = m$, sinon $w_1 = u_1 \bar{z}_3 z_3 z_4 v_2 = u_1 z_4 c \bar{c} z_4 z_4 v_2 \xrightarrow{(2)} u_1 z_4 \bar{z}_4 z_4 v_2 \xrightarrow{(1)} u_1 z_4 v_2 = m$. Donc, $m \in \text{Im}_{\text{Sr-red}}(w_1)$.

Nous avons $w_2 = u_1 x_1 z v_2 = u_1 x_1 z_1 z_2 v_2 = u_1 z_4 \bar{z}_2 z_2 v_2$.

Si $|z_2| \leq |z_4|$ alors $\bar{z}_2 \in \text{FG}(\bar{z}_4)$ et $w_2 = u_1 z_4 \bar{z}_2 z_2 v_2 \xrightarrow{(1)} u_1 z_4 v_2 = m$, sinon nous posons $\bar{z}_2 = \bar{z}_4 d$ avec $d \in \text{FG}[(z_4 \bar{z}_4)^*]$ et nous obtenons $w_2 = u_1 z_4 \bar{z}_2 z_2 v_2 = u_1 z_4 \bar{z}_4 d \bar{d} z_4 v_2 \xrightarrow{(2)} u_1 z_4 \bar{z}_4 z_4 v_2 \xrightarrow{(1)} u_1 z_4 v_2 = m$. Donc, $m \in \text{Im}_{\text{Sr-red}}(w_2)$.

Par conséquent, le système Sr-red est localement confluent. ♦

Fin de la preuve du théorème. ♦

Chapitre 3:

FIGURE TRACEE AVEC POSSIBILITE DE LEVER LE CRAYON

- 1 - Quelques définitions complémentaires
- 2 - Le système de réécriture S'
 - 2.1 Présentation du système S'
 - 2.2 Le système S' est complet
 - 2.3 Pas de système fini
 - 2.4 Méthode de simplification
- 3 - Le cas particulier des figures connexes
 - 3.1 Quelques définitions
 - 3.2 Une inégalité
 - 3.3 Caractérisation d'un mot minimal de Π'^*
 - 3.4 Relation entre la longueur d'un mot minimal et sa figure de base

Chapitre 3:

Figure tracée avec possibilité de lever le crayon

Pour simuler le lever de crayon dans un mot de figure, nous ajoutons quatre lettres "blanches" à notre alphabet. Elles permettent de déplacer le crayon d'une unité dans chaque direction du plan sans trace.

Cette extension a un double objectif:

- Décrire des figures non connexes:



- Optimiser la description des figures connexes: un mot minimal pour décrire la lettre "m" est plus court si le déplacement en mode levé est autorisé (représenté par une ligne pointillée).



Dans un premier temps, nous introduisons un système de réécriture S' défini sur ce nouvel alphabet Π' . Nous montrons que ce système (qui conserve les figures décrites) est *complet*: il permet, à partir de tout mot $w \in \Pi'^*$, de dériver tous les mots décrivant la même figure. Nous énonçons une méthode pour dériver tout mot de Π'^* en un mot plus court: malheureusement, nous n'obtenons pas de description minimale dans le cas général. C'est pourquoi, dans un second temps, nous nous limitons aux figures connexes pour lesquelles nous savons reconnaître les mots minimaux de Π'^* et estimer leur taille.

Remarque: F. Hinz et E. Welzl ont étudié certaines propriétés des langages de mots de figure avec lignes invisibles [HW]. Ils ont opté pour un alphabet différent: les quatre lettres de Π et deux symboles actionnant le lever et l'abaissement du crayon.

- Une *boucle* est un mot de Π^* dont le tracé est une courbe fermée (le point de départ et le point d'arrivée sont confondus).

$B = \{ b \in \Pi^* / e(b) = (0,0) \}$ est l'ensemble des boucles de Π^* .

Note: $\forall a \in \Pi, e(a_b) = e(a)$.

- La *multiplicité* d'un segment s relative au mot w de Π^* , $mult_w(s)$, est le nombre de passages (visibles ou invisibles) sur ce segment lors du tracé de w .

- Un mot $w \in \Pi^*$ est *réduit* si chaque segment de la figure est parcouru au plus deux fois (de manière visible ou invisible) par w :

(w est réduit) \Leftrightarrow ($\forall s \in \text{pic}(w) \cup \text{seg}_b(w) \quad mult_w(s) \leq 2$).

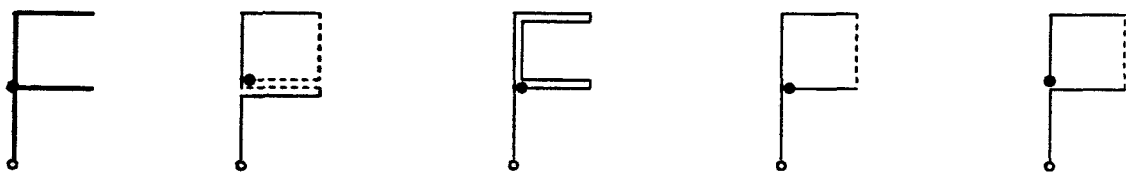
- Un mot $w \in \Pi^*$ est *optimal* si chaque segment de la figure est parcouru au plus une fois de manière visible par w .

- Un mot $w \in \Pi^*$ est *minimal* s'il n'existe aucun mot strictement plus court décrivant la même figure que w :

(w est minimal) \Leftrightarrow ($\{ w' \in \Pi^* / dpic(w) = dpic(w') \text{ et } |w'| < |w| \} = \emptyset$)

Remarque: En général, il existe plusieurs mots réduits, optimaux et minimaux décrivant une figure donnée.

Exemple: Les mots w_1, w_2, w_3 et w_4 décrivent la même figure:



$w_1 = ur\bar{r}_b ur\bar{u}_b \bar{r}_b$
non réduit
optimal
non minimal

$w_2 = uur\bar{r}\bar{u}\bar{r}\bar{r}$
réduit
non optimal
non minimal

$w_3 = uur\bar{u}_b \bar{r}$
réduit
optimal
minimal

$w_4 = uru_b \bar{r}\bar{u}$
réduit
optimal
minimal

3.2 Le système de réécriture S'

3.2.1 Présentation du système S'

Le système S' est caractérisé par 6 ensembles de règles de réécriture:

$$S' = S1 \cup S2 \cup S2' \cup S3 \cup S3' \cup S4$$

avec

$$S1 = \{ b \rightarrow \bar{b} \ / b \in \mathbb{B} \}$$

$$S2 = \{ a\bar{a} \rightarrow a\bar{a}_b \ / a \in \Pi \}$$

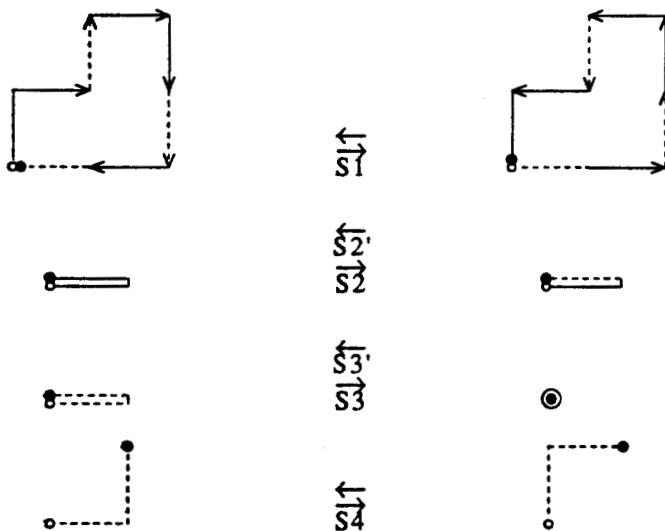
$$S2' = \{ a\bar{a}_b \rightarrow a\bar{a} \ / a \in \Pi \}$$

$$S3 = \{ a_b\bar{a}_b \rightarrow \varepsilon \ / a_b \in \Pi_b \}$$

$$S3' = \{ \varepsilon \rightarrow a_b\bar{a}_b \ / a_b \in \Pi_b \}$$

$$S4 = \{ a_b a'_b \rightarrow a'_b a_b \ / a_b, a'_b \in \Pi_b \}$$

Illustrons ces transformations à l'aide d'exemples:



Remarques (explicitées au § 2.1.1):

- . Toutes les règles de S' conservent les figures décrites.
- . Le système S' contient un nombre infini de règles: notons que seul l'ensemble S1 n'est pas borné.
- . Pour tout mot $w \in \Pi^*$, $\text{Im}_{S'}(w)$ contient une infinité de mots.
- . Le système S' est symétrique.

Notation:

- . Dans la suite, l'expression "la règle S1" signifiera "une règle de l'ensemble de règles S1".

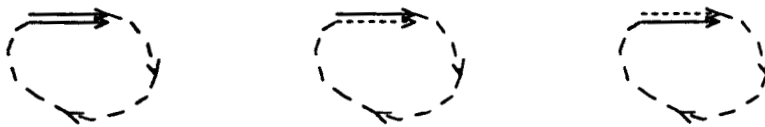
Définition:

- Pour tout $w \in \Pi^*$, w_b est le mot obtenu à partir de w en remplaçant toutes les lettres de Π par les lettres de Π_b correspondantes. Plus formellement, $w_b = h(w)$ avec h , l'homomorphisme défini par:

$$\forall a \in \Pi, h(a) = a_b \text{ et } \forall a_b \in \Pi_b, h(a_b) = a_b.$$

Les règles suivantes peuvent être obtenues en composant des règles de S' . Nous regroupons ces règles par affinité (ressemblance ou symétrie) et nous les appelons *transformations*.

- De $S1, S2$ et $S2'$, nous obtenons:

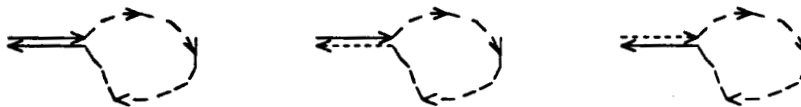


$$T1 = \{ \quad aca \quad \leftrightarrow \quad aca_b \quad \leftrightarrow \quad a_bca \quad / a \in \Pi, ca \in \mathbb{B} \}$$

Preuve:

$$T1: aca \underset{S1}{\leftrightarrow} a\bar{a}\bar{c} \underset{S2, S2'}{\leftrightarrow} a\bar{a}_b\bar{c} \underset{S1}{\leftrightarrow} aca_b \\ \underset{S1}{\leftrightarrow} a_b\bar{a}\bar{c} \underset{S1}{\leftrightarrow} a_bca. \blacklozenge$$

- et de $S1, S2, S2', S3$ et $S3'$:



$$T2 = \{ \quad ab\bar{a} \quad \leftrightarrow \quad ab\bar{a}_b \quad \leftrightarrow \quad a_b b\bar{a} \quad / a \in \Pi, b \in \mathbb{B} \}$$

Preuve:

$$T2: ab\bar{a} \underset{S3, S3'}{\leftrightarrow} ab\bar{a}_b\bar{a}_b \underset{S1}{\leftrightarrow} a\bar{a}_b b\bar{a}_b \underset{S2, S2'}{\leftrightarrow} a\bar{a}_b a_b b\bar{a}_b \underset{S3, S3'}{\leftrightarrow} ab\bar{a}_b \underset{S1}{\leftrightarrow} a_b b\bar{a} \underset{S1}{\leftrightarrow} a_b b\bar{a}. \blacklozenge$$

Les transformations $T1$ et $T2$ permettent de rendre visibles ou invisibles les passages multiples sur un segment parcouru au moins une fois de manière visible. Nous en déduisons le lemme suivant:

Lemme 3.2.1:

Pour tout mot $w \in \Pi^$, il existe un mot optimal, dérivé de w par S' .*

- De S3, S3' et S4, nous obtenons:



$$T3 = \{ \quad b_b \quad \leftrightarrow \quad \varepsilon \quad / \quad b_b \in \mathbb{B} \cap \Pi_b^* \}$$

Preuve:

Si $b_b \in \mathbb{B} \cap \Pi_b^*$ alors b_b est composée uniquement de lettres de Π_b et contient autant de r_b que de \bar{r}_b et autant de u_b que de \bar{u}_b . A l'aide de S4, nous faisons commuter les lettres de façon à les regrouper en couples (r_b, \bar{r}_b) et (u_b, \bar{u}_b) et nous les détruisons à l'aide de S3. Nous procédons à l'envers pour la réciproque. ♦

La transformation T3 permet de détruire ou d'insérer des boucles "blanches" dans un mot de Π^* .

- De S', nous pouvons également déduire les deux transformations suivantes:

$$T4 = \{ a\bar{a}a \leftrightarrow a / a \in \Pi' \}$$

$$T5 = \{ bb \leftrightarrow b / b \in \mathbb{B} \}$$

Preuve:

$$T4: a\bar{a}a \xleftrightarrow{S2, S2'} a\bar{a}a_b \xleftrightarrow{S2, S2'} a\bar{a}_b a_b \xleftrightarrow{S3, S3'} a.$$

$$T5: bb \xleftrightarrow{T1} bb_b \xleftrightarrow{T3} b. \blacklozenge$$

Nous en déduisons que le système S (du chapitre 2) peut être simulé à l'aide des règles du système S'. Par conséquent, pour tout mot $w \in \Pi^*$, si nous considérons les segments de $seg_b(w)$ fixés (en prenant $dpic(w) = (pic(w) \cup seg_b(w), e(w))$), nous pouvons appliquer l'algorithme du corollaire 2.1.12 pour dériver w en un mot minimal. Mais la difficulté est justement la suivante: Comment choisir judicieusement les segments parcourus de manière invisible?

3.2.2 Le système S' est complet

Nous montrons qu'à partir d'un mot $w \in \Pi^*$, nous pouvons dériver tous les mots décrivant la même figure que w (indépendamment de l'ensemble des segments parcourus en mode levé).

Théorème 3.2.2:

Pour tout mot $w \in \Pi^$, l'ensemble des mots dérivés de w par S' est l'ensemble des mots de Π^* décrivant la même figure que w .*

$$Im_{S'}(w) = \{ v \in \Pi^* / dpic(v) = dpic(w) \}$$

Preuve:

- . Puisque le système S' conserve les figures décrites, nous avons:
 $Im_{S'}(w) \subset \{ v \in \Pi^* / dpic(v) = dpic(w) \}$.
- . Pour la réciproque, il suffit de montrer que si deux mots w_1 et w_2 de Π^* décrivent la même figure, alors w_1 se dérive en w_2 par S' .

Soient deux mots de Π^* , w_1 et w_2 , décrivant la même figure. Nous pouvons supposer que w_1 et w_2 sont optimaux [lemme 3.2.1].

Soit $v_b \in \Pi_b^*$ tel que $e(v_b) = e(w_1) = e(w_2)$: les mots $\bar{w}_{1b}v_b$ et $\bar{w}_{2b}v_b$ sont des boucles de Π_b^* . Par conséquent, nous pouvons dériver w_1 et w_2 ainsi: $w_1 \xrightarrow{T_3} w_1' = w_1 \bar{w}_{2b} v_b$ et $w_2 \xrightarrow{T_3} w_2 \bar{w}_{1b} v_b \xrightarrow{S_1} w_2' = w_{1b} \bar{w}_2 v_b$.

Pour chaque segment s de $pic(w_1) = pic(w_2)$, il existe une unique lettre a_1 dans w_1 et une unique lettre a_2 dans w_2 engendrant le passage visible sur ce segment. Posons $w_1 = v_1 a_1 v_1'$ et $w_2 = v_2 a_2 v_2'$ avec $a_1, a_2 \in \Pi$ et $v_1, v_1', v_2, v_2' \in \Pi^*$.

Nous vérifions facilement que dans $w_1 \bar{w}_{2b}$, les lettres a_1 et \bar{a}_{2b} parcourent le même segment. En utilisant la transformation T_2 , nous obtenons:

$$w_1 \bar{w}_{2b} = v_1 a_1 v_1' \bar{v}'_{2b} \bar{a}_{2b} \bar{v}_{2b} \xrightarrow{T_2} v_1 a_1 v_1' \bar{v}'_{2b} \bar{a}_2 \bar{v}_{2b} \xrightarrow{T_2} v_1 a_{1b} v_1' \bar{v}'_{2b} \bar{a}_2 \bar{v}_{2b}$$

et en répétant cette opération pour tous les segments de $pic(w_1) = pic(w_2)$, nous en déduisons que: $w_1 \bar{w}_{2b} \xrightarrow{*} w_{1b} \bar{w}_2$. Par conséquent, $w_1 \xrightarrow{*} w_2$.

Comme le système S' est symétrique, nous avons:

$$w_1 \xrightarrow{*} w_1' \xrightarrow{*} w_2' \xrightarrow{*} w_2. \diamond$$

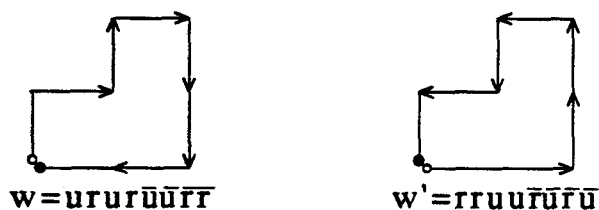
Remarque: Le système S' est complet dans le sens où il permet de dériver à partir de tout mot $w \in \Pi^*$, tous les mots décrivant la même figure. Notons que pour satisfaire cette propriété,

les six familles de règles de S' sont nécessaires.

Aucune d'entre elles ne peut être simulée par les cinq autres: elles sont indépendantes.

Nous justifions cette affirmation à l'aide d'exemples simples pour lesquels les mots w et w' décrivent la même figure.

Exemple 1:



Il est impossible de dériver w en w' sans utiliser la règle S1: en effet, les règles de $S2 \cup S2' \cup S3 \cup S3' \cup S4$ ne peuvent pas modifier la première lettre (qui est "non blanche") du mot à dériver.

Exemple 2:



Il est impossible de dériver w en w' (resp. w' en w) sans utiliser la règle S2 (resp. S2'): en effet, seule la règle S2 (resp. S2') permet de diminuer (resp. augmenter) d'une unité le nombre de passages visibles sur un segment.

Exemple 3:



Il est impossible de dériver w en w' (resp. w' en w) sans utiliser la règle S3 (resp. S3'): en effet, seule la règle S3 (resp. S3') permet de diminuer (resp. augmenter) la longueur du mot dérivé.

Exemple 4:



Il est impossible de dériver w en w' sans utiliser la règle S4: en effet, seule la règle S4 modifie la parité de la multiplicité des segments engendrés.

Avant l'étude de procédés améliorant la description des figures par des mots de Π^* , nous montrons qu'il n'existe pas de système de réécriture complet ayant un nombre fini de règles.

3.2.3 Pas de système fini

La proposition suivante s'applique à tout alphabet P dont les lettres sont associées à des déplacements dans le plan (il peut s'agir, par exemple, des alphabets Π ou Π' définis précédemment).

- Un système de réécriture σ de $P^* \times P^*$ est *complet* s'il permet de dériver à partir de tout mot $w \in P^*$, tous les mots décrivant la même figure.

Proposition 3.2.3:

Il n'existe pas de système de réécriture conservant les figures et complet ayant un nombre fini de règles.

Pour prouver cette proposition, nous introduisons quelques définitions et un lemme intermédiaire.

- Quelques définitions supplémentaires:

Si σ est un système de réécriture ayant un nombre fini de règles, alors il existe $p > 0$ tel que: $\forall (u \rightarrow v) \in \sigma, |u| \leq p$ et $|v| \leq p$.

Nous dirons que σ est *borné par p* .

$\forall w, w_1, w_2 \in P^*$,

$\text{seg}[w_1, w_2]$: est l'ensemble des segments parcourus au moins une fois de manière visible par le facteur w_2 en débutant le tracé au point $e(w_1)$.

$\forall s, s' \in \text{pic}(w)$:

$s <_w s'$: s est avant s' dans w s'il existe une décomposition de w vérifiant:

$$w = v_1 x v_2 x' v_3 \quad \text{avec } x, x' \in P$$

$$\{s\} = \text{seg}[v_1, x]$$

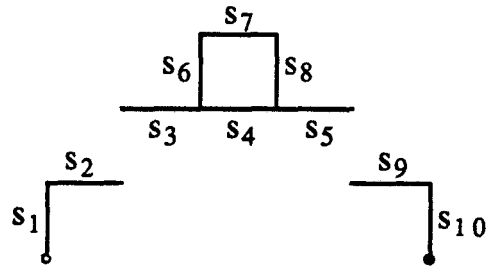
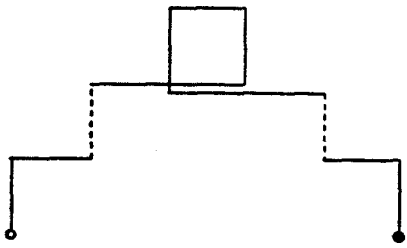
$$\{s'\} = \text{seg}[v_1 x v_2, x']$$

$\text{eloig}(s, s')$: l'éloignement de s à s' est la longueur d'un plus petit mot de P^* reliant s à s' :

$$\text{eloig}(s, s') = \min \{ |v_2| \mid \{s\} = \text{seg}[v_1, x], \{s'\} = \text{seg}[v_1 x v_2, x'], \\ v_1, v_2 \in P^* \text{ et } x, x' \in P \}$$

Exemple:

$$w = uru_b rru \bar{r} r \bar{u} r r \bar{u}_b r \bar{u}$$



$$\text{seg}[uru_b, rru] = \{s_3, s_4, s_8\}$$

$$s_2 <_w s_4, \quad s_4 <_w s_7, \quad s_7 <_w s_4,$$

$$\text{eloig}(s_2, s_8) = 3, \quad \text{eloig}(s_1, s_{10}) = 5.$$

- Un résultat intermédiaire:

Nous montrons que si un mot $w \in P^*$ engendre deux segments "assez éloignés", l'ordre dans lequel ils sont parcourus est invariant au cours des dérivations par un système borné conservant les figures.

Lemme 3.2.4:

Soient σ un système conservant les figures, borné par p , $p > 0$, $w \in P^$, $s, s' \in \text{pic}(w)$ avec $\text{eloig}(s, s') \geq p$, $w' \in \text{Im}_\sigma(w)$:*

$$(s <_w s') \Rightarrow (s <_{w'} s')$$

Preuve:

Il suffit de démontrer ce résultat après une étape de dérivation. D'après les hypothèses, nous pouvons décomposer w ainsi: $w = v_1 x v_2 x' v_3$ avec $x, x' \in P$, $\{s\} = \text{seg}[v_1, x]$ et $\{s'\} = \text{seg}[v_1 x v_2, x']$. Puisque $\text{eloig}(s, s') \geq p$, nous avons nécessairement $|v_2| \geq p$. Il est donc impossible en appliquant une règle de σ d'affecter à la fois x et x' . Supposons que le passage de w à w' ait modifié un facteur de $v_1 x v_2$ (le raisonnement est similaire pour $v_2 x' v_3$). Comme le système σ conserve les figures, ce nouveau facteur parcourt au moins une fois le segment s de manière visible: nous obtenons $w' = v'_1 z v'_2 x' v_3$ avec $\{s\} = \text{seg}[v'_1, z]$ et $\{s'\} = \text{seg}[v'_1 z v'_2, x']$.

Par conséquent, $s <_{w'} s'$. ♦

Preuve de la proposition 3.2.3:

Supposons que σ soit un système conservant les figures, borné par $p > 0$.

Soit $w = r^n \bar{r}^n r^n$ avec $n > p + 1$. La figure décrite par w est:



Nous vérifions que $m = r^n$ est le mot minimal décrivant cette figure.

Intéressons-nous à deux décompositions particulières de w :

- (1) $w = r r^{n-1} \bar{r} \bar{r}^{n-1} r^n$ avec $\{s_1\} = \text{seg}[\epsilon, r]$ et $\{s_n\} = \text{seg}[r r^{n-1}, \bar{r}]$
 (2) $w = r^n \bar{r} \bar{r}^{n-1} r r^{n-1}$ avec $\{s_1\} = \text{seg}[r^n \bar{r}^{n-1}, r]$ et $\{s_n\} = \text{seg}[r^n, \bar{r}]$.

Par définition, nous avons $s_1 <_w s_n$ et $s_n <_w s_1$. Comme $\text{eloig}(s_1, s_n) \geq p$, le lemme 3.2.4 nous donne: $\forall w' \in \text{Im}_\sigma(w)$, $s_1 <_{w'} s_n$ et $s_n <_{w'} s_1$.

Ce résultat signifie que tout mot dérivé de $w = r^n \bar{r}^n r^n$ avec $n > p + 1$ par un tel système voit son tracé passer au moins deux fois sur l'un des segments s_1 ou s_n : par conséquent, $m = r^n \notin \text{Im}_\sigma(w)$. Nous en déduisons que le système σ n'est pas complet. ♦

3.2.4 Méthode de simplification

Nous établissons quelques résultats intermédiaires pour faciliter l'énoncé de la méthode. Le premier résultat est une extension du lemme 2.1.2 à l'alphabet Π' .

Lemme 3.2.5:

Soit $w \in \Pi'^$. Pour tout segment $s \in \text{pic}(w) \cup \text{seg}_b(w)$ tel que $\text{mult}_w(s) \geq 3$, il existe w' , dérivé de w par S' , vérifiant $|w'| = |w| - 2$ tel que:*

- . $\text{mult}_{w'}(s) = \text{mult}_w(s) - 2;$
- . $\text{mult}_{w'}(s') = \text{mult}_w(s')$ pour tout $s' \in \text{pic}(w) \cup \text{seg}_b(w) - \{s\}$.

Preuve:

Soient $w \in \Pi'^*$ et $s \in \text{pic}(w) \cup \text{seg}_b(w)$, un segment tel que $\text{mult}_w(s) \geq 3$. Si $s \in \text{pic}(w)$, nous dérivons w à l'aide de T1 et T2 de façon à ce que les trois premiers passages sur s s'effectuent de manière visible; sinon les trois premiers passages sur s sont nécessairement invisibles. Comme les règles R1 et R2 de S étendues à l'alphabet Π' peuvent être simulées par S' , nous nous reportons à la preuve du lemme 2.1.2 pour conclure. ♦

Deux corollaires de ce lemme (justifiés précédemment au § 2.1.2):

Corollaire 3.2.6:

Tout mot minimal est réduit.

Corollaire 3.2.7:

Pour tout mot $w \in \Pi^$, il existe un mot réduit, dérivé de w par S' .*

Quelques définitions:

- . Un facteur f d'un mot $w \in \Pi^*$ est *invisible* si $f \in \Pi_b^*$.
- . Un facteur invisible f d'un mot $w \in \Pi^*$ est *direct* si $f \in (r_b + u_b)^* + (r_b + \bar{u}_b)^* + (\bar{r}_b + u_b)^* + (\bar{r}_b + \bar{u}_b)^*$.

Lemme 3.2.8:

Pour tout mot $w \in \Pi^$, il existe un mot dérivé de w par S' , tel que tous ses facteurs invisibles soient directs.*

Preuve:

Pour tout facteur invisible de w , nous faisons commuter ses lettres par S_4 de façon à regrouper les lettres opposées et nous les effaçons par S_3 . ♦

Lemme 3.2.9:

Soit w , un mot réduit de Π^ . Pour tout segment $s \in \text{seg}_b(w)$ parcouru deux fois dans le même sens par w , il existe un mot w' , dérivé de w par S' , vérifiant $|w'| = |w| - 2$ tel que:*

- . $\text{mult}_w(s) = 0$;
- . $\text{mult}_w(s') = \text{mult}_{w'}(s')$ pour tout $s' \in \text{pic}(w) \cup \text{seg}_b(w) - \{s\}$.



Preuve:

Soit w , un mot réduit de Π^* dont le tracé parcourt le segment $s \in \text{seg}_b(w)$ deux fois dans le même sens. Posons $w = w_1 a_b w_2 a_b w_3$ avec $a_b \in \Pi_b$, $w_1, w_2, w_3 \in \Pi^*$ et $w_2 a_b \in \mathbb{B}$. Nous dérivons w ainsi:

$$w = w_1 a_b w_2 a_b w_3 \xrightarrow{S_1} w_1 a_b \bar{a}_b \bar{w}_2 w_3 \xrightarrow{S_3} w_1 \bar{w}_2 w_3.$$

Lors de cette dérivation, la multiplicité de s diminue de 2 et la multiplicité des autres segments de $\text{pic}(w) \cup \text{seg}_b(w)$ est inchangée. ♦

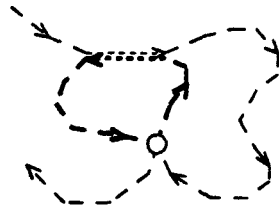
D'où le corollaire suivant:

Corollaire 3.2.10:

Pour tout mot réduit $w \in \Pi^$, il existe un mot w' , dérivé de w par S' , tel que: si un segment $s \in \text{seg}_b(w')$ est parcouru deux fois par w' , ce segment relie nécessairement deux composantes connexes (distinctes) de la figure.*

Preuve:

Soit w , un mot réduit de Π^* . Supposons qu'il existe un segment $s \in \text{seg}_b(w)$ parcouru deux fois par w , reliant deux sommets d'une même composante connexe.



Dans ce cas, la lettre de w engendrant le premier ou le deuxième passage sur s appartient à une boucle (ne parcourant qu'une seule fois le segment s). Si cela est nécessaire, nous dérivons w par S_1 de façon à obtenir les deux passages sur s dans le même sens et nous appliquons le lemme 3.2.9: le segment s ne sera plus parcouru par le mot dérivé obtenu. ♦

Les opérations décrites à travers ces lemmes agissent localement sur les mots de figure. Or, nous savons qu'il est nécessaire d'élargir notre champ d'action et d'envisager des procédés globaux si nous espérons trouver un mot minimal dans le cas général. C'est pourquoi, nous avons essayé de mettre au point un procédé s'appuyant (comme dans le chapitre précédent) sur la recherche de cycles à améliorer.

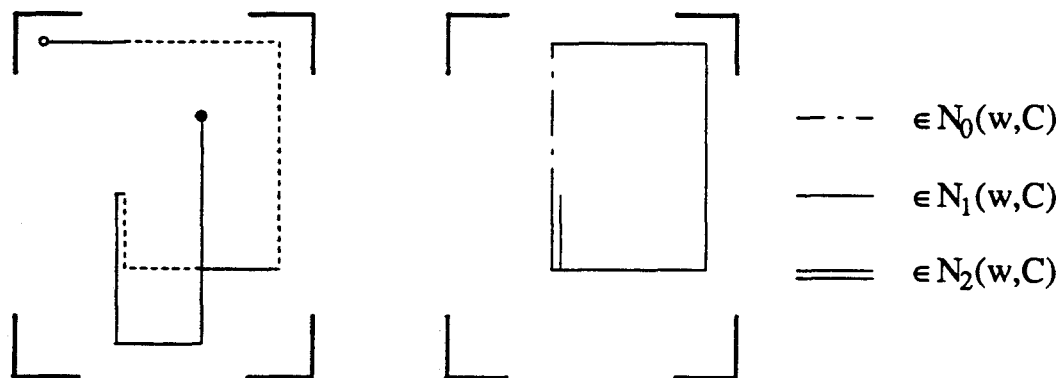
Nous complétons les définitions établies pour le lemme 2.1.8. Soit $w \in \Pi'^*$:

- $\text{Rect}(w)$ représente le plus petit rectangle dont les cotés sont parallèles aux axes et qui contient tous les segments parcourus (de manière visible ou invisible) par w .
- $\text{seg}_n(w)$ est l'ensemble des segments situés à l'intérieur de $\text{Rect}(w)$ qui ne sont jamais parcourus (de manière visible ou invisible) par w .

Soit C , un cycle issu de $\text{pic}(w) \cup \text{seg}_b(w) \cup \text{seg}_n(w)$. Nous notons:

- $N_0(w,C)$, l'ensemble des segments de C , non parcourus par w .
- $N_1(w,C)$, l'ensemble des segments de C , de multiplicité égale à 1 relativement à w .
- $N_2(w,C)$, l'ensemble des segments de C , de multiplicité égale à 2 relativement à w .
- $F(w,C)$, le plus long facteur invisible de w décrivant une chaîne constituée de segments appartenant à $N_1(w,C) \cap \text{seg}_b(w)$.

Exemple:



$w = rr_b r_b \bar{u}_b \bar{u}_b \bar{u}_b \bar{r}_b \bar{r}_b u_b \bar{u}_b r u u u$ Un cycle C : $F(w,C)$ est en italique dans w .

Lemme 3.2.11:

Pour tout mot réduit $w \in \Pi'^*$ et pour tout cycle C issu de $\text{pic}(w) \cup \text{seg}_b(w) \cup \text{seg}_n(w)$ vérifiant :

$$(*) \quad k = [2 * |F(w,C)| + |N_2(w,C)|] - [|N_0(w,C)| + |N_1(w,C)|] > 0,$$

il existe un mot w' , dérivé de w par S' , strictement plus court que w .
(plus précisément, $|w'| = |w| - k$)

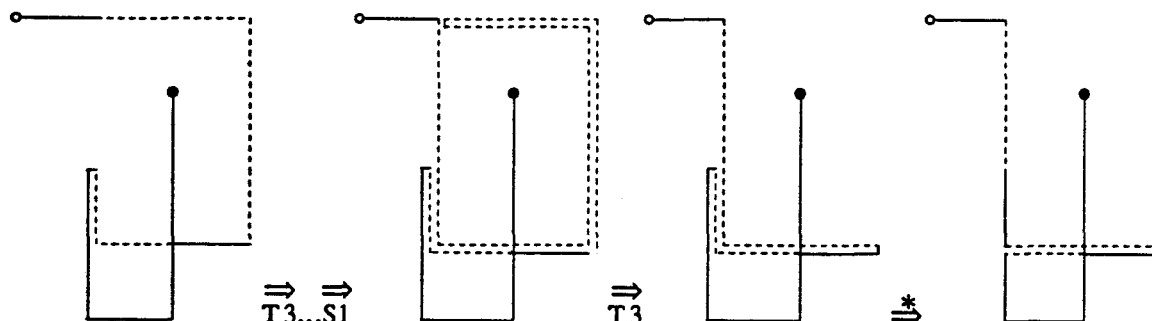
Preuve:

Soit w , un mot réduit de Π'^* et C , un cycle issu de $\text{pic}(w) \cup \text{seg}_b(w) \cup \text{seg}_n(w)$ vérifiant (*). Nous pouvons supposer que la chaîne décrite par $F(w,C)$ débute le cycle C . Posons $f=F(w,C)$ et $c \in \Pi_b^*$ tel que fc décrive le cycle C (de manière invisible): nous obtenons $w=w_1fw_2$ avec $w_1, w_2 \in \Pi'^*$ et $fc \in \Pi_b^* \cap \mathbb{B}$. Dérivons w de la façon suivante:

$$w = w_1fw_2 \xrightarrow{T_3} w_1fcfw_2 \xrightarrow{S_1} w_1\bar{c}ffw_2 \xrightarrow{T_3} w_1\bar{c}w_2.$$

Tous les segments de $N_2(w,C)$ sont maintenant de multiplicité égale à 3. En appliquant le lemme 3.2.5 à ces segments, nous obtenons un mot dérivé dont la longueur a augmenté de $|fc| = |N_0(w,C)| + |N_1(w,C)| + |N_2(w,C)|$ et a diminué de $2*|F(w,C)| + 2*|N_2(w,C)|$. Comme le cycle C et le mot w vérifient (*), nous savons que le gain est strictement positif: nous obtenons un mot strictement plus court que w de k . ♦

En clair, sur l'exemple précédent:



$$k = 2*|F(w,C)| + 2*|N_2(w,C)| - |N_0(w,C)| - |N_1(w,C)| - |N_2(w,C)|$$

$$k = 2*5 + 2*1 - 2 - 7 - 1 = 2$$

A partir des outils que nous venons d'introduire, nous énonçons une méthode pour dériver un mot de Π'^* par S' de façon à "améliorer" le tracé associé.

3.3 Le cas particulier des figures connexes

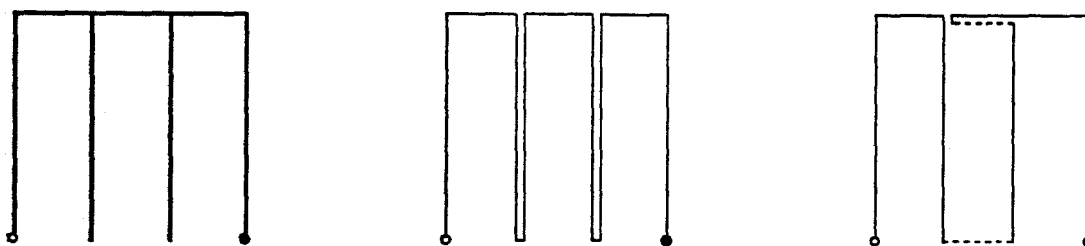
Etant donné une figure connexe, nous établissons une relation entre ses différents paramètres et nous déterminons la longueur d'un mot minimal de Π^* décrivant cette figure. Ceci nous permet de caractériser les descriptions minimales des figures connexes et nous nous intéressons au rapport "longueur d'un mot minimal" / "nombre de segments", pour des figures connexes quelconques.

3.3.1 Quelques définitions

Soit F , une figure connexe avec points de départ et d'arrivée fixés. Nous définissons:

- . $W(F)$: le nombre de segments de la figure F ;
- . $l(F)$: la longueur d'un mot minimal de Π^* décrivant F ;
- . $l'(F)$: la longueur d'un mot minimal de Π'^* décrivant F .

Exemple:



F , une figure connexe Un mot minimal sur Π^* Un mot minimal sur Π'^*
 $W(F)=15$ $l(F)=21$ $l'(F)=17$

3.3.2 Une inégalité

Dans ce paragraphe, nous établissons une relation entre les différents paramètres d'une figure connexe F .

Inégalité 3.3.1:

Soit F , une figure connexe dont le point de départ est l'origine du repère O et le point d'arrivée, le point $T(x,y)$. Nous avons:

$$|X| + |Y| \leq W(F) \leq l'(F) \leq l(F) \leq 2 * W(F) - (|X| + |Y|).$$

Les trois premières inégalités sont évidentes. Démontrons seulement la dernière. Pour cela, nous utiliserons le lemme suivant:

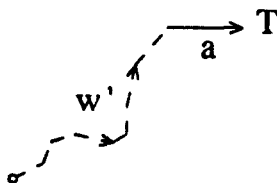
Lemme 3.3.2:

Pour tout mot réduit $w \in \Pi^$ décrivant F du point O au point T , il existe une chaîne allant de O à T constituée de segments parcourus une seule fois par w . Une telle chaîne est appelée une traverse de w .*

Preuve:

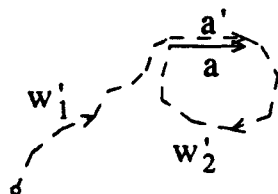
La démonstration s'effectue par récurrence sur $|w|$.

- . Si $|w|=1$, il existe un seul segment parcouru une seule fois allant de O à T .
- . Supposons la propriété vraie pour tout mot de longueur inférieure ou égale à n et montrons qu'elle est encore vérifiée pour $n+1$. Soit $w \in \Pi^*$ avec $|w|=n+1$ et posons $w=w'a$ avec $a \in \Pi$. Deux cas sont possibles:
 - La dernière lettre de w engendre un segment s non parcouru par w' : ce segment est parcouru une seule fois. D'après l'hypothèse de récurrence, nous savons qu'il existe une traverse de w' . En y juxtaposant s , nous obtenons une traverse de w .



- La dernière lettre de w parcourt un segment s déjà engendré par w' : Posons $w'=w_1'a'w_2'$ avec $a' \in \{a, \bar{a}\}$ engendrant le segment s .

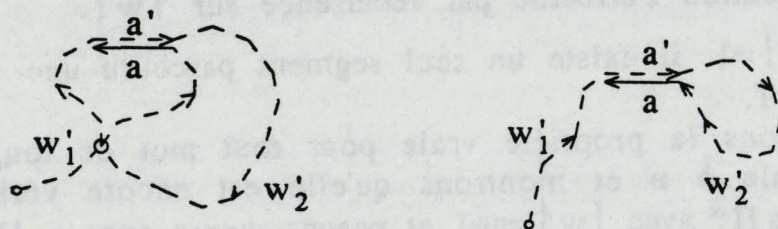
. Si $a'=a$ alors $a'w_2' \in \mathbb{B}$ et le mot $v=w_1'\bar{w}_2'\bar{a}a$ est équivalent à w pour la recherche d'une traverse (en effet, le mot v peut être obtenu en dérivant w par la règle R1 du système S qui conserve les figures décrites et la multiplicité des segments). Comme $|w_1'\bar{w}_2'| \leq n$, l'hypothèse de récurrence nous permet de dire qu'il existe une traverse t de $w_1'\bar{w}_2'$. Cette chaîne est également une traverse pour v : la lecture du facteur $\bar{a}a$ ne peut en aucun cas augmenter la multiplicité d'un segment de t (car v est réduit).



. Si $a' = \bar{a}$ alors $w_2' \in \mathcal{B}$.

Soit il existe une boucle dans w' contenant a' : le mot obtenu en "renversant" cette boucle par R1 est équivalent à w pour la recherche d'une traverse. Nous nous retrouvons dans le cas précédent.

Soit il n'existe pas de boucle dans w' contenant a' : le tracé de w_2' ne rencontre jamais celui de w_1' . Par conséquent, la multiplicité des segments parcourus par w_1' est conservée dans w : une traverse de w_1' sera également une traverse de w . D'après l'hypothèse de récurrence, nous savons que celle-ci existe.



Fin de la preuve du lemme. ♦

Preuve de l'inégalité 3.3.1:

D'après le corollaire 2.1.3, nous savons que tout mot minimal de Π^* est réduit (chaque segment est parcouru au plus deux fois). Par conséquent: $l(F) \leq 2 * W(F)$.

D'après le lemme 3.3.2, pour tout mot réduit de Π^* décrivant F de O à T , il existe au moins $|X| + |Y|$ segments parcourus une seule fois (en effet, la plus courte chaîne reliant les points O à T contient $|X| + |Y|$ segments). Par conséquent: $l(F) \leq 2 * W(F) - (|X| + |Y|)$. ♦

Les cas limites:

Existe-t-il des figures qui justifient que les inégalités soient exprimées au sens large?

$$|X| + |Y| \leq W(F) \leq l'(F) \leq l(F) \leq 2 * W(F) - (|X| + |Y|).$$

- Pouvons-nous avoir $|X| + |Y| = W(F)$?

Dans ce cas, nous obtenons:

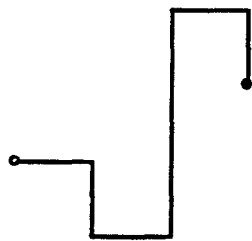
$$|X| + |Y| = W(F) \leq l'(F) \leq l(F) \leq |X| + |Y|$$

et par conséquent:

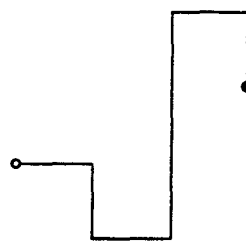
$$|X| + |Y| = W(F) = l'(F) = l(F) = |X| + |Y|$$

Une telle figure peut être décrite par un mot contenant autant de lettres qu'il y a de segments dans la figure: tous ces segments sont parcourus une seule fois.

Exemple:



La figure F



Un tracé minimal sur Π^* et Π'^*

- Pouvons-nous avoir $W(F) = l'(F)$?

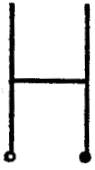
Tous les segments de F doivent être parcourus au moins une fois de manière visible. Cette égalité signifie qu'il existe un mot de Π'^* contenant $W(F)$ lettres dont au moins $W(F)$ sont issues de Π . Par conséquent: $W(F) = l'(F) = l(F)$.

Il s'agit du même type de figure que dans le cas précédent.

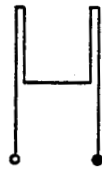
- Pouvons-nous avoir $l'(F) = l(F)$?

Dans ce cas, le lever de crayon ne permet pas une description plus courte de la figure proposée.

Exemple:

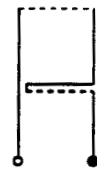


La figure F



Un tracé minimal sur Π^*

$$l(F)=7$$



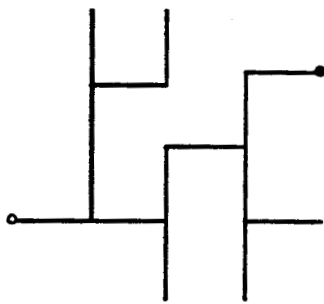
Un tracé minimal sur Π'^*

$$l'(F)=7$$

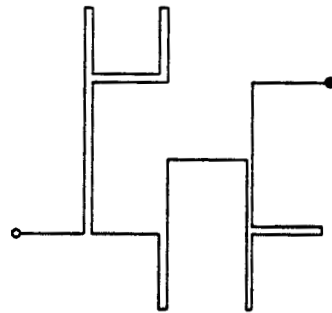
- Pouvons-nous avoir $l(F) = 2*W(F) - (|X| + |Y|)$?

Dans ce cas, tous les segments de F sont parcourus deux fois sauf une chaîne "directe" de O à T.

Exemple:



La figure F



Un tracé minimal sur Π^*

3.3.3 Caractérisation d'un mot minimal de Π^*

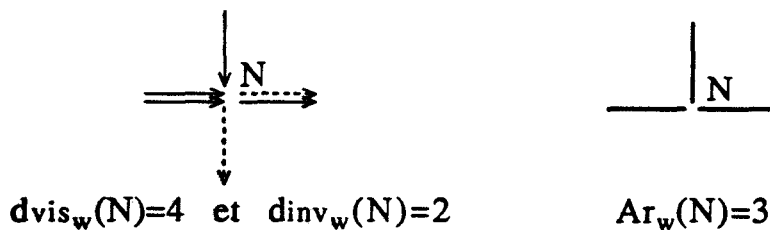
Dans ce cas particulier des figures connexes, nous montrons que le tracé d'un mot minimal de Π^* doit vérifier certaines conditions. Celles-ci nous permettent de déterminer la longueur d'un mot minimal de Π^* pour une figure connexe donnée avec points de départ et d'arrivée fixés.

Quelques définitions:

Soit w , un mot de Π^* et notons F , la figure décrite associée.

- Le *degré visible* d'un noeud N relatif à un mot w de Π^* , $dvis_w(N)$, est le nombre de passages visibles arrivant en N ou partant de N , lors du tracé de w .
Par convention: Pour les points de départ et d'arrivée du tracé, la valeur du degré visible est majorée de 1 (pour simuler l'arrivée du crayon avant le tracé et le départ du crayon après le tracé).
- Le *degré invisible* d'un noeud N relatif à un mot w de Π^* , $dinv_w(N)$, est le nombre de passages invisibles arrivant en N ou partant de N , lors du tracé de w .
- L'*arité* d'un noeud N relatif à un mot w de Π^* , $Ar_w(N)$, est le nombre de segments de $pic(w)$ ayant N comme extrémité.
Par convention: Pour les points de départ et d'arrivée du tracé, la valeur de l'arité est majorée de 1 (pour simuler l'arrivée du crayon avant le tracé et le départ du crayon après le tracé).

Exemple:



- La *distance* entre deux noeuds N et N' de coordonnées (x,y) et (x',y') est définie par: $dist(N,N')=|x-x'|+|y-y'|$
- Une chaîne est *directe* si elle peut être décrite par un mot $w \in (a+a')^*$ avec $a \in \{r, \bar{r}\}$ et $a' \in \{u, \bar{u}\}$ (définition p.30). Notons que la longueur d'une telle chaîne est égale à la distance entre ses extrémités.

Soit w , un mot minimal de Π'^* décrivant F . Sans perte de généralité, nous pouvons supposer que w est optimal [lemme 3.2.1]. Voici quelques remarques sur le tracé de w :

Fait 3.3.3:

Les segments parcourus de manière invisible peuvent être regroupés en chaînes directes dont les extrémités sont des noeuds d'arité impaire.

Preuve:

. Considérons le graphe obtenu de la façon suivante: les arêtes sont les segments parcourus de manière invisible (comme F est connexe, chaque segment est parcouru au plus une fois de manière invisible par un mot minimal [corollaire 3.2.6 & 3.2.10]).

D'après le théorème d'Euler [Bc], nous pouvons *recouvrir ce graphe (chaque arête exactement une fois) par des chaînes dont les extrémités sont des sommets de degré invisible impair*. Nous éliminons la possibilité où l'une de ces chaînes puisse être un cycle: dans ce cas, il serait possible de parcourir le tracé de w (en renversant certaines portions) de façon à ce que tous les segments de ce cycle soient décrits successivement. Ce nouveau parcours serait engendré par un mot de Π'^* contenant une boucle de Π_b^* qui pourrait être éliminée (transformation T3) et nous obtiendrions un mot strictement plus court que w , décrivant la même figure. Contradiction.

Ces chaînes sont nécessairement *directes*: de manière analogue, nous pourrions parcourir ces chaînes en une seule fois. Si l'une de ces chaînes n'était pas directe, il existerait une description plus courte.

. D'autre part, pour tout noeud N , le tracé arrive en N autant de fois qu'il en repart (même pour les points de départ et d'arrivée d'après la convention). Par conséquent, $d_{vis_w}(N) + d_{inv_w}(N)$ est pair. Comme le mot w est optimal, nous avons $d_{vis_w}(N) = Ar_w(N)$ et nous en déduisons que $d_{inv_w}(N)$ et $Ar_w(N)$ ont la même parité. ♦

Fait 3.3.4:

Ces chaînes joignent tous les noeuds d'arité impaire de la figure.

Preuve:

Pour tout noeud N d'arité impaire de la figure, nous savons que $d_{inv_w}(N)$ est un nombre impair. Par conséquent, il existe au moins un segment parcouru de manière invisible contenant N . ♦

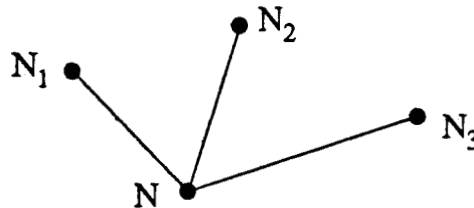
A partir de ces remarques, nous pouvons déterminer la longueur d'un mot minimal de Π^* décrivant une figure donnée.

Méthode 3.3.5:

Soit F , une figure connexe avec points de départ et d'arrivée fixés.

- . Repérer les noeuds d'arité impaire de la figure.
- . Les relier par des chaînes directes (invisibles) de façon à ce que la somme de leur longueur soit minimale (nous la noterons $Som(F)$).

Remarque: Nous pouvons ne considérer que les chaînes qui relient ces noeuds deux par deux, les autres cas n'apportant aucune amélioration: en effet, supposons qu'un noeud N soit relié à trois noeuds N_1 , N_2 et N_3 (le degré invisible doit être un nombre impair supérieur à un).



De façon évidente, nous avons:

$$\text{dist}(N, N_1) + \text{dist}(N, N_2) + \text{dist}(N, N_3) \leq \text{dist}(N_1, N_2) + \text{dist}(N, N_3).$$

- . La longueur d'un mot minimal de Π^* décrivant cette figure est donnée par:

$$l(F) = W(F) + Som(F).$$

Remarque: Ce raisonnement nous permet de construire un mot minimal de Π^* décrivant F : en ajoutant à la figure F ces chaînes directes (invisibles), nous obtenons un graphe dont tous les sommets ont un degré (visible+invisible) pair. D'après le théorème d'Euler, nous savons parcourir ce graphe en passant exactement une fois sur chaque arête.

3.3.4 Relation entre la longueur d'un mot minimal et sa figure de base

Nous savons que toute figure F peut être décrite par un mot réduit. Par conséquent, la longueur d'un mot minimal est majorée par deux fois le nombre de segments de la figure:

$$l'(F) \leq l(F) \leq 2*W(F)$$

Dans cette section, nous abordons un problème posé par G. Paun [Pa]:

Existe-t-il des figures connexes telles qu'en choisissant les "meilleurs" points de départ et d'arrivée pour son tracé, le rapport "longueur d'un mot minimal" / "nombre de segments" soit toujours très proche de 2 ?

ou plus formellement

Notons $l_m(F)$ (resp. $l'_m(F)$), la longueur d'un mot minimal de Π^* (resp. Π'^*) obtenu en choisissant les points de départ et d'arrivée les plus avantageux (donnant une description de F la plus courte possible).

Existe-t-il des figures (ou plutôt des familles de figures (F_n)) telles que:

$$\lim_{n \rightarrow \infty} \frac{l'_m(F_n)}{W(F_n)} = 2 ?$$

Nous nous limitons, dans un premier temps, à la description d'une figure par un mot de Π^* (sans possibilité de lever le crayon).

Résultat 3.3.6:

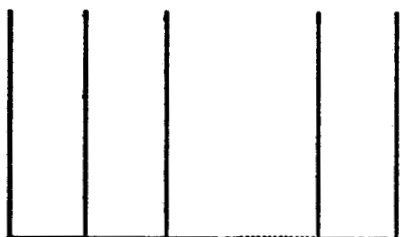
Il existe des familles de figures (F_n) telles que:

$$\lim_{n \rightarrow \infty} \frac{l_m(F_n)}{W(F_n)} = 2$$

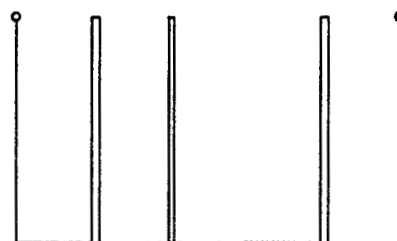
Preuve:

Citons l'exemple issu de [MRW]:

Soit (F_n) , la famille de figures définie par les mots $w_n = (\bar{u}^n r u^n)^n$.



Une figure (de base) F_n

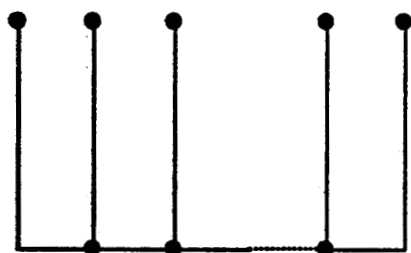


Un mot minimal de Π^*

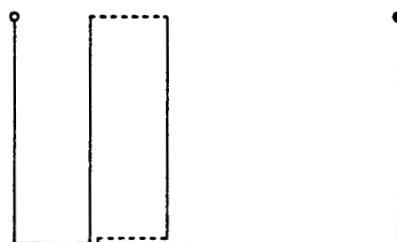
Nous avons $W(F_n) = n(n+1) + n = n^2 + 2n$ et $l_m(F_n) = W(F_n) + n(n-1) = 2n^2 + n$ et par conséquent:

$$\lim_{n \rightarrow \infty} \frac{l_m(F_n)}{W(F_n)} = 2 \diamond.$$

Que se passe-t-il si nous autorisons le déplacement en mode levé sur cet exemple ?



Une figure (de base) F_n



Un mot minimal de Π'^*

Remarque: Nous choisissons les points de départ et d'arrivée du tracé parmi les noeuds d'arité impaire (qui deviennent alors d'arité paire).

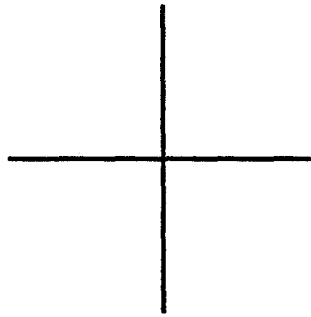
Nous avons $l'_m(F_n) = W(F_n) + (n-1) = n^2 + 3n - 1$ et par conséquent:

$$\lim_{n \rightarrow \infty} \frac{l'_m(F_n)}{W(F_n)} = 1.$$

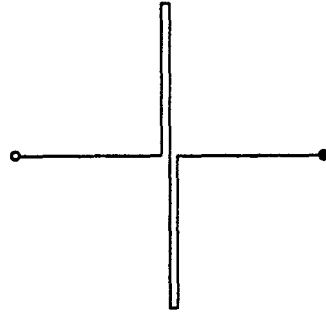
Dans ce cas, le déplacement en mode levé est très avantageux. Est-ce toujours ainsi ?

Citons l'exemple issu de [Pa]:

Soit (F_n) , la famille de figures définie par les mots $w_n = \Gamma^n U^n \bar{U}^{2n} U^n \Gamma^n$.



Une figure (de base) F_n

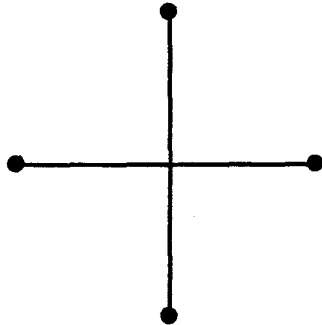


Un mot minimal de Π^*

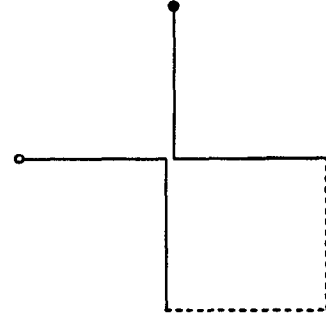
Nous avons $W(F_n) = 4n$ et $l_m(F_n) = W(F_n) + 2n = 6n$ et par conséquent:

$$\lim_{n \rightarrow \infty} \frac{l_m(F_n)}{W(F_n)} = 3/2 .$$

Que se passe-t-il si nous autorisons le déplacement en mode levé sur cet exemple ?



Une figure (de base) F_n



Un mot minimal de Π^*

Nous avons $l'_m(F_n) = W(F_n) + 2n = 6n$ et par conséquent:

$$\lim_{n \rightarrow \infty} \frac{l'_m(F_n)}{W(F_n)} = 3/2 .$$

Dans ce cas, le déplacement en mode levé n'apporte aucune amélioration.

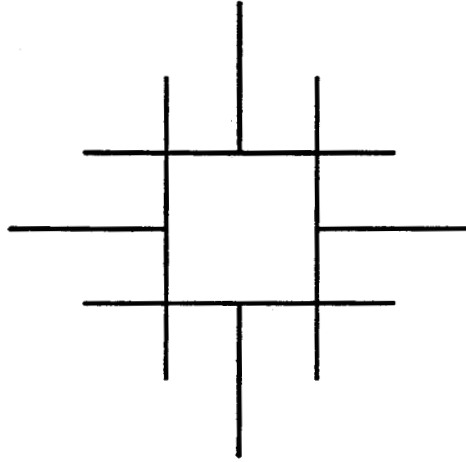
De nombreux exemples nous laissaient penser qu'il n'existait pas de figure F telle que:

$$\frac{l'_m(F)}{W(F)} > 3/2 .$$

Mais ...

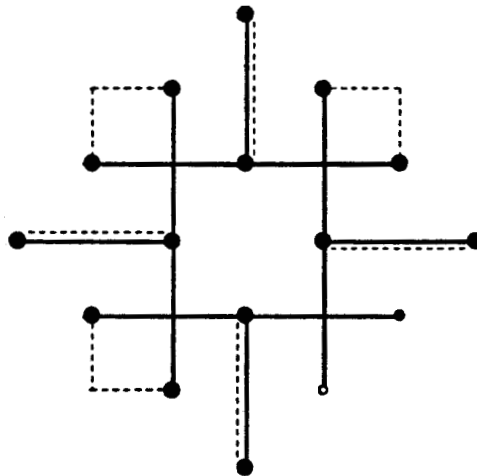
Mais ...

Soit F, la figure de base suivante:



Calculons $l'_m(F)$:

Cette figure comporte 16 noeuds d'arité impaire: Nous en retirons 2 pour les points de départ et d'arrivée du tracé et nous regroupons les 14 noeuds restants, deux par deux par des chaînes de longueur 2 (nous ne pouvons pas faire mieux).



Nous avons $W(F) = 24$ et $l'_m(F) = W(F) + 2 \cdot 14/2 = 38$ et par conséquent:

$$\frac{l'_m(F)}{W(F)} > 3/2 .$$

Nous émettons la conjecture suivante:

Conjecture 3.3.7:

Il n'existe pas de figure F telle que:

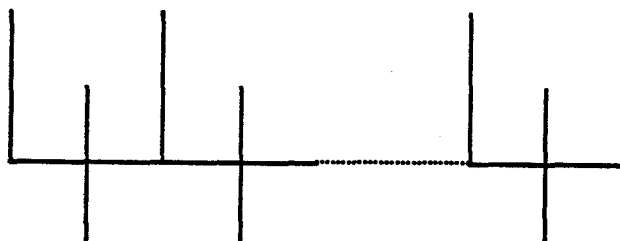
$$\frac{l'_m(F)}{W(F)} \geq 5/3 .$$

Remarque:

. Nous pouvons être aussi proches que nous le voulons de la borne $5/3$.

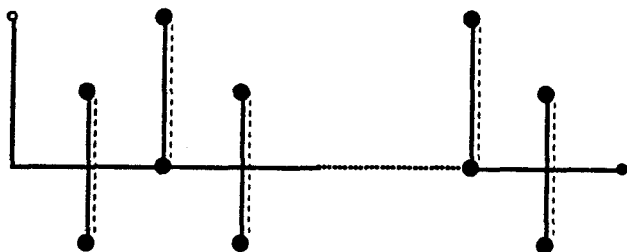
Exemple:

Soit (F_n) , la famille de figures définie par les mots $w_n = (uu\bar{u}\bar{u}ru\bar{u}\bar{u}r)^n$.



Une figure (de base) F_n

Pour chaque motif élémentaire, nous avons 4 points d'arité impaire (distants au moins de 2 unités les uns des autres) pour 6 segments. Pour obtenir un mot minimal, nous relierons $(4n-2)$ points d'arité impaire deux par deux, par des chaînes de longueur 2.



Nous avons $W(F_n) = 6n$ et $l'_m(F_n) = W(F_n) + 2(4n-2)/2 = 10n-2$ et par conséquent:

$$\lim_{n \rightarrow \infty} \frac{l'_m(F_n)}{W(F_n)} = 5/3 .$$

Chapitre 4:

FIGURE PARTICULIERE : CONTOUR DE POLYOMINO

- 1 - Définitions et propriétés
 - 1.1 Les polyominos
 - 1.2 Les mots de contour
- 2 - Quel est le problème ?
- 3 - La preuve
 - 3.1 Idée de la preuve
 - 3.2 Le système de réécriture R
 - 3.3 Le système de réécriture T
- 4 - Extension aux polyominos semi-convexes
 - 4.1 Quelques définitions
 - 4.2 Quel est le problème ?
 - 4.3 La preuve
- 5 - Autres extensions envisagées

FIGURE PARTICULIERE : CONTOUR DE POLYOMINO

1	1.1 Définitions et notations
2	1.2 Les polyominoes
3	1.3 Les mots de contour
4	1.4 Quel est le problème ?
5	1.5 Les preuves
6	2.1 L'ensemble de la preuve
7	2.2 Les contours de longueur 2
8	2.3 Les contours de longueur 3
9	2.4 Extension aux polyominoes semi-concaves
10	3.1 Quelques définitions
11	3.2 Quel est le problème ?
12	3.3 Les preuves
13	4.1 Autres extensions envisagées

Chapitre 4:

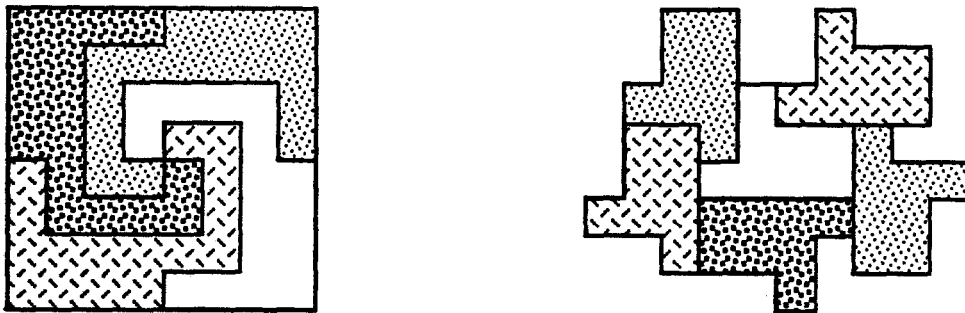
Figure particulière: Contour de polyomino

Un polyomino est une union finie de carrés unitaires juxtaposés formant une pièce sans trou. Il peut être caractérisé par un mot de figure décrivant son contour.



Un exemple de "polyomino usuel" pourrait être un carreau destiné à recouvrir le sol de votre maison. Mais attention, choisir une forme originale pour carrelé votre intérieur vous permettra-t-il de recouvrir correctement votre sol (sans trou) ?

Exemple [St]:



D. Beauquier et M. Nivat [BN] ont trouvé une condition simple sur les mots de contour de polyominos permettant de décider si un polyomino est apte à paver le plan.

Dans cette optique, les problèmes de décidabilité concernant les polyominos et leurs mots de contour suscitent un certain intérêt. D. Beauquier [Bd] a démontré que nous ne pouvons pas décider si un langage rationnel $R \subset \Pi^*$ contenait un mot de contour de polyomino. Nous nous posons ici la même question en limitant le problème à certaines classes de polyominos et nous y répondons positivement.

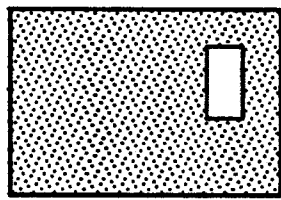
4.1 Définitions et propriétés

Pour démontrer ces résultats, nous avons besoin de quelques outils supplémentaires.

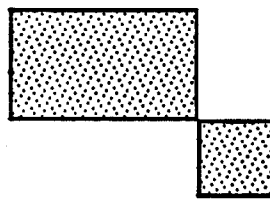
4.1.1 Les polyominos

Un *polyomino* p est une union finie de carrés unitaires juxtaposés formant une pièce sans trou.

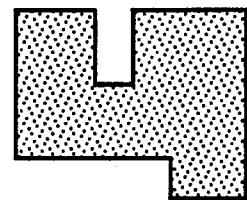
Exemple: Seule la pièce p_3 est un polyomino.



P_1



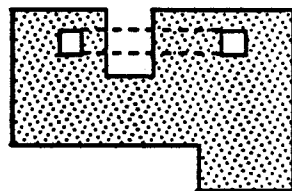
P_2



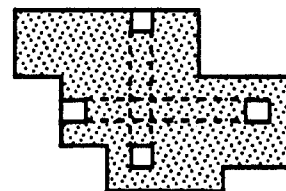
P_3

Un polyomino p est *convexe* si en choisissant deux carrés unitaires de p sur la même ligne horizontale ou verticale, tous les carrés situés entre eux sont à l'intérieur de p .

Exemple: Seul le polyomino p_4 est convexe.



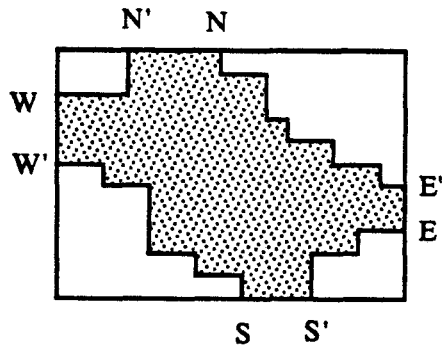
P_3



P_4

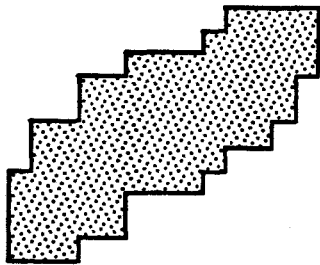
Quelques propriétés des polyominos convexes:

Soient p , un polyomino convexe et R , le plus petit rectangle contenant p . Le polyomino p est en contact avec le rectangle R le long de quatre segments. Nous nommons les extrémités de ces segments de la façon suivante: le point à l'extrémité gauche du segment horizontal le plus au sud est noté S , puis en tournant dans le sens des aiguilles d'une montre, nous rencontrons successivement W' , W , N' , N , E' , E et S' .

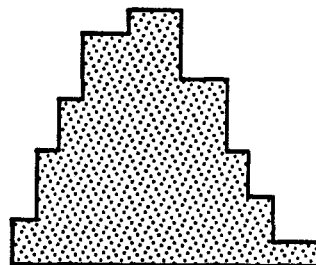


- Un *polyomino parallélogramme* est un polyomino convexe tel que: $(S=W'$ et $N=E')$ ou $(W=N'$ et $E=S')$.
- Un *polyomino pile* est un polyomino convexe tel que: $(E=S'$ et $S=W')$ ou $(S=W'$ et $W=N')$ ou $(W=N'$ et $N=E')$ ou $(N=E'$ et $E=S')$.

Exemple:

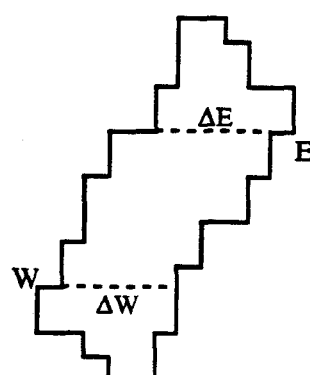
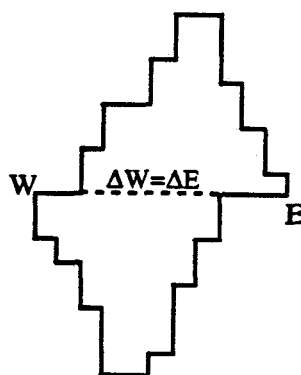
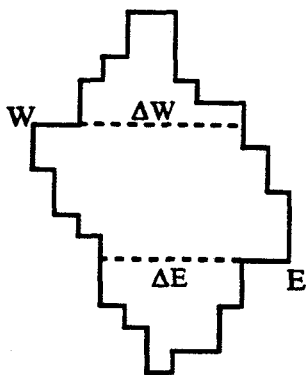


Un polyomino parallélogramme



un polyomino pile

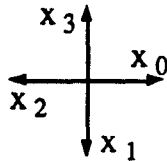
Soit p , un polyomino convexe. Notons ΔW et ΔE , les droites horizontales passant respectivement par les points W et E . D'après [DV], ces deux droites découpent le polyomino p en trois parties (éventuellement vides): celle située entre ΔW et ΔE est un polyomino parallélogramme et les deux autres sont des polyominos piles. En fait, trois cas se distinguent en fonction des positions relatives de ΔW et ΔE : dans chacun de ces cas, la partie de polyomino située au-dessus ou au-dessous de ΔW est soit un polyomino pile, soit l'union d'un polyomino parallélogramme et d'un polyomino pile.



4.1.2 Les mots de contour

Pour simplifier l'expression de certaines définitions, nous renommons les éléments de Π de la façon suivante:

$$\Pi = \{x_0, x_1, x_2, x_3\}$$



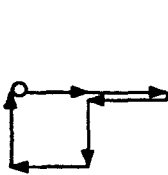
- Une *boucle* est un mot de Π^* dont le tracé est une courbe fermée. L'ensemble des boucles de Π^* est défini par:

$$\mathbb{B} = \{ w \in \Pi^* / e(w) = (0,0) \}.$$

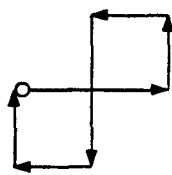
- Une boucle est *élémentaire* si le tracé associé ne se rencontre pas avant le retour au point de départ. L'ensemble des boucles élémentaires de Π^* est défini par:

$$\mathbb{BE} = \{ w \in \mathbb{B} / \forall v \in F(w) - \{\varepsilon, w\} \quad v \notin \mathbb{B} \}.$$

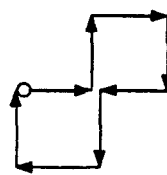
Exemple: Seul le mot w_4 est une boucle élémentaire.



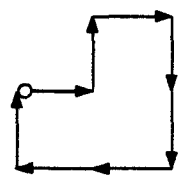
w_1



w_2



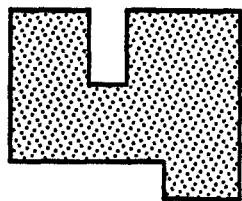
w_3



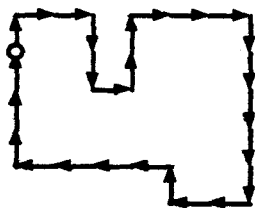
w_4

- Un polyomino p peut être caractérisé par un mot $w \in \Pi^*$ de la façon suivante: le tracé de w coïncide exactement avec la frontière du polyomino p . Ce mot est appelé *mot de contour* du polyomino p .

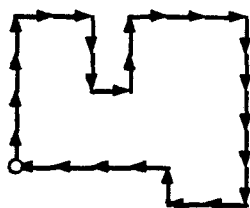
Exemple:



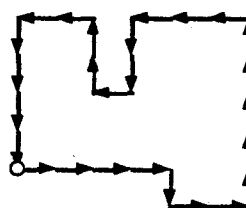
Un polyomino p .



w_1



w_2



w_3

Les mots w_1 , w_2 et w_3 sont des mots de contour du polyomino p .

Remarque: Il n'y a pas unicité du mot de contour caractérisant un polyomino p . Ce mot varie en fonction du point de départ de la description et du sens de parcours. Nous constatons que si w est un mot de contour du polyomino p , alors tous les mots de $[w] \cup [\bar{w}]$ sont également des mots de contour du polyomino p .

D'après [Bd], nous savons que:

BE est l'ensemble des mots de contour de polyominos.

Evidemment, il n'existe pas de bijection entre l'ensemble des mots de contour de polyominos et l'ensemble des polyominos.

- Un mot de contour de polyomino est dit *positif* si son tracé parcourt la frontière du polyomino dans le sens des aiguilles d'une montre (*négatif* sinon).

Pour définir les mots de contour positifs d'un polyomino convexe, nous introduisons l'ensemble \mathbb{P} suivant:

$$\mathbb{P} = [x_0 (x_0 + x_1)^* x_1 (x_1 + x_2)^* x_2 (x_2 + x_3)^* x_3 (x_3 + x_0)^*]$$

Dans la suite, les indices doivent être lus modulo 4: ils peuvent prendre les valeurs 0, 1, 2 ou 3.

Lemme 4.1.1:

- (1) $w \in \mathbb{P} \Leftrightarrow (w \in SM(\mathbb{P}) \text{ et } \forall i \in \{0,1,2,3\} x_i x_{i+1} \in F([w]), x_i x_{i+2} \notin F([w]))$
- (2) $\forall w \in \mathbb{P}, \forall i \in \{0,1,2,3\} \text{ card}(F(w) \cap x_{i-1} x_i^+ x_{i+1}) \leq 1$
- (3) $\forall i \in \{0,1,2,3\} x_{i-1} x_{i+1} x_{i-1} x_{i+1} \in SM(\mathbb{P})$

Preuve:

(1) \Rightarrow : D'après la forme des mots de \mathbb{P} , nous constatons que:

$$\forall i \in \{0,1,2,3\} x_i x_{i+1} \in F([w]) \text{ et } x_i x_{i+2} \notin F([w]).$$

(1) \Leftarrow : Caractérisons les sous-mots de \mathbb{P} :

$SM(\mathbb{P}) = [(x_0 + x_1)^* (x_1 + x_2)^* (x_2 + x_3)^* (x_3 + x_0)^*]$ et notons $M_i = (x_i + x_{i+1})^*$ pour tout $i \in \{0,1,2,3\}$. Soit $w \in SM(\mathbb{P})$, nous avons $w \in [M_0 M_1 M_2 M_3]$. Par conséquent, il existe $k \in \{0,1,2,3\}$ tel que:

$$w \in M_k M_{k+1} M_{k+2} M_{k+3} M_k.$$

Mettons en évidence le facteur $x_{k+2} x_{k+3}$, le plus à gauche dans le mot w :

$$w \in M_k M_{k+1} (x_{k+2} + x_{k+3})^* x_{k+2} x_{k+3} (x_{k+2} + x_{k+3})^* M_{k+3} M_k.$$

La lettre qui précède ce facteur dans le mot w , ne peut pas être x_{k+3} : D'après les hypothèses, la partie gauche de w (avant le premier facteur $x_{k+2} x_{k+3}$) doit contenir un facteur $x_{k+1} x_{k+2}$ et aucun facteur $x_{k+1} x_{k+3}$. Si cette partie gauche se termine par $(x_{k+3})^n$ ($n > 0$), la lettre juste avant ce facteur (différente de x_{k+3}) ne peut être ni x_{k+2} (car $x_{k+2} x_{k+3}$ ne doit pas apparaître avant dans le mot w), ni x_{k+1} (car $x_{k+1} x_{k+3} \notin F([w])$): nous aurions $w \in (x_k)^* (x_{k+3})^+ x_{k+2} x_{k+3} (x_{k+2} + x_{k+3})^* M_{k+3} M_k$ et $x_{k+1} x_{k+2} \notin F([w])$. Ceci est en contradiction avec les hypothèses. Par conséquent:

$$w \in M_k M_{k+1} (x_{k+2})^* x_{k+2} x_{k+3} (x_{k+2} + x_{k+3})^* M_{k+3} M_k$$

$$w \in M_k M_{k+1} x_{k+2} M_{k+2} M_{k+3} M_k.$$

En itérant ce processus, nous obtenons:

$$w \in M_k x_{k+1} M_{k+1} x_{k+2} M_{k+2} x_{k+3} M_{k+3} x_k M_k \subset \mathbb{P}.$$

4.2 Quel est le problème ?

D'après [Bd], nous savons que: Si $R \subset \Pi^*$ est un langage reconnaissable, le problème "Le langage R contient-il une boucle ?" est décidable et le problème "Le langage R contient-il un mot de contour de polyomino ?" est indécidable.

Nous limitons le problème aux mots de contour de polyominos convexes. Plus précisément, nous traitons uniquement le cas des mots de contour positifs de polyominos convexes (le cas négatif est strictement identique). D'où: Le problème "Le langage R contient-il un mot de contour positif de polyomino convexe ?" est-il décidable ?

Puisque CP est fermé par conjugaison, nous avons $[R] \cap CP = [R \cap CP]$. Par conséquent, les trois problèmes suivants sont équivalents:

$$(R \cap CP \neq \emptyset) \Leftrightarrow ([R \cap CP] \neq \emptyset) \Leftrightarrow ([R] \cap CP \neq \emptyset).$$

Posons $S = [R] \cap P$. Comme $CP \subset P$, nous obtenons:

$$([R] \cap CP) = ([R] \cap P \cap CP) = (S \cap CP).$$

Les langages reconnaissables étant fermés par conjugaison et intersection, notre question peut également s'exprimer ainsi: "Le langage reconnaissable S contient-il un mot de contour positif de polyomino convexe ?" Cette formulation est plus agréable car S est un ensemble fermé par conjugaison et inclus dans P .

La section suivante est consacrée à la preuve du théorème suivant:

Théorème 4.2.1:

Le problème de savoir si un langage reconnaissable de Π^ contient un mot de contour (positif) d'un polyomino convexe est décidable.*

Remarque: J. Dassow et F. Hinz [Da1, Da2, DH1] semblent avoir des résultats très proches des nôtres. Il n'en est rien: leurs propos concernent les figures associées au mot du langage en tant qu'ensembles de segments. Par exemple, le mot "urdlu" caractérise un carré unitaire en passant deux fois sur un même segment. Pour eux, il s'agit d'un mot représentant une courbe simple fermée (de chaque noeud partent exactement deux segments) tandis que pour nous, ce mot n'est même pas une boucle: il ne s'agit pas d'un mot de contour de polyomino.



4.3 La preuve

4.3.1 Idée de la preuve

D'abord, nous introduisons un système de réécriture R sur Π et nous prouvons que si $R^*(w)$ contient un élément de CP alors nécessairement $w \in CP$.

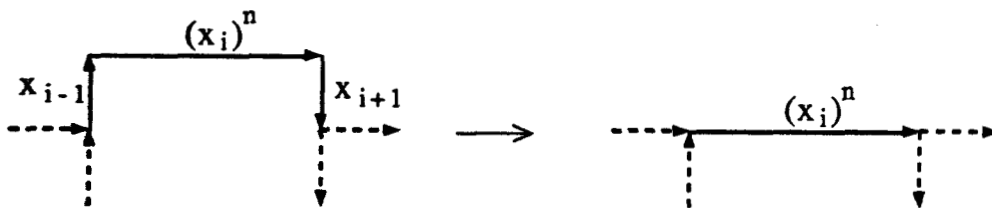
Puis, nous définissons un système de réécriture T sur Π tel que $T(S) \subset R^*(S)$ et nous montrons que si S contient un élément de CP alors $T^{\leq 2}(S)$ contient un mot de contour positif d'un carré unitaire. Comme $T^{\leq 2}(S)$ est un langage reconnaissable que nous savons construire, nous pouvons décider si $T^{\leq 2}(S)$ contient un mot particulier.

De ces résultats:

Nous pouvons décider si un langage reconnaissable de Π^* contient un mot de contour positif de polyomino convexe.

4.3.2 Le système de réécriture R

$$R = \bigcup_{i=0}^{i=3} \{ R_i \}$$



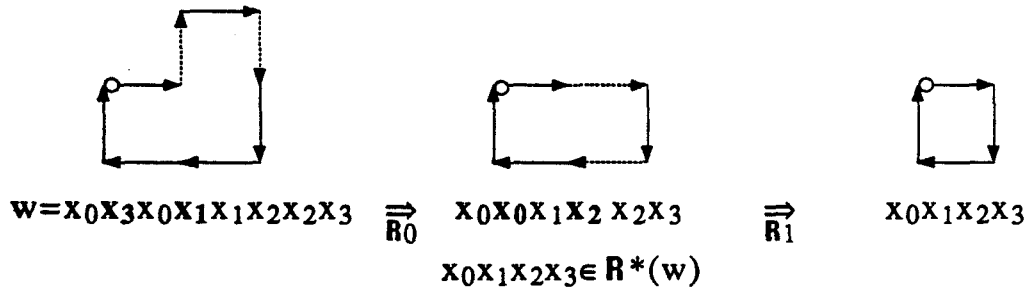
$$R_i = \{ \quad z x_{i-1} (x_i)^n x_{i+1} z' \quad \rightarrow \quad z (x_i)^n z' \quad \text{avec } z \in \{ x_{i-1}, x_i \}, z' \in \{ x_i, x_{i+1} \} \quad \text{et} \quad n \geq 1 \}$$

Quelques notations: $\forall w \in \Pi^*$,

$\forall r \in R$, $r(w)$ est l'ensemble des mots dérivés de w en appliquant la règle r .

$$R_i(w) = \bigcup_{r \in R_i} r(w) \quad R(w) = \bigcup_{i=0}^{i=3} R_i(w) \quad R^*(w) = \bigcup_{p \geq 0} R^p(w)$$

Exemple:



Remarque 4.3.1:

$\forall w \in \Pi^*, \forall i \in \{0,1,2,3\}, \forall r \in R_i, \forall w' \in r(w)$, nous avons:

- $|w'| = |w| - 2;$
- $|w'|_{x_i} = |w|_{x_i} \quad \text{et} \quad |w'|_{x_{i+2}} = |w|_{x_{i+2}};$
- $|w'|_{x_{i-1}} - |w'|_{x_{i+1}} = |w|_{x_{i-1}} - |w|_{x_{i+1}};$

Nous en déduisons: $w \in B \Leftrightarrow w' \in B$

Remarque 4.3.2:

$$\forall w \in P, \forall r \in R, \quad \text{card}(r(w)) \leq 1$$

Ceci est une conséquence directe du lemme 4.1.1 (2).

Notation:

- Dans la suite, comme nous considérons w dans P , $r(w)$ représentera l'unique élément de l'ensemble $r(w)$ (s'il existe).

Lemme 4.3.3:

P est fermé par R

Preuve:

Nous montrons que: $\forall w \in P, R(w) \subset P$.

Si l'une des règles r de R_j ($j \in \{0,1,2,3\}$) peut être appliquée à w , nous vérifions que $r(w) \in P$: De façon évidente, $r(w) \in SM(P)$ et d'après le lemme 4.1.1 (1), il suffit de montrer que: $\forall i \in \{0,1,2,3\} \quad x_i x_{i+1} \in F([r(w)])$ et $x_i x_{i+2} \in F([r(w)])$.

La règle r peut être appliquée à w si celui-ci contient un facteur de la forme $zx_{j-1}(x_j)^n x_{j+1}z'$ avec $n \geq 1$, $z \in \{x_{j-1}, x_j\}$, $z' \in \{x_j, x_{j+1}\}$ qui se transformera en $z(x_j)^n z'$ dans $r(w)$. La disparition des deux lettres (en gras) n'engendre pas de facteurs de la forme $x_i x_{i+2}$ dans $r(w)$, $i \in \{0, 1, 2, 3\}$ (ceci est dû au contexte).

Il ne nous reste qu'à vérifier que $x_{j-1}x_j$ et $x_j x_{j+1}$ existent encore dans $F([r(w)])$: Si $z=x_{j-1}$, alors évidemment $x_{j-1}x_j \in F([r(w)])$; Si $z=x_j$, d'après la forme des mots de \mathbb{P} , il existe nécessairement un facteur $x_{j-1}x_j^*$ devant z dans l'un des mots de $[w]$. Par conséquent, $x_{j-1}x_j \in F([r(w)])$. Pour des raisons similaires, nous avons $x_j x_{j+1} \in F([r(w)])$. ♦

Lemme 4.3.4:

Soient $w \in P$ et $w' \in R(w)$, ($w \in CP \Leftrightarrow w' \in CP$)

Preuve:

En se référant au lemme 4.3.3, il suffit de montrer que:

$$\forall w \in P, \forall w' \in R(w), (w \in BE \Leftrightarrow w' \in BE).$$

D'après la remarque 4.3.1, nous savons que ($w \in B \Leftrightarrow w' \in B$). Il ne nous reste qu'à montrer la propriété suivante:

$$\begin{aligned} & \forall w \in P \cap B, \forall w' \in R(w) \\ & \text{(Aucun facteur propre de } w \text{ n'est une boucle)} \\ & \Leftrightarrow \\ & \text{(Aucun facteur propre de } w' \text{ n'est une boucle).} \end{aligned}$$

Il s'agit de vérifier que R ne permet pas de créer ou de détruire des boucles dans w .

\Leftarrow : Considérons un mot $w \in P \cap B$ dont un facteur propre b est une boucle. Nous montrons que: $\forall i \in \{0, 1, 2, 3\}$, $\forall w' \in R_i(w)$, le mot w' contient également une boucle comme facteur propre.

Posons $w = w_1 b w_2 = u_1 \alpha u_2$ avec $\alpha = x_{i-1}(x_i)^n x_{i+1}$, $n \geq 1$ et $w_1, w_2, u_1, u_2 \in \Pi^*$. D'après le lemme 4.1.1 (2), une règle de R_i efface nécessairement les deux lettres (en gras) de α .

Cas 1: α est un facteur de b :

Posons $b = b_1 \alpha b_2$ avec $b_1, b_2 \in \Pi^*$. Nous avons $b_1(x_i)^n b_2$, facteur propre de w' et élément de B .

Cas 2: α et b sont des facteurs disjoints de w :

Le facteur b est encore un facteur propre de w' .

Cas 3: . α et b sont des facteurs qui se chevauchent dans w :
 Nous montrons que ce cas ne peut pas exister.

Supposons que α se situe sur w_1 et b (l'autre chevauchement se traite de façon identique). Puisque $w \in \mathbb{P} \cap \mathbb{B}$, nous avons $w_1 w_2 \in \mathbb{B}$ et une occurrence de chaque lettre de Π apparaît dans $w_1 w_2$ et dans b .

Posons $w_1 = w'_1 x_{i-1} (x_i)^{n'}$, $b = (x_i)^{n''} x_{i+1} v$ avec $w'_1, v \in \Pi^*$ et $n' + n'' = n$.
 Soit $w'' = w_2 w_1 b \in [w]$, nous avons $w'' = w_2 w'_1 x_{i-1} (x_i)^{n' + n''} x_{i+1} v$ et nous savons que v contient une occurrence de x_{i-1} , et $w_2 w'_1$ une occurrence de x_{i+1} . Par conséquent, $x_{i+1} x_{i-1} x_{i+1} x_{i-1}$ est un sous-mot de w'' : Ceci est impossible car $w'' \in \mathbb{P}$ [lemme 4.1.1 (3)].

\Rightarrow : Soit $w \in \mathbb{P} \cap \mathbb{B}$ et supposons que $w' \in R(w)$ contienne un facteur propre $b \in \mathbb{B}$. Nous montrons que nécessairement le mot w contient une boucle comme facteur propre.

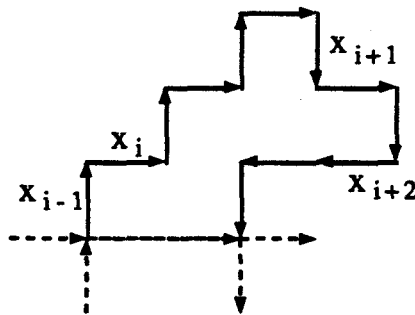
Posons $w' = w'_1 b w'_2$ et de la même façon, nous prouvons facilement que:

. Si les lettres effacées étaient toutes les deux dans b ou à l'extérieur de b , le mot w contenait également une boucle comme facteur propre.

. Si l'une des deux lettres était dans b et l'autre dans w'_1 (resp. w'_2), le mot w ne pouvait pas être un mot de \mathbb{P} et nous obtenons une contradiction. ♦

4.3.3 Le système de réécriture T

$$T = \bigcup_{i=0}^{i=3} \{T_i\}$$



$$T_i = \{ z w z' \rightarrow z (x_i)^j z' \text{ avec } w \in (x_{i-1}+x_i)^* x_i (x_i+x_{i+1})^* x_{i+1} (x_{i+1}+x_{i+2})^* w(x_{i+2})^j \in CP, j \geq 1, z \in \{x_{i-1}, x_i\}, z' \in \{x_i, x_{i+1}\} \}$$

Remarque 4.3.5:

$$(w \in (x_{i-1}+x_i)^* x_i (x_i+x_{i+1})^* x_{i+1} (x_{i+1}+x_{i+2})^* \text{ et } w(x_{i+2})^j \in CP, j \geq 1) \Rightarrow (w \in x_{i-1} \Pi^*).$$

Quelques notations: $\forall w \in \Pi^*$,

$\forall t \in T$, $t(w)$ est l'ensemble des mots dérivés de w en appliquant la règle t .

$$T_i(w) = \bigcup_{t \in T_i} t(w) \quad T(w) = \bigcup_{i=0}^{i=3} T_i(w) \quad T^{\leq n}(w) = \bigcup_{0 \leq i \leq n} T^i(w)$$

Quelques définitions:

Soit $M=(\Pi, Q, \delta, q_0, F)$, un automate fini. Le langage reconnu par M est noté $L(M)$ et $Rec(\Pi^*)$ est l'ensemble des langages inclus dans Π^* et reconnus par un automate fini. Ces langages sont les langages définis par des expressions rationnelles sur Π (pour plus de détails concernant les automates finis et leurs langages, le lecteur se reportera à [Au], [HU] ou [Pi]).

Lemme 4.3.6:

Pour tout $R \in Rec(\Pi^*)$, $T(R)$ est un langage reconnaissable constructible.

Donnons quelques résultats intermédiaires pour démontrer ce lemme.

Pour tout $i \in \{0,1,2,3\}$, nous introduisons une transformation τ_i conservant la reconnaissabilité des langages et nous montrons que $T_i(R) = \tau_i(R)$ pour tout langage $R \in \text{Rec}(\Pi^*)$.

Soit $M = (\Pi, Q, \delta, q_0, F)$, un automate fini tel que $L(M) = R$. Nous définissons:

$$\tau_i(R) = \bigcup_{\substack{q, q', q'' \in Q \\ q_f \in F}} r_1(q_0, q) \quad r_2(q, q', q'') \quad r_3(q', q_f)$$

avec

$$r_1(q_0, q) = R_{q_0 q} \cap \Pi^* (x_{i-1} + x_i)$$

$$r_2(q, q', q'') = f_i (\pi_{\{x_i, x_{i+2}\}} ((r'_2(q, q'') \sqcup r''_2(q'', q')) \cap (x_{i+2}^* x_i^* (x_{i-1} x_{i+1}))^+ \cap L_i))$$

$$\text{avec } r'_2(q, q'') = (R_{q q''} \cap (x_{i-1} + x_i)^* x_i)^f \text{ et}$$

$$r''_2(q'', q') = R_{q'' q'} \cap (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^* \cap x_{i+1} \Pi^*$$

$$r_3(q', q_f) = R_{q' q_f} \cap (x_i + x_{i+1}) \Pi^*$$

et $R_{q_1 q_2}$, le langage reconnu par $M' = (\Pi, Q, \delta, q_1, \{q_2\})$;

π_B , la projection sur l'ensemble B ;

f_i , la réduction de Dyck sur $(x_i + x_{i+2})^*$ définie par:

$$\forall w \in (x_i + x_{i+2})^* \text{ avec } |w|_{x_i} - |w|_{x_{i+2}} = j, j \geq 0, f_i(w) = (x_i)^j;$$

$L_i = \{x_i w \in \Pi^* / \forall f \in FG(w) \mid |f|_{x_i} \geq |f|_{x_{i+2}}\}$, un langage algébrique.

Notons que les opérations de concaténation, \sqcup et \cap conservent la reconnaissabilité et les opérateurs π_B et f_i , l'algébricité des langages. Nous en déduisons:

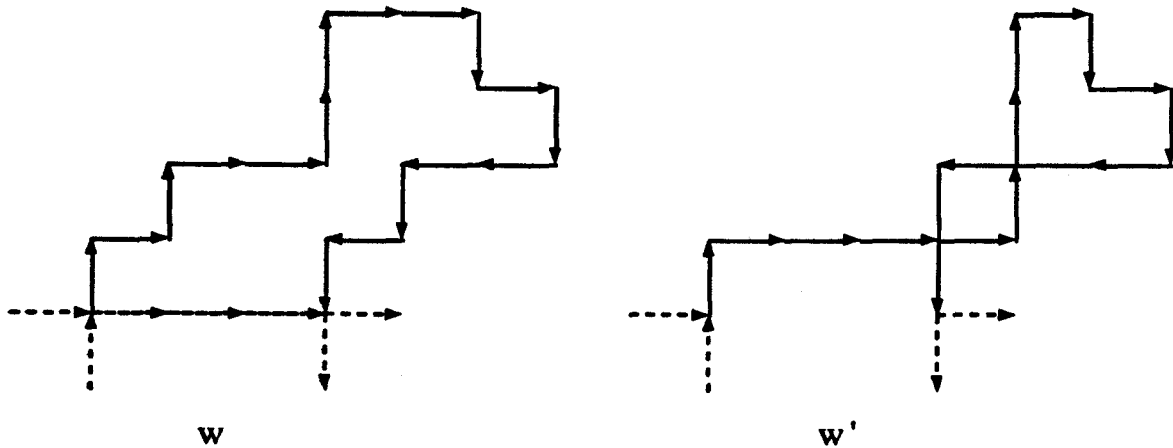
- . $r_1(q_0, q)$ et $r_3(q', q_f)$ sont des langages reconnaissables;
- . $r_2(q, q', q'')$ est un langage algébrique défini sur un alphabet à une lettre: un tel langage est reconnaissable [HU].

Ainsi, $\tau_i(R)$ est l'union finie d'ensembles reconnaissables:

$\tau_i(R)$ est un ensemble reconnaissable.

Illustrons d'abord la transformation τ_i sur un exemple pour se donner une idée intuitive "de ce qui se passe".

Cette transformation doit permettre de "couper une branche" tout en conservant l'appartenance du mot de contour à CP.



La réduction en $(x_0)^3$ ne doit être permise que dans le premier cas: il ne faut pas que nous puissions éliminer une boucle. Par conséquent, nous n'autorisons la transformation qu'après avoir vérifié qu'à chaque niveau (en partant du haut), il existe strictement plus d'occurrences de x_0 que de x_2 .

Pour cela, nous coupons w en deux parties: "la montée" w_1 et la "descente" w_2 . Nous renversons le facteur "montée" et nous comparons les occurrences de x_0 et de x_2 en ajustant le "niveau" à l'aide de x_1 et de x_3 . Nous obtenons:

$$\begin{array}{ll} w_1 = & x_3 x_0 x_3 x_0 x_0 x_3 x_3 x_0 x_0 \\ w_2 = & x_1 x_0 x_1 x_2 x_2 x_1 x_2 x_1 \\ w_1^r = & x_0 x_0 x_3 x_3 x_0 x_0 x_3 x_0 x_3 \\ w_2 = & x_1 x_0 x_1 x_2 x_2 x_1 x_2 x_1 \\ w_1' = & x_3 x_0 x_0 x_0 x_0 x_3 x_3 x_3 x_0 \\ w_2' = & x_1 x_0 x_1 x_2 x_2 x_2 x_1 x_1 \\ w_1'^r = & x_0 x_3 x_3 x_3 x_0 x_0 x_0 x_0 x_3 \\ w_2' = & x_1 x_0 x_1 x_2 x_2 x_2 x_1 x_1 \end{array}$$

puis, en ajustant les x_1 et x_3 :

$$\begin{array}{ll} w_1'^r = & x_0 x_0 x_3 x_3 x_0 x_0 x_3 x_0 x_3 \\ w_2 = & x_1 x_0 x_1 x_2 x_2 x_1 x_2 x_1 \\ w_1'^r = & x_0 x_3 x_3 x_3 x_0 x_0 x_0 x_0 x_3 \\ w_2' = & x_1 x_0 x_1 x_2 x_2 x_2 x_1 x_1 \end{array}$$

Nous mêlons $w_1'^r$ et w_2 en regroupant chaque occurrence de x_1 avec une de x_3 et en plaçant les occurrences de x_2 avant celles de x_0 . Ainsi, tous les facteurs gauches du mot obtenu contiennent strictement plus d'occurrences de x_0 que de x_2 si et seulement si la réduction peut être appliquée. Cette vérification est faite par un shuffle entre $w_1'^r$ et w_2 , suivi d'une intersection avec $(x_2^* x_0^* (x_3 x_1))^+$ et nous ne gardons que les mots de L_0 .

Nous obtenons:

$$\begin{array}{l} s = (w_1'^r \sqcup w_2) \cap (x_2^* x_0^* (x_3 x_1))^+ = x_0 x_0 x_3 x_1 x_0 x_3 x_1 x_2 x_2 x_0 x_0 x_3 x_1 x_2 x_0 x_3 x_1 \text{ et} \\ s' = (w_1'^r \sqcup w_2') \cap (x_2^* x_0^* (x_3 x_1))^+ = x_0 x_3 x_1 x_0 x_3 x_1 x_2 x_2 x_2 x_3 x_1 x_0 x_0 x_0 x_0 x_3 x_1 \\ \text{et nous avons bien } s \in L_0 \text{ et } s' \notin L_0 \text{ (à cause du facteur en gras).} \end{array}$$

Fait 4.3.7:

Soit $w = w_1 w_2$ avec $w_1 \in (x_{i-1} + x_i)^* x_i$ et $w_2 \in (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^* \cap x_{i+1} \Pi^*$
 $w(x_{i+2})^j \in \mathbb{C}\mathbb{P}$, $j \geq 1$,

\Leftrightarrow

$$f_i(\Pi_{\{x_i, x_{i+2}\}}((w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^*(x_{i-1}x_{i+1}))^+ \cap L_i)) = (x_i)^j.$$

Preuve:

Note: Si $(w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^*(x_{i-1}x_{i+1}))^+ \cap L_i \neq \emptyset$, cet ensemble contient un unique élément noté $c(w)$.

\Rightarrow : Soit $n = |w|_{x_{i-1}}$.

Puisque $w(x_{i+2})^j \in \mathbb{C}\mathbb{P}$, nous avons $|w|_{x_{i-1}} = |w|_{x_{i+1}} = n$ et d'après la forme des facteurs w_1 et w_2 , nous en déduisons que $|w_1|_{x_{i-1}} = |w_2|_{x_{i+1}} = n$.

Posons: $\forall k \in [0, n-1]$ $w_1 = u_{1,k} x_{i-1} v_{1,k}$ avec $|v_{1,k}|_{x_{i-1}} = k$,
 $w_2 = u_{2,k} x_{i+1} v_{2,k}$ avec $|u_{2,k}|_{x_{i+1}} = k$,

et montrons par récurrence sur k que:

$$(*) \quad \forall k \in [0, n-1] \quad |v_{1,k} u_{2,k}|_{x_i} > |v_{1,k} u_{2,k}|_{x_{i+2}}.$$

. Si $k=0$, nous avons $v_{1,k} \in x_i^+$ et $u_{2,k} = \varepsilon$: la propriété (*) est vraie.

. Supposons que $k \in [0, n-2]$ et $|v_{1,k} u_{2,k}|_{x_i} > |v_{1,k} u_{2,k}|_{x_{i+2}}$. Nous savons que $v_{1,k+1} u_{2,k+1} \in x_i^* x_{i-1} v_{1,k} u_{2,k} x_{i+1} (x_i^* + x_{i+2}^*)$ et nous vérifions facilement que si $|v_{1,k+1} u_{2,k+1}|_{x_i} \leq |v_{1,k+1} u_{2,k+1}|_{x_{i+2}}$, il existe nécessairement un facteur de $v_{1,k+1} u_{2,k+1}$ appartenant à \mathbb{B} . Comme $w(x_{i+2})^j \in \mathbb{C}\mathbb{P}$, ceci est impossible. La propriété (*) est démontrée.

D'après cette propriété, il est évident que:

$$(w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^*(x_{i-1}x_{i+1}))^+ \cap L_i \neq \emptyset$$

et comme le mot $c(w)$ contient exactement les mêmes lettres que w , nous avons:

$$f_i(\Pi_{\{x_i, x_{i+2}\}}(c(w))) = f_i(\Pi_{\{x_i, x_{i+2}\}}(w)) = (x_i)^j \text{ (car } w(x_{i+2})^j \in \mathbb{C}\mathbb{P}).$$

\Leftarrow :

Supposons que w contienne une boucle comme facteur: Il existe une décomposition $w_1 = w'_1 w''_1$ et $w_2 = w'_2 w''_2$ tel que $w''_1 w'_2 \in \mathbb{B}$. Posons $w''_1 = (x_i)^\alpha x_{i-1} v_1$ et $w'_2 = v_2 x_{i+1} (x_{i+2})^\beta$ avec $\alpha, \beta \geq 0$. Nous avons $w_1^r = v_1^r x_{i-1} (x_i)^\alpha w'^r_1$, $w_2 = v_2 x_{i+1} (x_{i+2})^\beta w''_2$ et $|v_1|_{x_{i-1}} = |v_2|_{x_{i+1}}$. Nous

vérifions facilement que $\forall m \in (w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^* (x_{i-1} x_{i+1}))^+$, $\exists f \in FG(m)$ tel que $|f|_{x_{i+2}} \geq |f|_{x_i}$. Par conséquent, $m \notin L_i$ et nous obtenons $(w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^* (x_{i-1} x_{i+1}))^+ \cap L_i = \emptyset$. Contradiction.

De plus, il est évident que:

$$(w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^* (x_{i-1} x_{i+1}))^+ \cap L_i \neq \emptyset \Rightarrow |w|_{x_{i-1}} = |w|_{x_{i+1}}.$$

Les conditions $f_i(\Pi_{\{x_i, x_{i+2}\}}(c(w))) = (x_i)^j$, $|w|_{x_{i-1}} = |w|_{x_{i+1}}$ et $F(w) \cap \mathbb{B} = \emptyset$ nous permettent de conclure que $w(x_{i+2})^j \in \mathbb{C}\mathbb{P}$. fin de la preuve du fait. ♦

Preuve du lemme 4.3.6:

$T_i(R) \subset \tau_i(R)$?

Soit $m \in R$.

Pour tout $m' \in T_i(m)$, nous pouvons décomposer m de la façon suivante: $m = \alpha w z' \beta$ avec $w \in (x_{i-1} + x_i)^* x_i (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^*$, $w(x_{i+2})^j \in \mathbb{C}\mathbb{P}$, $j \geq 1$, $z \in \{x_{i-1}, x_i\}$, $z' \in \{x_i, x_{i+1}\}$ et $\alpha, \beta \in \Pi^*$ tel que $m' = \alpha z (x_i)^j z' \beta$. Puisque $m \in R$, il existe $(q, q') \in Q^2$ et $q_f \in F$ tel que $\alpha z \in R_{q_0 q}$, $w \in R_{q q'}$ et $z' \beta \in R_{q' q_f}$. De façon évidente, nous avons $\alpha z \in r_1(q_0, q)$ et $z' \beta \in r_3(q', q_f)$. D'après la forme du facteur w , il existe une unique décomposition $w = w_1 w_2$ vérifiant $w_1 \in (x_{i-1} + x_i)^* x_i$ et $w_2 \in (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^* \cap x_{i+1} \Pi^*$. Soit q'' , l'état de Q tel que $w_1 \in R_{q q''}$ et $w_2 \in R_{q'' q'}$: nous avons $w_1^r \in r'_2(q, q'')$ et $w_2 \in r''_2(q'', q')$. Le fait 4.3.7 nous permet d'affirmer que $(x_i)^j \in r_2(q, q', q'')$. Par conséquent, $m' \in \tau_i(R)$.

$\tau_i(R) \subset T_i(R)$?

Soit $m' \in \tau_i(R)$, nous pouvons décomposer m' de la façon suivante: $m' = m_1 (x_i)^j m_2$ avec $m_1 \in r_1(q_0, q)$, $m_2 \in r_3(q', q_f)$, $(x_i)^j \in r_2(q, q', q'')$ et $(q, q', q'') \in Q^3$. Puisque $(x_i)^j \in r_2(q, q', q'')$, il existe $w_1, w_2 \in \Pi^*$ tel que $w_1^r \in r'_2(q, q'')$, $w_2 \in r''_2(q'', q')$ et $f_i(\Pi_{\{x_i, x_{i+2}\}}((w_1^r \sqcup w_2) \cap (x_{i+2}^* x_i^* (x_{i-1} x_{i+1}))^+ \cap L_i)) = (x_i)^j$. D'après le fait 4.3.7, nous savons que $w_1 w_2 (x_{i+2})^j \in \mathbb{C}\mathbb{P}$ et nous avons $m_1 w_1 w_2 m_2 \in R$ avec $m_1 \in \Pi^*(x_{i-1} + x_i)$, $m_2 \in (x_i + x_{i+1}) \Pi^*$ et $w_1 w_2 (x_{i+2})^j \in \mathbb{C}\mathbb{P}$. Toutes ces conditions nous permettent d'affirmer que $m' \in T_i(R)$.

Par conséquent:

Pour tout $i \in \{0, 1, 2, 3\}$ et pour tout $R \in \text{Rec}(\Pi^*)$: $T_i(R) = \tau_i(R)$.

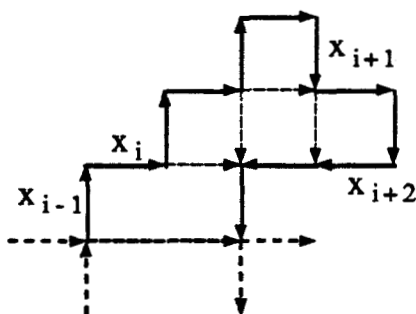
Comme $\tau_i(R)$ est un ensemble reconnaissable que nous savons construire, le lemme est démontré. ♦

Lemme 4.3.8:

Pour tout mot $m \in P$, $T(m) \subset R^*(m)$

Preuve:

Pour tout mot $m \in P$ et pour tout mot $m' \in T_i(m)$, $i \in \{0, 1, 2, 3\}$, nous pouvons décomposer m de la façon suivante: $m = \alpha z w z' \beta$ avec $z \in \{x_{i-1}, x_i\}$, $z' \in \{x_i, x_{i+1}\}$, $w \in (x_{i-1} + x_i)^* x_i (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^*$, $w(x_{i+2})^j \in CP$, $j \geq 1$, et $\alpha, \beta \in \Pi^*$ tel que $m' = \alpha z(x_i)^j z' \beta$.



Nous montrons par récurrence sur $|w|$ que $m' \in R^*(m)$.

Puisque $w(x_{i+2})^j \in CP$, il existe nécessairement une occurrence de x_{i+1} et de x_{i-1} dans w . Posons $w = u x_{i-1} (x_i)^n x_{i+1} v$ avec $n \geq 1$, $u, v \in \Pi^*$.

Cas 1: $v = \epsilon$.

Puisque $w(x_{i+2})^j \in CP$ et $v = \epsilon$, le facteur u ne peut pas contenir d'occurrence de x_{i-1} : nécessairement, $u \in x_i^*$.

Or d'après la remarque 4.3.5, nous savons que la première lettre de w est x_{i-1} : nécessairement, $u = \epsilon$.

Par conséquent, $n = j$ et $m' \in R_j(m)$.

Cas 2: $v \in (x_i + x_{i+1}) \Pi^*$.

D'après la forme du facteur w , nous avons $|v|_{x_{i+1}} \geq 1$ et le facteur u contient au moins une occurrence de x_{i-1} . Une règle de R_i peut être appliquée à w : nous obtenons $w' = u(x_i)^n v \in R_i(w)$. D'après le lemme 4.3.4 et puisque $w'(x_{i+2})^j \in R_i(w x_{i+2}^j)$, nous avons $w'(x_{i+2})^j \in CP$.

Par conséquent, nous obtenons $\alpha z w' z' \beta \in R_i(m)$ avec $|w'| = |w| - 2$, $w' \in (x_{i-1} + x_i)^* x_i (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^*$ et $w'(x_{i+2})^j \in CP$: la récurrence se poursuit en remplaçant w par w' .

Cas 3: $\cdot v \in x_{i+2} \Pi^+$.

De la même façon, le facteur v nous assure le contexte pour appliquer une règle de R_{i+1} et nous obtenons $\alpha z w' z' \beta \in R_{i+1}(m)$ avec $w' \in (x_{i-1} + x_i)^* x_i (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^*$, $|w'| = |w| - 2$ et $w'(x_{i+2})^j \in CP$: la récurrence se poursuit en remplaçant w par w' .

Cas 4: $\cdot v = x_{i+2}$.

Dans ce cas particulier, nous avons $w = x_{i-1} (x_i)^n x_{i+1} x_{i+2}$ avec $n = j + 1 \geq 2$. Puisque $m \in \mathbb{P}$, nous avons $z' = x_{i+1}$ et une règle de R_{i+1} peut être appliquée à m . Nous obtenons $\alpha z w' z' \beta \in R_{i+1}(m)$ avec $w' = x_{i-1} (x_i)^{n-1} x_{i+1}$. Nous vérifions facilement que $|w'| = |w| - 2$, $w' \in (x_{i-1} + x_i)^* x_i (x_i + x_{i+1})^* x_{i+1} (x_{i+1} + x_{i+2})^*$ et $w'(x_{i+2})^j \in CP$: la récurrence se poursuit en remplaçant w par w' . ♦

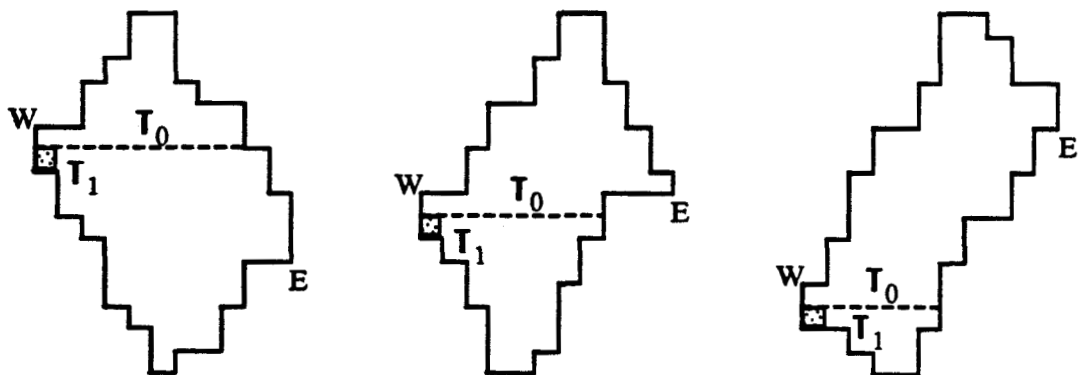
Proposition 4.3.9:

Pour tout mot $w \in P$, ($w \in CP \Leftrightarrow \exists i, x_i x_{i+1} x_{i+2} x_{i+3} \in T^{\leq 2}([w])$).

Preuve:

Les affirmations suivantes découlent des propriétés de trisection des polyominos convexes introduites précédemment (§ 4.1.1).

\Rightarrow : Soit w , un mot de CP et notons p , le polyomino convexe caractérisé par w . Supposons que p ne soit pas un carré unitaire. Nous appliquons à un certain mot de $[w]$ une règle de T_0 de façon à ce que le polyomino p' issu de cette transformation vérifie (W et W' distants d'une unité) et ($W=N'$). Si p' n'est pas un carré unitaire, il suffit d'appliquer une règle de T_1 pour obtenir le résultat désiré.



\Leftarrow : La réciproque est une conséquence directe des lemmes 4.3.8 et 4.3.4. ♦

Preuve du théorème 4.2.1:

Puisque $T^{\leq 2}(S)$ est un langage reconnaissable que nous savons construire (lemme 4.3.6), nous pouvons décider si $T^{\leq 2}(S)$ contient certains mots. D'après la proposition 4.3.9, nous concluons que "vérifier si un langage reconnaissable S contient un mot de contour positif d'un polyomino convexe" est un problème décidable.♦

4.4 Extension aux polyominos semi-convexes

Pouvons-nous généraliser ce résultat à d'autres polyominos particuliers?

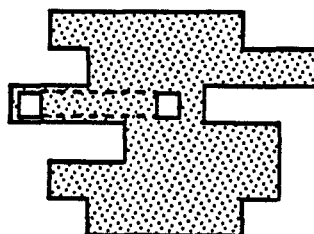
Dans cette section, nous étendons le théorème précédent aux polyominos convexes dans une seule direction.

4.4.1 Quelques définitions

Nous traitons le cas des polyominos horizontalement convexes: l'extension aux polyominos verticalement convexes est évidente.

- Un polyomino p est *horizontalement convexe* si en choisissant deux carrés unitaires sur la même ligne horizontale, tous les carrés situés entre eux sont à l'intérieur de p .

Exemple: Ce polyomino est horizontalement convexe.



Remarque: Un polyomino horizontalement convexe peut être vu comme un empilement non axé de rectangles d'épaisseur égale à 1.

Pour définir les mots de contour positifs d'un polyomino horizontalement convexe, nous introduisons l'ensemble \mathbb{H} suivant:

$$\mathbb{H} = [x_0^+ x_1 (x_0^* x_1 + x_2^* x_1)^* x_2^+ x_3 (x_0^* x_3 + x_2^* x_3)^*]$$

Beaucoup de similitudes existent entre les énoncés suivants et ceux de la section précédente: nous n'explicitons les preuves que dans le cas où celles-ci diffèrent.

Lemme 4.4.1:

- (1) $w \in H \Leftrightarrow (w \in SM(H) \text{ et } \forall i \in \{0,2\} x_{i-1}x_i^+x_{i+1} \cap F([w]) \neq \emptyset, x_i x_{i+2} \notin F([w]))$
- (2) $\forall w \in H, \forall i \in \{0,2\} \text{ card}(F(w) \cap x_{i-1} x_i^+ x_{i+1}) \leq 1$
- (3) $\forall i \in \{0,2\} x_{i-1} x_{i+1} x_{i-1} x_{i+1} \notin SM(H)$
- (4) $w \in H \text{ et } b \in (F([w]) - \{\varepsilon\}) \cap BE \Rightarrow b \in H$

Preuve:

(1) \Rightarrow : D'après la forme des mots de H , nous constatons que:

$$\forall i \in \{0,2\} x_{i-1}x_i^+x_{i+1} \cap F([w]) \neq \emptyset, x_i x_{i+2} \notin F([w]).$$

(1) \Leftarrow : Caractérisons les sous-mots de H :

$SM(H) = [(x_0 + x_1 + x_2)^* (x_2 + x_3 + x_0)^*]$ et notons $N_i = (x_i + x_{i+1} + x_{i+2})^*$ pour tout $i \in \{0,2\}$.

Remarque : $\forall i \in \{0,2\}$,

$$((m \in N_i) \text{ et } (F(m) \cap \{x_0x_2, x_2x_0\} = \emptyset)) \Rightarrow (m \in (x_0^*x_{i+1}+x_2^*x_{i+1})^*(x_0^*+x_2^*)).$$

Soit $w \in SM(H)$, nous avons: $w \in [N_0 N_2]$.

Sachant que $F([w])$ contient des facteurs de la forme $x_1x_2^+x_3$ et $x_3x_0^+x_1$, nous avons: $w \in [N_0 x_1x_2^+x_3 N_2] \cap [N_2 x_3x_0^+x_1 N_0]$.

D'après la remarque précédente et comme $F([w]) \cap \{ x_0x_2, x_2x_0 \} = \emptyset$:

$$w \in [(x_0^*x_1+x_2^*x_1)^*(x_0^*+x_2^*) x_1x_2^+x_3 (x_2^*x_3+x_0^*x_3)^*(x_0^*+x_2^*)] \\ \cap [(x_2^*x_3+x_0^*x_3)^*(x_0^*+x_2^*) x_3x_0^+x_1 (x_0^*x_1+x_2^*x_1)^*(x_0^*+x_2^*)]$$

D'où: $w \in [(x_0^*x_1+x_2^*x_1)^* x_2^+x_3 (x_2^*x_3+x_0^*x_3)^* x_0^+x_1] = H$.

(2) Cette propriété est une conséquence de la forme des mots de H .

(3) En constatant que $\pi_{\{x_1,x_3\}}(H) = [x_1^*x_3^*]$, nous en déduisons que:

$$\forall w \in H, \pi_{\{x_1,x_3\}}(w) \in x_1^*x_3^*x_1^* + x_3^*x_1^*x_3^*.$$

Par conséquent, $x_1x_3x_1x_3$ et $x_3x_1x_3x_1$ ne peuvent pas être des sous-mots de w . Rappel: $\pi_{\{x_1,x_3\}}$ représente la projection sur l'ensemble $\{ x_1, x_3 \}$.

(4) Soit $w \in \mathbb{H}$, nous avons $w \in [x_0^+x_1(x_0^*x_1+x_2^*x_1)^*x_2^+x_3(x_0^*x_3+x_2^*x_3)^*]$.

Si un facteur b du mot w est une boucle élémentaire non vide, il contient au moins une occurrence de chaque lettre de Π : par conséquent, il existe $k \in \{0,2\}$ tel que:

$$b \in (x_k^*x_{k+1}+x_{k+2}^*x_{k+1})^+x_{k+2}^+x_{k+3}(x_k^*x_{k+3}+x_{k+2}^*x_{k+3})^*(x_k^*+x_{k+2}^*).$$

De plus, nous avons $b \in SM(\mathbb{H})$, $\forall i \in \{0,2\} x_i x_{i+2} \in F([b])$ et $x_{k+1}x_{k+2}^+x_{k+3} \cap F([b]) \neq \emptyset$. D'après (1), il ne nous reste qu'à vérifier la propriété suivante:

$$x_{k+3}x_k^+x_{k+1} \cap F([b]) \neq \emptyset.$$

. Montrons que $b \in (x_k+x_{k+1})\Pi^*(x_k+x_{k+3})$:

Supposons que b débute ou se termine par une occurrence de x_{k+2} et décomposons b ainsi:

$$b = b_1 b_2 \text{ avec } b_1 = FG(b) \cap (x_k^*x_{k+1}+x_{k+2}^*x_{k+1})^*x_{k+2}^+ \\ \text{et } b_2 = FD(b) \cap x_{k+3}(x_k^*x_{k+3}+x_{k+2}^*x_{k+3})^*(x_k^*+x_{k+2}^*).$$

Comme $b \in BE$, nous avons $|b_1|_{x_{k+1}} = |b_2|_{x_{k+3}} = n$.

$$\text{Posons: } \forall j \in [0, n-1] \quad b_1 = u_{1,j} x_{k+1} v_{1,j} \text{ avec } |v_{1,j}|_{x_{k+1}} = j, \\ b_2 = u_{2,j} x_{k+3} v_{2,j} \text{ avec } |u_{2,j}|_{x_{k+3}} = j,$$

et constatons que:

$$|v_{1,0}u_{2,0}|_{x_{k+2}} > |v_{1,0}u_{2,0}|_{x_k} \text{ et } |v_{1,n-1}u_{2,n-1}|_{x_k} > |v_{1,n-1}u_{2,n-1}|_{x_{k+2}}.$$

Pour cela, il est nécessaire qu'il existe $j \in [0, n-2]$ tel que:

$$|v_{1,j}u_{2,j}|_{x_{k+2}} > |v_{1,j}u_{2,j}|_{x_k} \text{ et } |v_{1,j+1}u_{2,j+1}|_{x_k} > |v_{1,j+1}u_{2,j+1}|_{x_{k+2}}$$

et ainsi, il existe $f \in F(v_{1,j+1}u_{2,j+1})$ vérifiant: $|f|_{x_k} = |f|_{x_{k+2}}$ et $|f|_{x_{k+1}} = |f|_{x_{k+3}}$. Le mot f est une boucle, facteur propre de b . Contradiction.

. Si le facteur b débute par une occurrence de x_k ou de x_{k+1} , alors nécessairement $x_{k+3}x_k^+x_{k+1} \cap F([b]) \neq \emptyset$. ♦

Proposition 4.4.2:

$HP = H \cap BE$ est l'ensemble des mots de contour positifs des polyominos horizontalement convexes.

Preuve:

. Supposons qu'un mot $w \in HP$ soit un mot de contour d'un polyomino p non horizontalement convexe. Ce polyomino contient deux carrés unitaires sur la même ligne horizontale tels qu'il existe un carré unitaire entre eux n'appartenant pas à p : il existe $i \in \{0,2\}$ tel que $x_{i-1}x_{i+1}x_{i-1}x_{i+1} \in SM([w])$. Ceci est impossible [lemme 4.4.1 (3)].

- . D'après l'aspect d'un polyomino horizontalement convexe, nous concevons facilement que sa frontière soit caractérisée par un mot de H . ♦

Remarque: L'ensemble HP est fermé par conjugaison.

4.4.2 Quel est le problème ?

Soit $R \subset \Pi^*$, un langage reconnaissable. Le problème "*Le langage R contient-il un mot de contour positif de polyomino horizontalement convexe ?*" est-il décidable ?

Comme dans le cas précédent, ce problème est décidable si et seulement si le problème "*Le langage reconnaissable $S = [R] \cap H$ contient-il un mot de contour positif de polyomino horizontalement convexe ?*" est décidable.

Théorème 4.4.3:

Le problème de savoir si un langage reconnaissable de Π^ contient un mot de contour (positif) d'un polyomino horizontalement convexe est décidable.*

4.4.3 La preuve

Le raisonnement est le même que pour les polyominos convexes.

Lemme 4.4.4:

H est fermé par R

Preuve:

Nous montrons que: $\forall w \in H, R(w) \subset H$.

Si l'une des règles r de R_j ($j \in \{0,1,2,3\}$) peut être appliquée à w , nous vérifions que $r(w) \in H$: De façon évidente, $r(w) \in SM(H)$ et d'après le lemme 4.4.1 (1), il suffit de montrer que:

$$\forall i \in \{0,2\} \quad x_{i-1}x_i^+x_{i+1} \cap F([r(w)]) \neq \emptyset \text{ et } x_i x_{i+2} \notin F([r(w)]).$$

La règle r peut être appliquée à w si celui-ci contient un facteur de la forme $zx_{j-1}(x_j)^n x_{j+1}z'$ avec $n \geq 1$, $z \in \{x_{j-1}, x_j\}$, $z' \in \{x_j, x_{j+1}\}$ qui se transformera en $z(x_j)^n z'$ dans $r(w)$. La disparition des deux lettres (en gras) n'engendre pas de facteurs de la forme $x_i x_{i+2}$ dans $r(w)$, $i \in \{0,1,2,3\}$ (ceci est dû au contexte).

Il ne nous reste qu'à vérifier qu'il existe un mot de la forme $x_1 x_2^+ x_3$ et $x_3 x_0^+ x_1$ dans $F([r(w)])$. Nous avons:

$$w \in [x_0^+ x_1 (x_0^* x_1 + x_2^* x_1)^* x_2^+ x_3 (x_0^* x_3 + x_2^* x_3)^*]$$

ou plus précisément:

$$w \in [(x_0^+ + x_0^+ x_1 (x_0^* x_1 + x_2^* x_1)^* (x_0^* + x_2^*)) x_1 x_2^+ x_3 (x_0^* x_3 + x_2^* x_3)^*].$$

Seule une règle de R_2 peut altérer le facteur $x_1 x_2^+ x_3$ ($n > 0$) en détruisant les occurrences de x_1 et x_3 en gras: ceci est dû au contexte. Si $z=x_1$ et $z'=x_3$, il existe encore un facteur de la forme $x_1 x_2^+ x_3$ dans l'un des mots de $[w]$; Si $z=x_2$ (resp. $z'=x_2$), d'après la forme des mots de H , il existe nécessairement un facteur $x_1 x_2^*$ devant z (resp. un facteur $x_2^* x_3$ après z') dans l'un des mots de $[w]$. Par conséquent, $\forall r \in R \quad x_1 x_2^+ x_3 \cap F([r(w)]) \neq \emptyset$. Pour des raisons similaires, nous avons $x_3 x_0^+ x_1 \cap F([r(w)]) \neq \emptyset$. ♦

Lemme 4.4.5:

$$\text{Soient } w \in H \text{ et } w' \in R(w), \quad (w' \in HP \Rightarrow w \in HP)$$

Preuve:

En se référant au lemme 4.4.4, il suffit de montrer que:

$$\forall w \in H, \forall w' \in R(w), \quad (w' \in BE \Rightarrow w \in BE).$$

D'après la remarque 4.3.1, nous savons que: $(w \in B \Leftrightarrow w' \in B)$. Il ne nous reste qu'à montrer la propriété suivante:

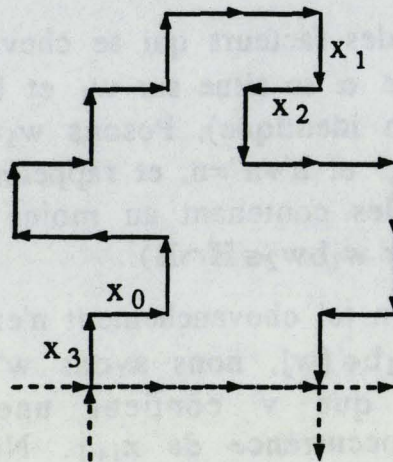
$$\begin{aligned} & \forall w \in H \cap B, \forall w' \in R(w) \\ & \text{(Aucun facteur propre de } w' \text{ n'est une boucle)} \\ & \Rightarrow \\ & \text{(Aucun facteur propre de } w \text{ n'est une boucle).} \end{aligned}$$

Il s'agit de vérifier que R ne permet pas de détruire des boucles dans w .

\Rightarrow : Considérons un mot $w \in H \cap B$ dont un facteur propre b est une boucle élémentaire. Nous montrons que: $\forall i \in \{0,1,2,3\}$, $\forall w' \in R_i(w)$, le mot w' contient également une boucle comme facteur propre.

Posons $w = w_1 b w_2$ avec $w_1, w_2 \in \Pi^*$. Pour tout mot $w' \in R_i(w)$, $i \in \{0,1,2,3\}$, il existe une décomposition de w vérifiant: $w = w_1 b w_2 = u_1 \alpha u_2$ avec $\alpha = x_{i-1}(x_i)^n x_{i+1}$, $n \geq 1$, $u_1, u_2 \in \Pi^*$ telle que $w' = u_1(x_i)^n u_2$.

Définissons une transformation T'_0 ainsi:



$$T'_0 = \{ z w z' \rightarrow z (x_0)^j z' \quad \text{avec } w \in (x_0^* x_3 + x_2^* x_3)^* x_0^+ x_1 (x_0^* x_1 + x_2^* x_1)^* \\ \text{et } w(x_2)^j \in \text{HP}, j \geq 1, z \in \{ x_3, x_0 \}, z' \in \{ x_0, x_1 \} \}$$

Remarque 4.4.6:

$$(w \in (x_0^* x_3 + x_2^* x_3)^* x_0^+ x_1 (x_0^* x_1 + x_2^* x_1)^* \text{ et } w(x_2)^j \in \text{HP}, j \geq 1) \Rightarrow (w \in (x_3 + x_2) \Pi^*).$$

Lemme 4.4.7:

Pour tout $R \in \text{Rec}(\Pi^*)$, $T'_0(R)$ est un langage reconnaissable constructible.

Preuve:

Il s'agit d'une preuve analogue à celle du lemme 4.3.6 en posant:

$$\tau'_0(R) = \bigcup_{\substack{q, q', q'' \in Q \\ q_f \in F}} r_1(q_0, q) \quad r_2(q, q', q'') \quad r_3(q', q_f)$$

avec

$$r_1(q_0, q) = R_{q_0 q} \cap \Pi^* (x_3 + x_0)$$

$$r_2(q, q', q'') = f_0(\pi_{\{x_0, x_2\}} \\ ((r'_2(q, q'') \sqcup r''_2(q'', q')) \cap (x_2^* x_0^* (x_3 x_1))^+ \cap L_0))$$

$$\text{avec } r'_2(q, q'') = (R_{q q''} \cap (x_0^* x_3 + x_2^* x_3)^* x_0^+)^r \text{ et}$$

$$r''_2(q'', q') = R_{q'' q'} \cap x_1 (x_0^* x_1 + x_2^* x_1)^*$$

$$r_3(q', q_f) = R_{q' q_f} \cap (x_0 + x_1) \Pi^*.$$

et $R_{q_1q_2}$, le langage reconnu par $M'=(\Pi, Q, \delta, q_1, \{q_2\})$;
 π_B , la projection sur l'ensemble B;
 f_0 , la réduction de Dyck sur $(x_0+x_2)^*$ définie par:
 $\forall w \in (x_0+x_2)^*$ avec $|w|_{x_0} - |w|_{x_2} = j, j \geq 0$, $f_0(w) = (x_0)^j$;
 $L_0 = \{x_0 w \in \Pi^* / \forall f \in FG(w) \mid |f|_{x_0} \geq |f|_{x_2}\}$, un langage algébrique.

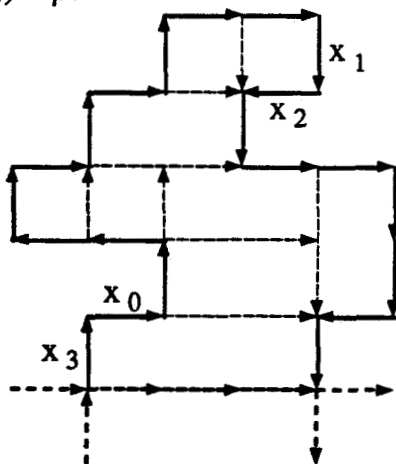
De manière analogue, nous avons: $\forall R \in Rec(\Pi^*) : \Gamma'_0(R) = \tau'_0(R)$. Comme $\tau'_0(R)$ est un ensemble reconnaissable que nous savons construire, le lemme est démontré. ♦

Lemme 4.4.8:

Pour tout mot $m \in H$, $\Gamma'_0(m) \subset R^(m)$*

Preuve:

Pour tout mot $m \in H$ et pour tout mot $m' \in \Gamma'_0(m)$, nous pouvons décomposer m de la façon suivante: $m = \alpha z w z' \beta$ avec $z \in \{x_3, x_0\}$, $z' \in \{x_0, x_1\}$, $w \in (x_0^* x_3 + x_2^* x_3)^* x_0 + x_1 (x_0^* x_1 + x_2^* x_1)^*$, $w(x_2)^j \in HP, j \geq 1$, et $\alpha, \beta \in \Pi^*$ tel que $m' = \alpha z (x_0)^j z' \beta$.



Nous montrons par récurrence sur $|w|$ que $m' \in R^*(m)$.

Puisque $w(x_2)^j \in HP$, il existe nécessairement une occurrence de x_3 dans w . Par conséquent, posons $w = u x_3 (x_0)^n x_1 v$ avec $n \geq 1, u, v \in \Pi^*$.

Cas 1: $v = \epsilon$.

Puisque $w(x_2)^j \in HP$ et $v = \epsilon$, le facteur u ne peut pas contenir d'occurrence de x_3 : nécessairement, $u \in (x_0^* + x_2^*)$.

Or d'après la remarque 4.4.6, nous savons que la première lettre de w est différente de x_0 : nécessairement, $u \in x_2^*$.

- . Si $u = \varepsilon$: nous obtenons $n=j$ et $m \in R_0(m)$.
- . Si $u \in x_2^+$: une règle de R_3 s'applique et le mot obtenu vérifie les hypothèses de la récurrence.

Cas 2: . $v \in (x_0+x_1)\Pi^*$.

D'après la forme du facteur w , nous avons $|v|_{x_1} \geq 1$ et le facteur u contient au moins une occurrence de x_3 .

. Si $u \in \Pi^*(x_3+x_0)$: une règle de R_0 s'applique et le mot obtenu vérifie les hypothèses de la récurrence.

. Si $u \in \Pi^*x_2$: nécessairement $u \in \Pi^*x_3x_2^+$, une règle de R_3 s'applique et le mot obtenu vérifie les hypothèses de la récurrence.

Cas 3&4: . $v \in x_2\Pi^*$: nécessairement $v \in x_2^+x_1\Pi^*$, une règle de R_1 s'applique et le mot obtenu vérifie les hypothèses de la récurrence. ♦

En posant $T' = \{ T'_0, T_1 \}$

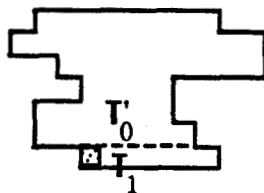
Proposition 4.4.9:

Pour tout mot $w \in H$, ($w \in HP \Leftrightarrow \exists i, x_i x_{i+1} x_{i+2} x_{i+3} \in T'^{s2}([w])$).

Preuve:

Les affirmations suivantes découlent du fait qu'un polyomino horizontalement convexe peut être vu comme un empilement non axé de rectangles d'épaisseur égale à 1.

\Rightarrow : Soit w , un mot de HP et notons p , le polyomino horizontalement convexe caractérisé par w . Supposons que p ne soit pas un rectangle d'épaisseur 1. Nous appliquons à un certain mot de $[w]$ une règle de T'_0 de façon à ce que le polyomino p' issu de cette transformation soit le rectangle d'épaisseur 1 le plus bas de p . Si p' n'est pas un carré unitaire, il suffit d'appliquer une règle de T_1 pour obtenir le résultat désiré.



\Leftarrow : La réciproque est une conséquence directe des lemmes 4.4.8 et 4.4.5. ♦

Le théorème 4.4.3 est démontré. ♦

4.5 Autres extensions envisagées

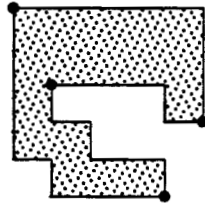
Nous aimerions étendre ce résultat à des classes plus générales de polyominos. Qu'en est-il pour des polyominos où nous bornerions le nombre de changements de direction en diagonale ?

$$\text{Posons } M = \bigcup_{i=0}^{i=3} (x_i + x_{i+1})^*.$$

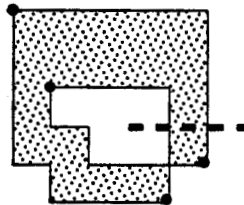
. Un polyomino p est k -borné s'il existe un mot de contour de p dans

$$kP = M^k \cap BE.$$

Exemple: Ce polyomino est 4-borné.



A priori, le principe de "couper des branches" afin d'obtenir un carré unitaire ne semble pas adapté: nous risquons de transformer une boucle non élémentaire en une boucle élémentaire.



Est-il possible que pour de "petites valeurs" de k , la question précédente reste décidable ? A partir de quel rang devient-elle indécidable ?

CONCLUSION

CONCLUSION

Conclusion

Dans ce travail, nous nous sommes intéressés à des systèmes de réécriture particulièrement adaptés aux langages de mots de figure.

Dans les deux premières parties consacrées à la description des figures connexes ou non connexes par des mots de figures, nous étudions trois systèmes de réécriture S , $Sr\text{-red}$ et S' dont la principale caractéristique est de conserver les figures décrites au cours des dérivations d'un mot de figure. Nous voyons que les systèmes S et S' sont complets (ils permettent à partir de tout mot de Π^* ou de Π'^* , de dériver tous les mots ayant la même figure décrite que le mot initial) et que $Sr\text{-red}$, autorisant moins de manipulations sur les mots de figure mais conservant la rationalité du langage, est un système confluent. Nous décrivons des méthodes pour dériver tout mot de figure en un mot plus court et nous obtenons un mot minimal si la figure associée est connexe.

La troisième partie établit la décidabilité de la question suivante: Si $R \subset \Pi^*$ est un langage rationnel, ce langage R contient-il un mot de contour de polyomino convexe (resp. horizontalement convexe et verticalement convexe) ? Nous prouvons ce résultat à l'aide de deux systèmes de réécriture R et T : en montrant d'une part que les mots de figure dérivés par R conservent leur non-appartenance à l'ensemble des mots de contour de la classe de polyominos considérée, et que d'autre part, le système T (pouvant être simulé par des règles de R) est tel que $T(R)$ est un langage reconnaissable que nous savons construire.

Sachant que la question précédente est indécidable pour les polyominos quelconques [Bd] et décidable pour les polyominos convexe et semi-convexes, essayons de restreindre l'intervalle où nous ne savons pas encore. Considérons les polyominos k -bornés définis au chapitre 4: Pouvons-nous croire que pour de "petites valeurs" de k , cette question reste décidable? A partir de quel rang devient-elle indécidable?

Conclusion

Dans ce travail nous nous sommes intéressés à des systèmes de réduction particulièrement adaptés aux langages de mots de figures.

Dans les deux premières parties consacrées à la description des figures connues ou non connues par des mots de figures nous étudions deux systèmes de réduction S_1 et S_2 dont la principale caractéristique est de conserver les figures décrites au cours des dérivations dans un langage. Nous voyons que les systèmes S_1 et S_2 sont complètes (ils permettent à partir de tout mot de la forme 1^n de dériver tous les mots ayant la même figure décrite que le mot initial) et que S_1 est strictement moins de manipulations que les mots de figure dans la conservation la totalité du langage est un système complet. Nous dérivons des méthodes pour dériver tout mot de figure en un mot plus court et pour obtenir un mot minimal si la figure initiale est connue.

La troisième partie est une conclusion de la description précédente. Elle est en langage naturel ce langage R contenant un mot de contenu de polynômes connus (voir l'introduction) connus et généralement connus. Les deux premiers de ces mots 1 et 2 sont de systèmes de réduction S_1 et S_2 en dérivant d'un mot que les mots de figure dérivés par S_1 conservent leur espérance à l'origine des mots de contenu de la classe de polynômes connues et que S_2 par le système S_1 peuvent être réduits par des règles de R en tel que $T(R)$ est un langage reconnaissable que nous avons construit.

Enfin une question préliminaire est indiquée pour les polynômes particuliers (voir la section) pour les polynômes connus et pour ceux connus de réduction de réduction (voir la section) ou nous ne savons pas encore. Connaître les polynômes à donner définis au chapitre 2. Nous nous donnons pour de "petits valeurs" de R dans ce cas pour une réduction. A partir de quel mot devient-elle reconnaissable?

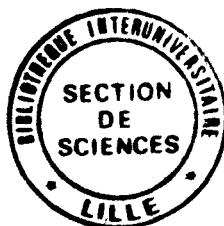
REFERENCES

Références

- [Au] : J.M. Autebert, *Langages algébriques* (Masson, 1987).
- [Bc] : C. Berge, *Théorie des graphes et ses applications* (Dunod, 1963).
- [BC] : F.J. Brandenburg, M.P. Chytil, On picture Languages: Cycles and Syntax-directed Transformations, preprint (1991), Université de Passau.
- [Bd] : D. Beauquier, An Undecidable Problem about Rational Sets and Contour Words of Polyominoes, *Information Processing Letters* 37 (1991), 257-263.
- [BD] : F.J. Brandenburg, J. Dassow, Reductions of Picture Words, submitted for publication.
- [Bj] : J. Berstel, *Transductions and Context-Free Languages* (Teubner, 1979), 36-40.
- [BLS] : D. Beauquier, M. Latteux, K. Slowinski, A Decidability Result about Convex Polyominoes, to appear in Springer Verlag Lecture Notes (LATIN'92).
- [Bm] : M. Benois, Parties Rationnelles du Groupe Libre, C. R. Acad. Sc. Paris, 269 (1969), 1188-1190.
- [BN] : D. Beauquier, M. Nivat, On Translating One Polyomino to Tile the Plane, *Discrete and Computational Geometry*, to appear.
- [Da1] : J. Dassow, Convexity and Simplicity of Chain Code Picture Languages, *Rostock. Math. Kolloq.* 34 (1988), 53-60.
- [Da2] : J. Dassow, Graph-theoretical Properties and Chain Code Picture Languages, *J. Inf. Process. Cybern.* EIK25 (1989), 423-433.
- [Da3] : J. Dassow, On the Connectedness of Pictures in Chain Code Picture Languages, *Theoretical Computer Science* 81 (1991), 289-294.
- [DH1] : J. Dassow, F. Hinz, Decision Problems and Regular Chain Code Picture Languages, to appear in *Discrete Applied Mathematics*.
- [DH2] : J. Dassow, F. Hinz, An Undecidability Result for Regular Languages and its Applications to Regulated Rewriting, *Bulletin of the EATCS* 38 (1989), 168-173.
- [DV] : M.P. Delest, G. Viennot, Algebraic Languages and Polyominoes Enumeration, *Theoretical Computer Science* 34 (1984), 169-206.
- [EJ] : J. Edmonds, E.L. Johnson, Matching, Euler tours and the Chinese Postman, *Math. Programming* 5 (1973), 88-124.

- [Fr] : H. Freeman, On the Encoding of Arbitrary Geometric Configurations, *Ire transactions on electronic computers* 10 (1961) 260-268.
- [GM] : M. Gondran, M. Minoux, *Graphes et algorithmes* (Eyrolles, 1990).
- [Gu1] : R. Gutbrod, A Transformation System for Generating Description Languages of Chain Code Pictures, *Theoretical Computer Science* 68 (1989), 239-252.
- [Gu2] : R. Gutbrod, A Transformation System for Chain Code Picture Languages: Properties and Algorithms, Université de Würzburg, Techn. Report n°10 (1990).
- [GS] : B. Gröbaum, G.C. Shephard, *Tilings and Patterns* (W.H. Freeman and Company New York, 1986).
- [Ha] : M. Harrison, *Introduction to formal language theory* (Addison-Wesley, reading, Mass, 1978).
- [Hi1] : F. Hinz, Question of Decidability for Context-free Chain Code Picture Languages, *Lecture Notes in Computer Science* 381 (Proc. IMYCS, 1988), 2-11.
- [Hi2] : F. Hinz, Classes of Picture Languages that Cannot be Distinguished in the Chain Code Concept and Deletion of Redundant Retreats, *Lecture Notes in Computer Science* 349 (Proc. STACS, 1989), 132-143.
- [Hi3] : F. Hinz, The Membership Problem for Context-free Chain Code Picture Languages, *Lecture Notes in Computer Science* 452 (Proc. MFCS, 1990), 329-336.
- [HU] : J.E. Hopcroft, J.D. Ullman, *Introduction to Automata Theory, Languages and Computation*, (Addison-Wesley, reading, Mass, 1979).
- [HW] : F. Hinz, E. Welzl, Regular Chain Code Picture Languages with Invisible Lines, Université de Graz, Report 252 (1988).
- [Ja] : M. Jantzen, *Confluent string rewriting*, EATCS Monographs on theoretical computer science 14 (Springer-Verlag, 1988).
- [Ki1] : C. Kim, Complexity and Decidability for Restricted Classes of Picture Languages, *Theoretical Computer Science* 73 (1990), 295-311.
- [Ki2] : C. Kim, Picture Iteration and Picture Ambiguity, *J. Computer and System Sciences* 40 (1990), 289-306.
- [KS1]: C. Kim, I.H. Sudborough, The Membership and Equivalence Problems for Picture Languages, *Theoretical Computer Science* 52 (1987), 177-191.
- [KS2]: C. Kim, I.H. Sudborough, On Reversal Bounded Picture Languages, to appear in *Theoretical Computer Science*.
- [MRW] : H.A. Maurer, G. Rozenberg, E. Welzl, Using String Languages to Describe Picture Languages, *Inform. and Control* 54 (1982), 155-185.

- [Pa]: G. Paun, Fives Remarks (and two open problems) about the Double Segments and the Invisible Segments in Picture Description Languages, manuscript (1990), Institut de Mathématique de Bucarest.
- [Pé]: J.P. Pécuchet, Automates Boustrophédon, Semi-groupe de Birget et Monoïde Inversif Libre, *R.A.I.R.O. Informatique Théorique* 19 (1985), 71-100.
- [Pi]: J.E. Pin, *Variétés de langages formels* (Masson, 1984).
- [Ro]: D. Robilliard, Communication personnelle.
- [Sa]: A. Salomaa, *Formal languages* (Academic Press, London, 1973).
- [Sl]: K. Slowinski, Picture Descriptions with Invisible Lines, Notes to appear in *Theoretical Computer Science*.
- [SS1]: P. Séébold, K. Slowinski, Minimizing Picture Words, *Lecture Notes in Computer Science* 464 (Proc. IMYCS, 1990), 234-243.
- [SS2]: P. Séébold, K. Slowinski, The Shortest Way to Draw a Connected Picture, *Computer Graphics Forum* 10 (1991), to appear.
- [SS3]: P. Séébold, K. Slowinski, Redundant Retreat Free Word, Submitted for publication.
- [St]: I. Stewart, Pavages et Tâtonnements, *Pour la science* 150 (1990), 94-99.
- [SW]: I.H. Sudborough, E. Welzl, Complexity and Decidability for Chain Code Picture Languages, *Theoretical computer Science* 36 (1985), 173-202.

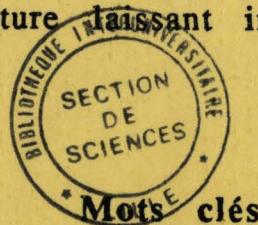


044 18 6665

Résumé

Une figure 2D constituée de segments unitaires horizontaux et verticaux peut être représentée par une séquence de symboles, chacun d'entre eux engendrant un déplacement unitaire dans l'une des quatre directions: droite, gauche, haut ou bas. L'ensemble de ces symboles est noté Π . Ainsi, à tout mot de Π^* est associée une unique figure inscrite dans le plan cartésien. Nous étudions différents systèmes de réécriture définis sur Π^* et conservant certaines propriétés des figures associées.

D'abord, l'alphabet Π contient quatre lettres engendrant des déplacements visibles (crayon baissé). Nous étudions deux systèmes de réécriture S et S_r -red conservant les figures: l'un d'entre eux permet d'obtenir tous les mots décrivant la même figure, l'autre est un système confluent fournissant un unique mot irréductible. Puis, nous complétons cet alphabet avec quatre lettres "blanches" engendrant des déplacements invisibles (crayon levé). Nous définissons un nouveau système de réécriture S' permettant d'obtenir exactement tous les mots décrivant la même figure qu'un mot initial. Pour chacun de ces deux alphabets, nous recherchons un parcours minimal décrivant une figure connexe donnée. Enfin, nous terminons par des résultats de décidabilité relatifs aux langages de mots de figure. Etant donné un langage rationnel inclus dans Π^* , nous pouvons décider si celui-ci contient des mots de contour de polyominos particuliers. Ces résultats sont établis à l'aide de deux systèmes de réécriture laissant invariantes certaines propriétés des mots de figure.



Systeme de Réécriture, Mot de Figure, Langage Rationnel, Polyomino.