Université " Med Boudiaf " - M'sila

Faculté des Mathématiques et de l'informatique
Département de Mathématiques

Polycopié de cours : Master 1ére année

Spécialité : Algèbre et Mathématiques Discrètes

Nacer Ghadbane

Semi groupes et automates finis

Avis du comité scientifique	Avis du conseil scientifique

Année 2017/2018

Notations

Σ : alphabet fini.

Σ^* : monoïde libre sur Σ.

$|w|$: la longueur du mot w.

$|w|_\sigma$: le nombre d'occurence de la lettre σ dans le mot w.

$S = (\Sigma, \mathcal{R})$: un semi système de réécriture de mots.

$IRR(w)$: le mot irréductible de w.

L : langage sur l'alphabet Σ.

$\prod_{i=1}^{i=n} E_i$: produit cartésiens d'ensembles.

$\mathcal{R}$: relation binaire.

$\mathcal{R}^0$: la relation d'identité.

$\rceil(\mathcal{R})$: le complémentaire de la relation $\mathcal{R}$.

$\mathcal{R}^n$: la n-ième composition de $\mathcal{R}$.

$\mathcal{R}^r$: la fermeture réflexive de $\mathcal{R}$.

$\mathcal{R}^s$: la fermeture symétrique de $\mathcal{R}$.

$\mathcal{R}^+$: la fermeture transitive de $\mathcal{R}$.

$\mathcal{R}^*$: la fermeture réflexive et transitive de $\mathcal{R}$.

$\mathcal{R}^{rst}$: la fermeture d'équivalence de $\mathcal{R}$.

S : semigroupe.

$End\,(S)$: l'ensemble des morphismes de S vers S.

$\cong$: isomorphe.

$\geq$: ordre bien fondé.

$>$: ordre strict.

$\preceq$: ordre lexicographique.

$L(\Gamma, \mathcal{R})$: langage engendré par un système de réécriture.

$\varphi\,(n)$: indicateur d'Euler.

$C\,(n)$: complexité d'un algorithme.

$Hom\,(\Sigma^*, \Delta^*)$: l'ensemble des morphismes de Σ^* vers Δ^*.

$Iso\,(\Sigma^*, \Delta^*)$: l'ensemble des isomorphismes de Σ^* vers Δ^*.

$\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$: automate fini.

$\left(Q^Q, \circ\right)$: monoïde de toutes les fonctions de Q vers Q.

$\Sigma^* / \sim_L$: monoïde quotient.

$Rat\left(\Sigma^*\right)$: les langages rationneles sur Σ.

$G = (V, \Sigma, P, S)$: grammaire algébrique.

$\mathcal{F}_p$: l'ensemble des applications de $\mathbb{N}^p$ dans $\mathbb{N}$.

$g \circ f$: application composée de f et g.

id_E : application identité de l'ensemble E.

$\max(A)$: plus grand élément de l'ensemble ordonné A.

$\min(A)$: plus petit élément de l'ensemble ordonné A.

Introduction générale

Le présent polycopié reprend un cours de premiére année Master, spécialité Algèbre et Mathématiques Discrètes, donné à l'Université de Mohamed Boudiaf-M'sila pendant les années 2016-2018. Le but de ce cours était de présenter aux étudiants les notions de base concernant les semigroupes, les automates finis et les grammaires algébriques. Nous supposons que le lecteur a une bonne connaissance de les premiers principes de la théorie des ensembles.

Ce travail se situe dans le cadre de la théorie des semigroupes, automates finis et des langages formels. La théorie des langages formels est née d'une tentative de modélisation des langues naturelles.

Historiquement, les deux mécanismes très connus pour définir un langage de mots finis d'une manière formelle sont principalement les suivants :

(1) Un mécanisme qui consiste à donner un processus de genération des mots, qui conduit à la notion de grammaire.

(2) Un deuxième mécanisme de reconnaissance qui est réalisé à l'aide d'automate.

Ce travail est composé de cinque chapitres.

Le premier chapitre consiste en un rappel des notions et notations utilisées par la suite :

relations binaires et leurs propriétés, monoïdes, mots et langages, homomorphismes des monoïdes.

Dans le second chapitre, on fait une étude sur les semi-systèmes de réécriture ainsi que certaines de leurs propriétés telles que : la terminaison et la confluence.

Dans le troisième chapitre, on donne les notions et les propriétés de base des automates finis.

Nous aborderons et traiterons, dans le quatrième chapitre la notion de grammaire algébrique.

Le chapitre cinque sera consacré au fonctions primitives récursives, complexité d'un algorithme et l'indécidabilité.

Nous avons d'ailleurs inclus un nombre considérable d'exemples. Les chapitres de ce polycopié ce terminent par des exercices non corrigés.

Nous tenons, à la fin de cette petite introduction, à solliciter la haute bienveillance de nos lecteurs de nous faire parvenir toutes leurs remarques via notre adresse E-mail : nacer.ghadbane@yahoo.com.

Table des matières

1 Notions élémentaires 7

1.1 Relations et lois de compositions internes 8
1.2 Monoïdes libres, mots, langages 14
1.3 Ensembles définis inductivement 24
1.4 Exercices 26

2 Les semi systèmes de réécriture de mots 36

2.1 Définitions et propriétés 38
2.2 La terminaison d'un semi-système de réécriture de mots. 40
2.3 La confluence d'un semi-système de réécriture de mots. . 47
2.4 Exercices 50

3 Automates et langages rationnels 53

3.1 Notations et définitions. 54
3.2 Langages rationnels et automates finis 58
3.3 Automate minimal 59
3.4 Exercices 62

4 Langages algébriques 64

4.1 Notations et définitions 65

4.2 Grammaires et langages réguliers 66
4.3 Exercices 67

5 Calculabilité, Complexité des algorithmes 70

5.1 Fonctions primitives récursives 71
5.2 Complexité d'un algorithme 73
5.3 L'indécidabilité 74
5.4 Exercices 75

Chapitre 1

Notions élémentaires

Introduction

Ce premier chapitre contient les définitions et les propriétés des outils que nous utiliserons par la suite : relations et lois de compositions internes, monoïdes libres, mots, langages.

Contenu

1.1. Relations et lois de compositions internes.
1.2. Monoïdes libres, mots, langages.
1.3. Ensembles définis inductivement.
1.3. Exercices.

1.1 Relations et lois de compositions internes

Dans ce qui suit, on donne quelques définitions et notations concernant les relations binaires et les lois de compositions internes.

Intuitivment une relation entre deux ensembles d'objets signifie que pour un caractère donné, certain éléments du premier ensemble sont en correspondance avec certains du second. Par exemple si F désigne l'ensemble de femmes et H celui des hommes, un élément $h \in H$ sera en relation avec un élément $f \in F$ si h est fils biologique de f. Alors la relation " être fils biologique de " est une relation de H vers F définie par les couples (h, f) qui la vérifient.

Définition 1.1.1 : Une relation binaire est un triplet $\mathcal{R} = (E, F, G)$. E est l'ensemble de départ de $\mathcal{R}$, F son ensemble d'arrivée et G une partie du produit cartésien $E \times F = \{(x, y) / x \in E, y \in F\}$, appelée graphe de $\mathcal{R}$ et souvent notée $\widehat{\mathcal{R}}$. On note $\mathcal{R} : E \longrightarrow F$, pour désigner une relation $\mathcal{R}$ de E dans F. On écrit $x\mathcal{R}y$ pour signifier que $(x, y) \in \widehat{\mathcal{R}}$. Lorsque $E = F$, on dit que $\mathcal{R}$ est une relation sur E.

Exemple 1.1.2 : On désigne par $\omega = (E, F, E \times F)$ la relation universelle de E vers F.

Définition 1.1.3 : Soit $\mathcal{R} \subseteq E \times F$ une relation binaire, on note $\mathcal{R}(e) = \{f \in F / e\mathcal{R}f\}$, l'image par $\mathcal{R}$ de e. $\mathcal{R}(e)$ étant pour tout e une partie de F, c'est-à-dire un élément de l'ensemble 2^F des parties de F, $\mathcal{R}$ induit une application ψ de E vers 2^F définie par : pour tout $e \in E, \psi(e) = \mathcal{R}(e) = \{f \in F / e\mathcal{R}f\}$.

Définition 1.1.4 : Soit $\mathcal{R} = (E, F, G)$ une relation binaire. Nous disposerons plusieurs représentations équivalentes de $\mathcal{R}$.

- Exhaustive : par la liste de toutes les paires $(e, f) \in \widehat{\mathcal{R}}$.
- Maricielle : par une matrice booléenne $M(\mathcal{R})$, $M(\mathcal{R}) \in \boldsymbol{M}_{|E|\times|F|}(\{0, 1\})$ avec

$$M(\mathcal{R})_{e,f} = \begin{cases} 1 \text{ si } (e, f) \in \widehat{\mathcal{R}} \\ 0 \text{ sinon.} \end{cases}$$

- Sagittale : par un ensemble de sommets $E \cup F$ et des flèches reliant $e \in E$ à $f \in F$ si, et seulement si, $(e, f) \in \widehat{\mathcal{R}}$.

• Cartésienne : elle utilise la fonction caractéristique de $\mathcal{R}$, c'est à dire la fonction $\chi_{\mathcal{R}}$ de $E \times F$ vers $\{0,1\}$ définie par : $\begin{cases} \chi_{\mathcal{R}}(e,f) = 1 & \text{si } (e,f) \in \widehat{\mathcal{R}} \\ \chi_{\mathcal{R}}(e,f) = 0 & \text{sinon} \end{cases}$.
On représente les éléments de E par des points sur une droite, et les éléments de F par des points sur une autre droite perperdiculaire. On met un point à l'intersection des parllèles en f et en e à ces deux droites respectivement si et seulement si $\chi_{\mathcal{R}}(e,f) = 1$.

Définition 1.1.5 : Toute relation $\mathcal{R} : E \longrightarrow F$ possède une relation inverse (ou réciproque), notée $\mathcal{R}^{-1}$. Il s'agit de la relation de F vers E qui est caractérisée par : $y\mathcal{R}^{-1}x \Longleftrightarrow x\mathcal{R}y$.

Définition 1.1.6 : On appelle relation identique sur E, notée $\mathbf{1}_E$ la relation définie par : $(e,e') \in \mathbf{1}_E \Longleftrightarrow e = e'$. ou encore par la diagonal du produit cartésien $E \times E$:

$$\mathbf{1}_E = \{(e,e)\,/e \in E\}.$$

Définition 1.1.7 : Les relations étants des parties de $E \times F$, on définit de manière usuelle le complémentaire $\mathcal{R}^c$, la réunion $\mathcal{R} \cup \mathcal{S}$ et l'intersection $\mathcal{R} \cap \mathcal{S}$ de relations $\mathcal{R}$ et $\mathcal{S}$. On a :

- $(x,y) \in \mathcal{R} \cup \mathcal{S}$ si, et seulement si $x\mathcal{R}y$ ou $x\mathcal{S}y$.
- $(x,y) \in \mathcal{R} \cap \mathcal{S}$ si, et seulement si $x\mathcal{R}y$ et $x\mathcal{S}y$.
- $x\mathcal{R}^c y$ si, et seulement si $(x,y) \notin \mathcal{R}$.

Définition 1.1.8 : Soient $\mathcal{R} : E \longrightarrow F$ et $\mathcal{S} : F \longrightarrow G$ deux relations. La relation composée notée $\mathcal{S} \circ \mathcal{R} : E \longrightarrow G$ a pour expression : $x(\mathcal{S} \circ \mathcal{R})z \Longleftrightarrow \exists y \in F, x\mathcal{R}y$ et $y\mathcal{S}z$.

Proposition 1.1.9 :

• Si $\mathcal{R}$ et $\mathcal{S}$ sont deux relations de E vers F, alors $M(\mathcal{R} \cup \mathcal{S}) = M(\mathcal{R}) \vee M(\mathcal{S})$
et $M(\mathcal{R} \cap \mathcal{S}) = M(\mathcal{R}) \wedge M(\mathcal{S})$.

• Si $\mathcal{R}$ est une relation de E vers F et $\mathcal{S}$ est une relation de F vers G, alors

$$M(\mathcal{S} \circ \mathcal{R}) = \underset{f \in F}{\vee} \Big(M(\mathcal{R})_{e,f} \wedge M(\mathcal{S})_{f,g} \Big).$$

Proposition 1.1.10 : Soient des relations $\mathcal{R} \subseteq E \times F$, $\mathcal{S} \subseteq F \times G$ et $\mathcal{T} \subseteq G \times H$. Alors

- $\mathcal{T} \circ (\mathcal{S} \circ \mathcal{R}) = (\mathcal{T} \circ \mathcal{S}) \circ \mathcal{R}$.
- $\mathbf{1}_F \circ \mathcal{R} = \mathcal{R} = \mathcal{R} \circ \mathbf{1}_E$.
- $(\mathcal{S} \circ \mathcal{R})^{-1} = \mathcal{R}^{-1} \circ \mathcal{S}^{-1}$.

Exemple 1.1.11 : Soient $E = \{1, 2, 3\}$ et $\mathcal{R}$ la relation sur E décrite en représentation matricielle par : $M(\mathcal{R}) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$. Alors $\mathcal{R}^{-1}$ est représentée dans la même reprèsentation par : $M(\mathcal{R}^{-1}) = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$. Nous aurons alors $M\left(\mathcal{R} \circ \mathcal{R}^{-1}\right) = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$ et $M(\mathcal{R}^{-1} \circ \mathcal{R}) = \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix}$.

Proposition 1.1.12 : Soit E un ensemble, F_0 une partie non vide de E et $\mathcal{R}$ une relation sur E. Considérons la suite $(F_i)_{i\in\mathbb{N}}$ de parties de E définie par :

$$F_{i+1} = F_i \cup \{x \in E / \exists y \in F_i \text{ tel que } x\mathcal{R}y\}.$$

S'il existe $i \in \mathbb{N}$ tel que $F_{i+1} = F_i$, alors pour tout $k \in \mathbb{N}$, $F_{i+k} = F_i$.

Preuve : Par l'absurde, supposons qu'il existe $j > i$ tel que $F_{j+1} \neq F_j$. On peut supposer que j est le plus petit entier tel que cette inégalité soit satisfaire, et donc que $F_j = F_{j-1}$. Soit $z \in F_{j+1} - F_j$. Par définition, $\exists y \in F_j$ tel que $z\mathcal{R}y$, et donc $\exists y \in F_{j-1}$ tel que $z\mathcal{R}y$, c'est-à-dire $z \in F_{(j-1)+1} = F_j$, contradiction.

Corollaire 1.1.13 : Si E est un ensemble fini, étant donnés F_0 et une ralation finie $\mathcal{R}$, on peut calculer effectivement l'ensemble $F = \bigcup_{i\in\mathbb{N}} F_i$.

Définition 1.1.14 : Soit $\mathcal{R}$ une relation binaire sur un ensemble E. On dit que $\mathcal{R}$ est :

- réflexive si : $\forall x \in E, x\mathcal{R}x$.
- antiréflexive si : il n'existe pas un élément $x \in E$ qui vérifie $x\mathcal{R}x$.
- symétrique si : $\forall x, y \in E, x\mathcal{R}y \Longrightarrow y\mathcal{R}x$.
- antisymétrique si : $\forall x, y \in E, x\mathcal{R}y$ et $y\mathcal{R}x \Longrightarrow x = y$.
- transitive si : $\forall x, y, z \in E, x\mathcal{R}y$ et $y\mathcal{R}z \Longrightarrow x\mathcal{R}z$.

Une relation $\mathcal{R}$ à la fois réflexive et transitive est une relation de préordre. Si de plus $\mathcal{R}$ est antisymétrique, on dit qu'il s'agit d'une relation d'ordre partiel.

Une relation d'ordre total est une relation d'ordre $\mathcal{R}$ telle que pour tout couple $(x, y) \in E \times E$, on ait $x\mathcal{R}y$ ou $y\mathcal{R}x$.

Une relation $\mathcal{R}$ sur E est localement finie si $\forall e \in E, \mathcal{R}(e)$ est fini.

Définition 1.1.15 : Une relation d'équivalence est une relation $\mathcal{R}$ sur E réflexive, symétrique et transitive. La partie $C(x) = \{y \in E / x\mathcal{R}y\}$, pour un élément $x \in E$, est appelé classe déquivalence de x modulo $\mathcal{R}$. Les classes d'équivalences modulo $\mathcal{R}$ forment une partition de E. L'ensemble de toutes ces classes, noté $E/\mathcal{R}$, est appelé ensemble quotient de E par $\mathcal{R}$: $E/\mathcal{R} = \{C(x) / x \in E\}$. L'index de $\mathcal{R}$ est le cardinal de $E/\mathcal{R}$. On définit la projection canonique $\pi : E \longrightarrow E/\mathcal{R}$ qui associe x à $C(x)$.

Définition 1.1.16 : On appelle chaîne pour une relation $\mathcal{R}$ sur E une suite finie ou infinie d'éléments : $e_0, e_1, ..., e_n, ...$ telle que, pour tout $i \geq 0, (e_i, e_{i+1}) \in \mathcal{R}$. Si $e_0, e_1, ..., e_n$ est une chaîne finie, l'entier n est la longueur de la chaîne. Un cycle pour une relation $\mathcal{R}$ sur E est une chaîne finie non vide $e_0, e_1, ..., e_n$ telle que $e_0 = e_n$. Une relation $\mathcal{R}$ est dite acyclique si elle est sans cycle. Une relation est noethérienne s'il n'existe pas de chaîne infinie.

Définition 1.1.17 : Une relation $\mathcal{R}$ sur E est bornée si pour tout $e \in E$, il existe $n \in \mathbb{N}$ tel que toute chaîne admettant e comme premier élément est de longueur bornée par n. Une relation bornée est donc noethérienne, la réciproque étant fausse.

Remarque 1.1.18 : L'ensemble des relations sur E peut à son être muni d'une relation d'ordre partiel. On ordonne ses éléments par inclusion de leurs graphes. Ainsi, on dira que $\mathcal{R}$ est plus fine que $\mathcal{S}$, ou que $\mathcal{S}$ est plus grossière que $\mathcal{R}$, si $\widehat{\mathcal{R}} \subset \widehat{\mathcal{S}}$.

Définition 1.1.19 : Soit $\mathcal{R}$ une relation binaire sur un ensemble E et $X \subseteq E$.

Soit $x \in X$, l'élément x est irréductible dans X pour la relation $\mathcal{R}$ s'il n'existe aucun y dans X tel que $x\mathcal{R}y$. x est irréductible pour $\mathcal{R}$ s'il est irréductible dans E.

Proposition 1.1.20 : Soit $\mathcal{R}$ une relation binaire sur un ensemble E.

- Si $\mathcal{R}$ est globalement finie et acyclique, alors elle est bornée.
- Si $\mathcal{R}$ est localement finie et noethérienne, alors elle est bornée.
- Si $\mathcal{R}$ est localement finie et acyclique, alors les propriétés suivantes sont équivalentes :

(1) $\mathcal{R}$ est noethérienne.
(2) $\mathcal{R}$ est bornée.
(3) $\mathcal{R}$ est globalement finie.

- Les propriétés suivantes sont équivalentes :

(1) $\mathcal{R}$ est noethérienne.

(2) Pour toute partie non vide X de E, il existe $x \in X$ avec x irréductible dans X.

Définition 1.1.21 : Soient $E_1, ..., E_k$ des ensembles partiellement ordonnés par les relations $\leq_i$, $i = 1, ..., k$, respectivement. L'ordre lexicographique $\preceq$ sur $E_1 \times ... \times E_k = \prod_{i=1}^{i=k} E_i$ est défini par : $(x_1, ..., x_k) \preceq (y_1, ..., y_k)$ si soit $(x_1, ..., x_k) = (y_1, ..., y_k)$, soit il existe un d, $1 \leq d \leq k$ tel que $x_d \leq_d y_d$ avec $x_d \neq y_d$ et pour tout $i = 1, ..., d-1$,on a $x_i = y_i$.

Exemple 1.1.22 : Soit $E_1 = E_2 = \mathbb{N}$ muni de l'ordre usuel. Alors, avec l'ordre lexicographique $\preceq$ sur $\mathbb{N}^2$, on a les couples $(0, i)$ précèdent les couples $(1, i)$, de même $(1, i)$ précèdent $(2, i)$ et ainsi de suite, pour tout $i \in \mathbb{N}$.

Définition 1.1.23 : Soit $\mathcal{R}$ une relation binaire sur un ensemble E, $e, f \in E$ et $n \in \mathbb{N}$. On appelle $\mathcal{R}$-chemin de longeur n de e à f toute suite $w = (e_0, e_1, ..., e_n) \in E^{n+1}$ de $n+1$ éléments de E tels que $e_0 = e, e_n = f$ et $(e_i, e_{i+1}) \in \mathcal{R}$, pour tout $i = 0, ..., n-1$. Si $e = f$, alors w est un cycle. Un $\mathcal{R}$-chemin dont tous les éléments sont distincts est appelé hamiltonien.

Définition 1.1.24 : Soient E un ensemble non vide et p une propriété des relations vérifiée par $E \times E$. Si l'intersection de toute famille de relations vérifiant P est une relation qui vérifie P, alors il existe pour toute relation $\mathcal{R}$ une plus petite relation vérifiant P et contenant $\mathcal{R}$. On l'appelle la P-fermeture de $\mathcal{R}$. C'est le cas pour les propriétés de réflexivité, de symétrie, de transitivité et toutes les combinaisons de ces propriétés.

Proposition 1.1.25 : Soit p une propriété des relations vérifiéé par $E \times E$. Soit $\mathcal{R}$ une relation binaire sur un ensemble E et soit P une propriété qui peut être vérifiée par $\mathcal{R}$ ou non. On cherche s'il existe une relation $\widetilde{\mathcal{R}}$ possédant la propriété P avec $\widetilde{\mathcal{R}}$ contenant $\mathcal{R}$. On demande de plus que $\widetilde{\mathcal{R}}$ soit minimale, c'est-à-dire, s'il existe une autre relation $\mathcal{S}$ possèdant la propriété P on doit avoir :

$$\mathcal{R} \subseteq \widetilde{\mathcal{R}} \subseteq \mathcal{S},$$

En d'autres mots, la relation $\widetilde{\mathcal{R}}$ est la plus petite relation, au sens de l'inclusion, contenant $\mathcal{R}$ et possédant la propriété P.

• Par exemple, si la propriété P est la réflexivité, la symétrie ou la transitivité, La relation universelle $\omega = E \times E$ possède la propriété P et contenant toute relation $\mathcal{R}$ sur E.

D'autre part, pour toute famille de relations $\mathcal{S}$ de $E \times E$ vérifiant la propriété P, on a bien la relation $\bigcap \mathcal{S}$ vérifie aussi cette propriété. Il en résulte que :

$$\widetilde{\mathcal{R}} = \bigcap_{\substack{\mathcal{R} \subseteq \mathcal{S} \\ \mathcal{S} \text{ vérifie } P}} \mathcal{S}$$

est la plus petite relation binaire contenat $\mathcal{R}$ est possédant la propriété P. On a les formules suivantes :

- Si P est la réflexivité, alors $\widetilde{\mathcal{R}} = \mathcal{R}^r = \mathcal{R} \cup \mathbf{1}_E$.
- Si P est la symétrie, alors $\widetilde{\mathcal{R}} = \mathcal{R}^s = \mathcal{R} \cup \mathcal{R}^{-1}$.
- Si P est la transitivité, alors $\widetilde{\mathcal{R}} = \mathcal{R}^+ = \bigcup_{n=1}^{+\infty} \mathcal{R}^n$.

avec $\mathcal{R}^0 = \mathbf{1}_E, \mathcal{R}^{n+1} = \mathcal{R}^n \circ \mathcal{R}$ pour $n \in \mathbb{N}$.

La relation $\mathcal{R}$ est globalement finie si $\forall e \in E, \mathcal{R}^t(e)$ est fini.

La relation $\mathcal{R}$ est acyclique si $\mathcal{R}^+$ est anti-réflexive.

Remarque 1.1.26 : Dans le cas où l'ensemble E est fini, de cardinal k, l'identité $\mathcal{R}^+ = \bigcup_{n=1}^{+\infty} \mathcal{R}^n$ s'écrit sous la formule plus sympathique suivante : $\mathcal{R}^+ = \bigcup_{n=1}^{n=k} \mathcal{R}^n$.

Exemple 1.1.27 : Soit $E = \{1, 2, 3, 4\}$ et $\mathcal{R} = \{(1,2), (1,3), (2,1), (2,3), (3,4)\}$, on a donc

- $\mathcal{R}^s = \mathcal{R} \cup \mathcal{R}^{-1} = \{(1,2), (1,3), (2,1), (2,3), (3,4), (3,1), (3,2), (4,3)\}$.
- $\mathcal{R}^r = \mathcal{R} \cup \mathbf{1}_E = \{(1,2), (1,3), (2,1), (2,3), (3,4), (1,1), (2,2), (3,3), (4,4)\}$.
- $\mathcal{R}^+ = \{(1,2), (1,3), (2,1), (2,3), (3,4)(1,1), (1,4)(2,2)(2,4)\}$.
- $\mathcal{R}^* = \mathcal{R}^+ \cup \mathcal{R}^r = \{(1,2), (1,3), (2,1), (2,3), (3,4)(1,1), (1,4)(2,2)(2,4), (1,1), (3,3), (4,4)\}$.
- $\widetilde{\mathcal{R}} = \mathcal{R}^* \cup \mathcal{R}^s = \left\{ \begin{array}{c} (1,2), (1,3), (2,1), (2,3), (3,4), (1,1), (2,2)(2,4), (1,4), (3,3), \\ (4,4), (3,1), (3,2), (4,3), (4,2) \end{array} \right\}$.

Proposition 1.1.28 : Soit $\mathcal{R}$ une relation binaire définie sur un ensemble E. On a

1. $(\mathcal{R}^r)^s = (\mathcal{R}^s)^r$.
2. $(\mathcal{R}^r)^t = (\mathcal{R}^t)^r$.
3. $(\mathcal{R}^t)^s \subseteq (\mathcal{R}^s)^t$.

Théorème 1.1.29 : Soit $\mathcal{R}$ une relation binaire définie sur un ensemble E. Il existe une relation d'équivalence $\widetilde{\mathcal{R}}$ telle que :

1. $\mathcal{R} \subseteq \widetilde{\mathcal{R}}$.

2. Si $\mathcal{S}$ est une relation d'équivalence vérifiant $\mathcal{R} \subseteq \mathcal{S} \subseteq \widetilde{\mathcal{R}}$, alors $\mathcal{S} = \widetilde{\mathcal{R}}$.

3. La fermeture d'équivalence de $\mathcal{R}$ est définie par :

$$\widetilde{\mathcal{R}} = \bigcap_{\substack{\mathcal{R} \subseteq \mathcal{S} \\ \mathcal{S} \text{ relation d'équivalence}}} \mathcal{S} = \mathbf{1}_E \bigcup (\bigcup_{n=1}^{+\infty} (\mathcal{R} \cup \mathcal{R}^{-1})^n) = \bigcup_{n=1}^{+\infty} (\mathcal{R} \cup \mathcal{R}^{-1} \cup \mathbf{1}_E)^n.$$

Définition 1.1.30 : Soit $(E, \leq)$ un ensemble ordonné, on appelle minorant (respectivement majorant) d'une partie X de E un élément $e \in E$, tel que $\forall x \in X, e \leq x$ (respectivement $\forall x \in X, x \leq e$.

Un élément m est un élément minimal d'une partie X de E s'il vérifie :

$$(1)\ m \in X,$$
$$(2)\ \forall x \in X, x \leq m \Longrightarrow x = m.$$

On définit de façon analogue un élément maximal.

Un élément m est un plus petit élément ou élément minimum d'une partie X de E s'il vérifie :

$$(1)\ m \in X,$$
$$(2)\ \forall x \in X, m \leq x.$$

On définit de façon analogue un plus grand élément ou élément maximum.

On appelle borne inférieure (respectivement borne supérieure) d'une partie X de E le maximum, s'il existe, des minorants de X noté $\wedge(X)$ (respectivement le minimum des majorants de X, noté $\vee(X)$.

1.2 Monoïdes libres, mots, langages

Définition 1.2.1 : Soit E un ensemble. On appelle loi de composition interne ou opération binaire sur E toute application de $E \times E$ dans E.

Définition 1.2.2 : Lorsqu'un ensemble E est muni d'une opération $*$, on dit que le couple $(E, *)$ forme un magma. Dans le cas où E est fini, on peut consigner la loi $*$ qui le structure dans un tableau dénommé table de Cayley.

Définition 1.2.3 : Soit E un ensemble. Une loi $*$ définie sur E est dite :

- Commutative si : $\forall x, y \in E, x * y = y * x$.
- Associative si : $\forall x, y, z \in E : (x * y) * z = x * (y * z)$.

Selon le cas, le magma $(E, *)$ est alors qualifié de commutatif ou associatif.

Définition 1.2.4 : Soient $(E, *)$ et (F, Δ) deux magmas. Un homomorphisme, ou simplement morphisme, est une application $h : E \longrightarrow F$ telle que :
$\forall x, y \in E : h(x * y) = h(x) \Delta h(y)$. En d'autres termes, un morphisme est une application qui préserve la loi. Quand $E = F$ et $* = \Delta$, on dit h est un endomorphisme.

Un isomorphisme est un morphisme bijectif.

Exemple 1.2.5 : La fonction $\log : \mathbb{R}^+ - \{0\} \longrightarrow \mathbb{R}$ représente un morphisme, et même un isomorphisme de $(\mathbb{R}^+ - \{0\}, \div)$ sur $(\mathbb{R}, -)$, puisque, outre le fait qu'elle soit bijective, on a : $\log(a \div b) = \log(a) - \log(b)$.

Définition 1.2.6 : Soit $(E, *)$ un magma. On dit qu'un élément $e \in E$ est idempotent si $e * e = e$. Le magma est qualifié d'impotent lorsque tous ses éléments le sont. On dit qu'un élément $e \in E$ est absorbant si $\forall x \in E, x * e = e * x = e$. On dit qu'un élément $e \in E$ est neutre si $\forall x \in E, x * e = e * x = x$. Soit $(E, *)$ un magma admettant un élément neutre e, on dit qu'un élément $x \in E$ est symétrisable s'il existe $x' \in E$ vérifiant $x * x' = x' * x = e$.

Exemple 1.2.7 :

- $(\mathbb{N} - \{0\}, +)$ est un magma sans idempotents.
- $(\mathbb{N}, +)$ contient exactement un idempotent, l'entier 0.
- $(\mathbb{N}, +)$ contient deux idempotents 0 et 1.
- $(\mathbb{N}, \min)$ est un magma idempotent.

Définition 1.2.8 : On appelle semi groupe un magma associatif. Un sous semi groupe d'un semi groupe S toute partie H de S qui constitue un semi groupe pour la loi induite par celle de S. En fait, H est un sous semi groupe de S si et seulement si H est stable pour la loi de S : $\forall x, y \in H, xy \in H$.

Exemple 1.2.9 : $(\mathbb{N} - \{0\}, +), (\mathbb{N}, \min)$ sont des semi groupes.

Définition 1.2.10 : On appelle monoïde tout semi groupe possédant un élément neutre. ce dernier est généralement noté 1. On note M tout monoïde $(M, *, 1)$.

Exemple 1.2.11 : $(\mathbb{N}, +, 0), (\mathbb{N} \cup \{+\infty\}, \min, +\infty)$ sont des monoïdes.

Remarque 1.2.12 : Un Monoïde $(M, \cdot, 1)$ qui est tel que tout élément de M possède un symétrique est un groupe.

Exemple 1.2.13 : Tout groupe est un monoïde, $(\mathbb{N}, +, 0)$ est un monoïde qui n'est pas un groupe.

Définition 1.2.14 : Soient M un monoïde et $N \subseteq M$. On dit que N est un sous monoïde de M si :

- $1 \in N$, 1 étant l'élément neutre de M,
- $\forall x, y \in N, xy \in N$, N sous semi groupe de M.

Définition 1.2.15 : Soient $(M, \cdot, 1_M)$ et $(N, *, 1_N)$ deux monoïdes. Un morphisme de monoïdes $h : M \longrightarrow N$ est une application qui vérifie : $\forall x, y \in M, h(x \cdot y) = h(x) * h(y)$ et $h(1_M) = 1_N$.

Exemple 1.2.16 : La fonction exponentielle représente un isomorphisme de $(\mathbb{R}, +)$ dans $(\mathbb{R}_+ - \{0, \times\})$. Elle est bijective et vérifie : $\exp(x + y) = \exp(x) \times \exp(y)$ et $\exp(0) = 1$.

Définition 1.2.17 : Soient $(M, \cdot, 1)$ un monoïde et X, Y deux parties de M. On pose $XY = \{xy / x \in X, y \in Y\}$, l'opération sur les parties ainsi définies donne à l'ensemble $\mathcal{P}(M) = 2^M$ une structure de monoïde.

Soit X une partie de M. On définit sa puissance itérée de la manière suivante :

$$\begin{cases} X^0 = \{1\} \\ X^{n+1} = XX^n, n \geq 0 \end{cases}$$

Puis l'on pose $X^* = \bigcup_{n \geq 0} X^n = \{x_1 x_2 ... x_n / n \geq 0, x_i \in X, 1 \leq i \leq n\}$.

Proposition 1.2.18 : X^* représente le sous monoïde de M engendré par X, pour tout $X \subseteq M$.

Exemple 1.2.19 : On a $(\mathbb{N}, +) = \{1\}^*$. Et pour tout élément $n \in \mathbb{N}$, dans le monoïde $(\mathbb{N} \cup \{+\infty\}, \min, +\infty)$, on a $\{n\}^* = \{n, +\infty\}$.

Définitions 1.2.20 : Soit $(M, *, 1)$ un monoïde , une congruence sur $(M, *, 1)$ est une relation d'équivalence $\equiv$ stable par multiplication à droite et à gauche, c'est-a-dire :

$$\forall x, y, z \in M : x \equiv y \Rightarrow x \cdot z \equiv y \cdot z \text{ et } z \cdot x \equiv z \cdot y$$

Définition 1.2.21 : Soit M un monoïde et $\equiv$ une congruence défine sur M. Le quotient $M/\equiv$ est le monoïde des classes de congruence de M pour la relation $\equiv$. La loi de composition de $M/\equiv$ est définie de la manière suivante: $\overline{u} *_{M/\equiv} \overline{v} = \overline{u *_M v}$.

La projection naturelle (la surjection canonique) de M dans $M/\equiv$ est noté P.

Exemple 1.2.22 : Soit le monoïde $(\mathbb{N}, +)$ et soit la relation $\equiv$ définie par $x \equiv y$ si, et seulement si, x et y ont même parité. la relation $\equiv$ est une congruence. Le quotient de $\mathbb{N}$ par cette relation donne un monoïde comprenant deux éléments, notés $\overline{0}$ et $\overline{1}$ correspondant respectivement aux entiers pairs et impairs.

Définition 1.2.23 : Soit $\equiv$ une congruence sur un monoïde M.

Une partie X de M est dite saturée par $\equiv$ si $\forall x \in X : \overline{x} \subseteq X$.

Définition 1.2.24 : Soit $(M, \cdot, 1_M)$ un monoïde. Pour tout couple (x, y) d'éléments de M, le quotient à gauche de x par y noté $y^{-1}x$ est l'ensemble $\{z \in M : y \cdot z = x\}$. Le quotient à gauche d'un sous ensemble de M par y est l'union des quotients des éléments du sous ensemble par y, i.e, si $X \subseteq M$, alors $y^{-1}X = \bigcup_{x \in X} y^{-1}x$.

Définition 1.2.25 : Soit $(M, \cdot, 1_M)$ un monoïde et A un ensemble de générateurs de M. Le graphe de Cayley (gauche) de M par rapport à A est le graphe (M, E), où $E = \{(x, a, y) \in M \times A \times M : y = a \cdot x\}$.

Définition 1.2.26 : Soit X un ensemble et M un monoïde, une application $\cdot$ de $M \times X$ dans X est une action à gauche de M sur X si :

1. $\forall x \in X, 1_M \cdot x = x$
2. $\forall x \in X, \forall m_1, m_2 \in M : (m_1 m_2) \cdot x = m_1 \cdot (m_2 \cdot x)$.

Définition 1.2.27 : Soit Σ un ensemble de symboles, appelé alphabet. Les éléments de Σ sont aussi appelés des lettres.

Soit Σ un alphabet, un mot sur Σ est une suite finie de symbole. Par exemple, 00110 et 110 sont deux mots sur l'alphabet $\{0, 1\}$. La longueur d'un mot w est le nombre de symboles constituant ce mot, on le note $|w|$. Ainsi, $|00110| = 5$ et $|110| = 3$. L'unique mot de longueur 0 est le mot correspondant à la suite vide. Ce mot s'appelle le mot vide et on le note 1, ou bien ϵ. L'ensemble des mots sur Σ est noté Σ^*. Par exemple
$\{0, 1, 2\}^* = \{\epsilon, 0, 1, 2, 00, 01, 02, 11, 12, 20, 21, 22, 000, 001, ...\}$ (ϵ est le mot vide).

• Si σ est une lettre de l'alphabet Σ, pour tout mot $w = a_1 a_2 ... a_k$ de Σ^*, on note par :

$$|w|_\sigma = card\{i \in \{1,2,...,k\} : a_i = \sigma\}.$$

le nombre d'occurrences de la lettre σ dans le mot w et $w(i)$ sa i-ème lettre.

Par exemple $|00110|_0 = 3$ et $|00110|_1 = 2$, $00110\,(1) = 0$, $00110\,(4) = 1$.

Exemple 1.2.28 : Le biologiste intéressé par l'étude de l'ADN utilisera un alphabet à quatre lettres $\{A, C, G, T\}$ pour les quatre constituants des gènes: Adénine, Cytosine, Guanine et Thymine.

Proposition 1.2.29 : Soit Σ un alpabet,

1. l'ensemble Σ^* est infini.

2. l'ensemble Σ^* est dénombrable.

Démonstration 1. l'ensemble Σ^* est infini, en effet on a $\Sigma^* = \bigcup_{n=0}^{+\infty} \Sigma^n = \Sigma^0 \cup \Sigma ... \Sigma^n \cup ....$

2. On montre que Σ^* est dénombrable. Comme Σ est fini, on peut donc numéroter ses éléments, par exemple, si $\Sigma = \{\alpha, \beta, \gamma\}$, alors $n\,(\alpha) = 1, n\,(\beta) = 2, n\,(\gamma) = 3$. Ensuite, soit u un mot de Σ^*, on considère les longueurs $|u|$ premiers nombres premiers, par exemple si $|u| = 5$, on a les 5 premiers nombres premiers sont $p(1) = 2, p(2) = 3, p(3) = 5, p(4) = 7, p(5) = 11$. On forme le nombre $f\,(u) = \prod_{i=1}^{i=|u|} p(i)^{n(u(i))}$, où $u\,(i)$ désigne la ième lettre de u. Par exemple si $u = \alpha\gamma\beta\alpha\alpha$, alors $f\,(u) = \prod_{i=1}^{i=|u|} p(i)^{n(u(i))} = \prod_{i=1}^{i=5} p(i)^{n(u(i))} = 2^1 \times 3^3 \times 5^2 \times 7^1 \times 11^1$. Donc on peut définir une application $f : \Sigma^* \longrightarrow \mathbb{N}, u \longmapsto f\,(u) = \prod_{i=1}^{i=|u|} p(i)^{n(u(i))}$. Par l'unicité de la décomposition d'un entier en facteurs premiers, l'application f est injective. Enfin, comme f est injective et l'ensemble $\mathbb{N}$ est dénombrable, alors Σ^* est dénombrable.

Définition 1.2.30 : La concaténation est l'opération qui associe à deux mots u et v le mot noté $u.v$ ou uv défini par: si $u = \alpha_1\alpha_2...\alpha_n$ et $v = \beta_1\beta_2...\beta_p$, alors $uv = \gamma_1\gamma_2...\gamma_{n+p}$ avec $\gamma_i = \alpha_i$ pour $i = 1, ...n$ et $\gamma_{n+i} = \beta_i$ pour $i = 1, ...p$. Par exemple, la concaténation des mots 00011 et 011 donne le mot 00011011. On vérifie facilement que la concaténation est une opération associative admettant le mot vide comme élément neutre :

$$\forall x, y, z \in \Sigma^* : (xy)z = x(yz).$$
$$\forall x \in \Sigma^* : x\epsilon = \epsilon x = x.$$

Propriété 1.2.31 : Soit Σ un alphabet quelconque le monoïde Σ^* possède les deux propriétés suivantes:

1. tout élément de Σ^* est une suite finie d'éléments de Σ.

2. deux suites distinctes d'éléments de Σ définissent deus éléments distincts de Σ^*.

Définition 1.2.32 : On dit qu'un mot $u \in \Sigma^*$ est facteur de $w \in \Sigma^*$ s'il existe deux mots $f, g \in \Sigma^*$ tel que $w = fug$.

Exemple 1.2.33 : Soit l'alphabet $\Sigma = \{a, b, c\}$, et le mot $w = aabc$, alors ab est un facteur de w, mais ac ne l'est pas.

Propriété 1.2.34 : Soient t, u, v, w quatre mots de monoïde libre Σ^*.

1. Si $tu = vw$ et $|t| \leq |v|$ alors il existe un unique mot z de Σ^* tel que $v = tz$ et $u = zw$.
2. Si $uv = wt$ et $|u| = |w|$ alors $u = w$ et $v = t$.
3. Pour tout $i \in \mathbb{N} - \{0\}$, $(u^i = v^i \Rightarrow u = v)$.
4. Le monoïde libre Σ^* est simplifiable, c'est à dire,

$$4.1\ uv = uw \Rightarrow v = w;$$
$$4.2\ uv = wv \Rightarrow u = w;$$
$$4.1\ uvw = utw \Rightarrow v = t.$$

5. Les propositions suivantes sont équivalentes:

$$5.1\ uv = vu,$$
5.2 Il existe deux entiers n et m non tous deux nuls tels que $u^n = v^m$,
5.3 Il existe un mot z et deux entiers p et q tels que $u = z^p$ et $v = z^q$.

Exemple 1.2.35 : L'application longueur $|.| : \Sigma^* \longrightarrow \mathbb{N}$ est un morphisme de monoïdes entre $(\Sigma^*, .)$ et $(\mathbb{N}, +)$. En effet,

$$\forall u, v \in \Sigma^* : |uv| = |u| + |v| \text{ et } |\epsilon| = 0.$$

Exemple 1.2.36 : Soit $\Sigma = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ un alphabet, $n \in \mathbb{N} \setminus \{0, 1\}$.

La fonction de Parikh $\Psi : \Sigma^* \longrightarrow \mathbb{N}^n,\ \Psi(w) = (|w|_{\alpha_1}, ..., |w|_{\alpha_n})$,

est un morphisme de monoïdes entre $(\Sigma^*, .)$ et $(\mathbb{N}^n, +)$.

Exemple 1.2.37 : Soit $\Sigma = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ un alphabet, $n \in \mathbb{N} \setminus \{0, 1\}$.

Et soit $\lambda : \Sigma \longrightarrow \mathbb{N},\ \alpha_i \longmapsto \lambda(\alpha_i)$. On définit $\widetilde{\lambda} : \Sigma^* \longrightarrow \mathbb{N}$ comme suit:

$$\widetilde{\lambda}(w) = \sum_{i=1}^{i=n} \lambda(\alpha_i) \left|w\right|_{\alpha_i}.$$

$\widetilde{\lambda}$ est un homomorphisme de monoïdes.

Et si $\forall 1 \leq i \leq n,\ \lambda(\alpha_i) = 1$, alors $\widetilde{\lambda} = |.|$ (le morphisme de longueur).

La proposition suivante justifie le fait que le monoïde Σ^* soit appelé monoïde libre.

Cette propriété caractérise le monoide libre engendré par Σ.

Proposition 1.2.38 : Toute fonction $\mu : \Sigma \longrightarrow M$ de Σ dans un monoïde M se prolonge de façon unique en un morphisme de Σ^* dans M.

Démonstration. L'esistence: Posons

$\widetilde{\mu}(\epsilon) = e_M$ *et* $\widetilde{\mu}(\alpha_1\alpha_2...\alpha_n) = \mu(\alpha_1)\mu(\alpha_2)...\mu(\alpha_n),\ n \in \mathbb{N},\ \alpha_i \in \Sigma,\ 1 \leq i \leq n.$

Et facile de voir que $\widetilde{\mu}$ est bien un homomorphisme

Unicité: Soient $\widetilde{\mu}$ et $\widetilde{\lambda}$ deux homomorphismes de A^* dans M tels que:

$\forall \alpha \in A, \widetilde{\mu}(\alpha) = \widetilde{\lambda}(\alpha)$

Alors $\widetilde{\mu}(1) = \widetilde{\lambda}(1) = e_M$ et pour tout mot $w = \alpha_1\alpha_2...\alpha_n$

On a $\widetilde{\mu}(w) = \widetilde{\mu}(\alpha_1\alpha_2...\alpha_n) = \mu(\alpha_1)\mu(\alpha_2)...\mu(\alpha_n) = \widetilde{\lambda}(\alpha_1\alpha_2...\alpha_n) = \widetilde{\lambda}(w).$ $\square$

Définition 1.2.39 : Un morphisme entre deux monoïdes libres Σ^* et Δ^* est une application $\psi : \Sigma^* \longrightarrow \Delta^*$ qui satisfait :

$$\psi(xy) = \psi(x)\psi(y) \quad \forall x, y \in \Sigma^*.$$

Notons que cet morphisme ψ est complètement déterminé ayant les images des lettres de Σ dans Δ^*, i. e, $\psi(\sigma)$ pour tout σ appartenant à Σ. Nous dirons que le morphisme ψ est non trivial s'il existe au moins une lettre $\sigma \in \Sigma$ pour laquelle $\psi(\sigma) \neq \epsilon$.

En fait remarquer que la propriété $\psi(xy) = \psi(x)\psi(y)$ implique $\psi(\epsilon) = \epsilon$.

Définition 1.2.40 : Une relation binaire $\mathcal{R}$ sur un ensemble Σ^* est noethérienne s'il n'existe pas une chaine infinie d'éléments de Σ^* en relation par $\mathcal{R}$, en d'autres mots il n'existe pas une suite infinie $(w_n)_{n\in\mathbb{N}}$ d'éléments de Σ^* tel que, pour tout entier naturel n, on a $w_n \mathcal{R} w_{n+1}$.

Un ordre sur un ensemble Σ^* est bien fondé s'il ne contient pas de suite infini d'éléments de Σ^* strictement décroissante.

La partie strict d'une relation d'ordre bien fondé $\geq$ noté $>$ et définie par: $x > y$ si $x \geq y$ et $x \neq y$.

Exemples 1.2.41 : La relation $\mathcal{R}_1$ définie sur $\mathbb{N}$ par $x\mathcal{R}_1 y \Longleftrightarrow y = x + 1$ est bien fondé.

La relation $\mathcal{R}_2$ définie sur $\mathbb{N}$ par $x\mathcal{R}_2 y \Longleftrightarrow x$ divise y et $x \neq y$ est bien fondé.

La relation usielle $\geq$ est un ordre bien fondé sur l'ensemble des entiers $\mathbb{N}$.

Proposition 1.2.42 : Soit $h : \Sigma^* \longrightarrow \Gamma^*$ un morphisme, la congruence associée à h, notée $\equiv_h$, est définie par :

pour tous $u, v \in \Sigma^*, u \equiv_h v \Longleftrightarrow h(u) = h(v)$.

Soit $\mathcal{R}$ une relation sur Σ^* qui vérifie $h(r) = h(s)$ pour tout $(r, s) \in \mathcal{R}$, alors il existe un unique morphisme $\psi : \Sigma^* / \overset{*}{\underset{\mathcal{R}}{\leftrightarrow}} \longrightarrow \Gamma^*$ tel que $\psi \circ p = h$ où $\overset{*}{\underset{\mathcal{R}}{\leftrightarrow}}$ est la congruence engendré par $\mathcal{R}$ et p est la surjection canonique.

Soient P et $P^{'}$ deux partitions de monoïde libre Σ^*, on dit que P est plus fine que $P^{'}$ si: $\forall p \in P, \exists p' \in P'$ tel que $p \subseteq p'$. Dans ce cas on dit que P est plus fine que P' ou bien P' est plus grossière que P.

Définition 1.2.43 : Un langage sur un alphabet Σ est simplement un ensemble (fini ou infini) de mots sur Σ. En d'autres termes, un langage est une partie de Σ^*. On distingue en particulier le langage vide $\emptyset$ qui ne contient aucun mot.

Définition 1.2.44 : Soient $L, M \subseteq \Sigma^*$, deux langages. La concaténation des langages L et M est le langage,

$$LM = \{uv, u \in L, v \in M\}$$

En particulier, on peut définir la puissance $n - ième$ d'un langage L, $n > 0$, par:

$$L^n = \{w_1 w_2 ... w_n, \forall i \in \{1, 2...n\}\, w_i \in L\}.$$

Et on pose $L^0 = \{\epsilon\}$.

Exemple 1.2.45 : Soient les deux langages $L = \{u \in \Sigma^* : |u| \text{ est paire}\}$ *et* $K = \{u \in \Sigma^* : |u| \text{ est impa}$

On a alors les égalités suivantes:

$LK = KL = K$.

$LL = L$.

$KK = L \setminus \{\epsilon\}$.

Définition 1.2.46 : Soit L est un langage sur un alphabet Σ. La congruence syntaxique de L notée $\equiv_L$ est définie par,

pour tous $u, v \in \Sigma^*, (u \equiv_L v) \Longleftrightarrow (\forall x, y \in \Sigma^*, xuy \in L \Longleftrightarrow xvy \in L)$.

Définition 1.2.47 : Soit Σ un alphabet. La famille des langages rationnels, notée $Rat(\Sigma^*)$ est la plus petite famille de langages de Σ^* vérifiant les conditions suivantes :

1. $\emptyset \in Rat(\Sigma^*)$,

2. $\forall \sigma \in \Sigma, \{\sigma\} \in Rat(\Sigma^*)$,

3. $Rat(\Sigma^*)$ est fermée (stable) par union et produits finis, c'est à dire : $\forall L_1, L_2 \in Rat(\Sigma^*), L_1 \cup L_2$ et $L_1 L_2$ sont aussi dans $Rat(\Sigma^*)$,

4. $\forall L \in Rat(\Sigma^*), L^* \in Rat(\Sigma^*)$, fermeture par étoile.

Les trois opérations union, produit et étoile, qui interviennent dans la définition, sont qualifiées d'opérations rationnelles.

Remarque 1.2.48 : La réunion de deux langages est très souvent notée additivement : on écrit $L + K$ pour $L \cup K$.

Exemple 1.2.49 :

- $\{\epsilon\}$ est un langage rationnel, car on a $\{\epsilon\} = \emptyset^*$.
- Σ est rationnel : $\Sigma = \bigcup_{\sigma \in \Sigma} \{\sigma\}$.
- Σ^* est rationnel.
- $\{\alpha\beta\}^*$ est rationnel. En effet : $\{\alpha\beta\}^* = (\{\alpha\}\{\beta\})^*$.
- Si $w \in \Sigma^*$, $\{w\}$ est rationnel. Si $w = \sigma_1 ... \sigma_n$, on a $\{w\} = \{\sigma_1\} ... \{\sigma_n\}$.
- Tout langage fini est rationnel. En effet, si L est fini, on a $L = \bigcup_{w \in L} \{w\}$.

Proposition 1.2.50 : Soit $h : \Sigma^* \longrightarrow \Gamma^*$ un morphisme de monoïdes libres. On a :

1. $\forall L \in Rat(\Sigma^*), h(L) \in Rat(\Gamma^*)$.

2. Si de plus h est surjectif, alors $\forall K \in Rat(\Gamma^*), \exists L \in Rat(\Sigma^*), K = h(L)$.

Définition 1.2.51 : Soit Σ un alphabet. Considérons l'alphabet $\widetilde{\Sigma} = \Sigma \cup \{\emptyset, \epsilon, +, *, (,)\}$. On définit l'ensemble des expressions rationnelles sur Σ comme étant le plus petit langage de $\left(\widetilde{\Sigma}\right)^*$ satisfaisant les conditions suivantes :

1. $\emptyset, \epsilon$ sont des expressions rationnelles.

2. $\forall \sigma \in \Sigma, \sigma$ est une expression rationnelle.

3. Si φ et ψ sont des expressions rationnelles, $(\varphi + \psi)$ et $(\varphi\psi)$ le sont également.

4. Si φ est une expression rationnelle, φ^* aussi.

Définition 1.2.52 : On définit une application de l'ensemble des expressions rationnelles sur Σ dans celui des langages de Σ^*, $\gamma \longmapsto L(\gamma)$ de la manière suivante :

1. $L(\emptyset) = \emptyset$ et $L(\epsilon) = \{\epsilon\}$,
2. $\forall \sigma \in \Sigma, L(\sigma) = \{\sigma\}$,
3 $L(\alpha + \beta) = L(\alpha) \cup L(\beta) = L(\alpha) + L(\beta)$ et $L(\alpha\beta) = L(\alpha) L(\beta)$,
4. $L(\alpha^*) = L(\alpha)^*$.

On dit que l'expression γ représente ou dénote le langage $L(\gamma)$.

Remarque 1.2.53 :

• Dans l'écriture des expressions rationnelles, on omet très souvent les parenthèses inutiles pour la compréhension : on écrit par exemple $\alpha\beta$ au lieu de $(\alpha\beta)$.

• A une expression rationnelle correspond toujours un langage unique (L est une application). Mais L n'est ni injective ni surjective : il existe des langages qui ne peuvent pas être représentés par une expression rationnelle. De plus, un même langage peut admettre plusieurs d'expressions rationnelles.

Exemples 1.2.54 :

• Le langage $\{\alpha^n \beta^n / n > 0\}$ sur $\Sigma = \{\alpha, \beta\}$ n'a pas d'expression rationnelle.

• Les deux expressions α^* et $(\alpha^*)^*$ dénote le même langage $\{\alpha\}^*$.

Théorème 1.2.55 : Soit Σ un alphabet. Un langage L sur Σ est rationnel si et seulement s'il existe une expression rationnelles sur Σ dénotant L.

Proposition 1.2.56 : Soient R, S et T des expressions rationnelles. On a :

1. $R + S = S + R$, $R + R = R = R + \emptyset = \emptyset + R$, $(R + S) + T = R + (S + T)$.
2. $R\epsilon = \epsilon R = R$, $R\emptyset = \emptyset R = \emptyset$, $(RS) T = R(ST)$, $RS \neq SR$ (en général).
3. $R(S + T) = RS + RT$, $(S + T) R = SR + TR$.
4. $R^* = R^* R^* = (R^*)^* = (\epsilon + R)^*$, $\emptyset^* = \epsilon^* = \epsilon$, $R^* = \epsilon + RR^* = \epsilon + R^* R = \epsilon + R^+$.
5. $(R + S)^* = (R^* + S^*)^* = (R^* S^*)^* = (R^* S)^* R^* = R^* (SR^*)^*$, $(R + S)^* \neq R^* + S^*$ (en général).
6. $R(SR)^* = (RS)^* R$, $(R^* S)^* = \epsilon + (R + S)^* S$, $(RS^*)^* = \epsilon + R(R + S)^*$.

Proposition 1.2.57 : (Régle d'Arden) Soient L, S, T trois langages sur un alphabet Σ. Si $\epsilon \notin S$, on a : $L = SL + T \Longleftrightarrow L = S^* T$.

Démonstration : • S^*T est bien solution : $S(S^*T)+T = S^+T+T = (S^+ + \epsilon)T = S^*T$.

• S^*T est une solution minimale, i.e, Si L est une solution, alors $S^*T \subseteq L$: la démonstration par récurrence sur la hauteur d'étoile, si $i=0, S^0T = T \subseteq L$ car $L = SL+T$, hyp. réc. pour $i=n : S^nT \subseteq L$, pour $i=n+1, S^{n+1}T = SS^nT \subseteq SL \subseteq SL+T = L$.

• Si $\epsilon \notin S$, alors S^*T est l'unique solution : on suppose la non unicité de la solution S^*T, soit X une autre solution et soit un mot w de longueur minimale tel que $w \in X - S^*T$, on a $w \in X = SX+T$ et $w \notin T$ donc $w = uv$ avec $u \in S$ et $v \in X$. Or $v \notin S^*T$ (sinon w aussi) donc $v \in X - S^*T$. Contradiction, la longueur de w était supposée minimale dans $X - S^*T$.

• Si $\epsilon \in S$, alors pour tout $Y \subseteq \Sigma^*$, $L = S^*T + S^*Y$ est aussi solution :

$S(S^*T+S^*Y)+T = S^+T+S^+Y+T = S^*T+S^*Y$.

1.3 Ensembles définis inductivement

On peut généraliser le principe des preuves par récurence à des ensembles autres que $\mathbb{N}$. Il suffit ces ensembles soient inductifs, c'est-à-dire des ensembles pour lesquels on a un moyen de construction (des éléments de "base" et des "constructeurs") : Soit E un ensemble. On définit inductivement un sous ensemble X de E lorsque l'on se donne des règles de construction des éléments de X, règles que l'on sépare en deux types de règles :

(i) Les de bases : qui indiquent les éléments qui sont dans X.

(ii) Les règles inductives : qui donnent un moyen de construire les éléments de X à partir de ceux déja construits.

Définition 1.3.1 : (Raisonnement par récurence d'ordre k) Soit P une propriété définie sur $\mathbb{N}$,

$$\text{Si } \begin{cases} \bigwedge_{i=0}^{k-1} P(i) \\ \forall n \in \mathbb{N}, \left(\bigwedge_{i=0}^{k-1} P(n-i)\right) \Longrightarrow P(n+1) \end{cases} \text{ alors } \forall n \in \mathbb{N}, P(n).$$

Définition 1.3.2 : (Raisonnement par récurence forte) Soit P une propriété définie sur $\mathbb{N}$,

$$\text{Si } \begin{cases} P(0) \\ \forall n \in \mathbb{N}, (\forall k \leq n, P(k)) \Longrightarrow P(n+1) \end{cases} \text{ alors } \forall n \in \mathbb{N}, P(n).$$

Définition 1.3.3 : Soit E un ensemble, une définition inductive d'une partie X de E est la donnée :

(i) d'une partie B de E

(ii) d'un ensemble O_p d'opérations sur les éléments de E.

X est alors le plus petit ensemble vérifiant :

(i) **base :** tous les éléments de B appartiennent à X $(B \subset X)$.

(ii) **induction** : pour toute opération Φ de O_p d'arité $m \in \mathbb{N}$ et $x_1, ..., x_m \in X$, on a $\Phi(x_1, ..., x_m) \in X$.

Exemples 1.3.4 : On considère l'ensemble E des suites finies, non vides, de 0 et de 1. $E = \{0, 1, 00, 01, 10, 11, 000, 001, ...\}$. On définit l'ensemble X_d par :

(i) 1 est dans X_d.

(ii) si x est dans X_d alors la suite x suivie de 0 est dans X_d.

X_d peut être défini inductivement par :

(i) base : $B = \{1\}$.

(ii) induction : $O_p = \{AjouteZero : x \longmapsto x0\}$.

Exemples 1.3.5 : L'ensemble des expressions logique est le plus petit ensemble noté $\mathcal{E}$ tel que :

(i) base : vrai et faux $\in \mathcal{E}$.

(ii) induction : $\forall e_1, e_2 \in \mathcal{E} : \begin{cases} e_1 \wedge e_2 \in \mathcal{E} \\ e_1 \vee e_2 \in \mathcal{E} \\ \neg e_1 \in \mathcal{E} \end{cases}$.

Théorème 1.3.6 : Tout élément d'un ensemble défini inductivement peut s'obtenir à partir d'éléments de base en appliquant un nombre fini détapes inductives.

Démonstration : Il suffit de considérer la suite d'ensembles suivants :

(i) $X_0 = B$

(ii) $X_{n+1} = X_n \cup \{\Phi(x_1, ..., x_k) : x_1, ..., x_k \in X_n \text{ et } \Phi \in O_p\}$

On montre alors que $X = \bigcup_{n \geq 0} X_n$. Pour tout élément x de X, il existe $n \in \mathbb{N}$ tel que $x \in X_n$. Le plus petit des n tels que $x \in X_n$ donne le nombre d'étapes inductives qu'il faut appliquer aux éléments de base pour obtenir x.

Définition 1.3.6 : Soit X un ensemble défini inductivement à partir de (B, O_p). Définir une fonction $f : X \longrightarrow F$ inductivement consiste à :

(i) base : définir les valeurs $f(x) \in F$ pour tout $x \in B$.

(ii) induction : pour toute opération $\Phi \in O_p$ d'arité n, pour tout $x_1, ..., x_n \in X$, exprimer $f(\Phi(x_1, ..., x_n))$ en fonction de $f(x_1), ..., f(x_n)$.

Exemples 1.3.7 : On considère l'ensemble $\mathcal{E}$ de l'exemple **1.3.5.** On définit la fonction $\text{eval} : \mathcal{E} \longrightarrow \{0, 1\}$ de la façon suivante :

(i) base : $\text{eval}(vrai) = 1, \text{eval}(faux) = 1$

(ii) induction : $\forall e, e_1, e_2 \in \mathcal{E} :$ $\begin{cases} \text{eval}(e_1 \vee e_2) = \max(\text{eval}(e_1), \text{eval}(e_2)) \\ \text{eval}(e_1 \wedge e_2) = \min(\text{eval}(e_1), \text{eval}(e_2)) \\ \text{eval}(\neg e) = 1 - \text{eval}(e) \end{cases}$

Théorème 1.3.8 : Soit X un ensemble défini inductivement à partir d'une base B et d'un ensemble d'opérateurs O_p. On veut démontrer une propriété P vraie pour tout élément de X, auterment dit : $\forall x \in X, P(x)$. Si les deux conditions sont vérifiées :

(i) base : $P(x)$ est vraie pour tout x de B.

(ii) induction : pour tout Φ de O_p d'arité k, pour tout $x_1, ..., x_k \in X$, si $P(x_1), ..., P(x_k)$ sont vraies, alors $P(\Phi(x_1, ..., x_k))$ est vraie. Alors P est vraie pour tout élément de X.

Exemples 1.3.9 : La preuve par récurrence sur $\mathbb{N}$ est une preuve par induction :

(i) base : $B = \{0\}$

(ii) induction : $O_p = \{succ\}$ où $succ : n \longmapsto n + 1$

1.4 Exercices

Exercice 1.4.1 :

- Soient n et p deux entiers naturels.

1. Démontrer que s'il existe une injection de $\{1, 2, ..., n\}$ dans $\{1, 2, ..., p\}$ alors $n \leq p$.

2. Démontrer que s'il existe une bijection d'un ensemble E dans $\{1, 2, ..., n\}$ et une bijection de E dans $\{1, 2, ..., p\}$ alors $n = p$.

- Soient E, F deux ensembles, montrer que:

1. S'il existe une application de E dans E qui est surjective sans être injective, ou injective sans être surjective, alors E est infini.

2. S'il existe une application de E dans F qui est injective et si E est infini, alors F est infini.

3. En déduire que $\mathbb{Z}, \mathbb{Q}, \mathbb{R}, \mathbb{C}$ sont infinis.

4. Un ensemble F est infini si et seulement, si s'il existe une injection de $\mathbb{N}$ dans F.

- L'objectif de ce chapitre est de prouver le théorème de Cantore suivant:

Soit E un ensemble, alors il n'existe pas de bijection de E sur $p(E)$.

La démonstration par l'absurde: soit f une bijection de E sur $p(E)$. Construisons la partie A de E définie par:

$$A = \{x \in E / x \notin f(x)\}$$

1. Montrer l'existence d'un élément e de E qui vérifie $e \notin A \Longleftrightarrow e \in f(e) \Longleftrightarrow e \in A$

2. Soit E un ensemble infini quelconque, montrer que l'infini de $P(E)$ est strictement plus grand que celui de E.

    - Soit F un enseble infini et f une application injective de $\mathbb{N}$ dans F, montrer que l'application h définie de la façon suivante:

$$h : F \longrightarrow F \setminus \{f(o)\}$$
$$\begin{cases} x & \text{si} \quad x \notin f(\mathbb{N}) \\ f(n+1) & \text{si } x = f(n) \in f(\mathbb{N}) \end{cases}$$

est bijective.

- Montrer que s'il existe une injection d'un ensemble E dans $\mathbb{N}$ alors E est fini ou dénombrable.

- Montrer que si E est dénobrable et s'il existe une injection de F dans E alors F est fini ou dénombrable

- Soit $f : \mathbb{N} \times \mathbb{N} \longrightarrow \mathbb{N}; f(n,m) = 2^n 3^m$, montrer que f est injective et en déduire que $\mathbb{N} \times \mathbb{N}$ est dénombrable.

- Soient E et F deux ensembles dénombrables, montrer que $E \times F$, $E \cup F$ sont dénombrables.

- En déduire que $\mathbb{N}^*, \mathbb{Z}, \mathbb{Q}$ sont dénombrables.

- Etudier la bijectivité des applications suivantes: $h_1 : \mathbb{N} \setminus \{0\} \longrightarrow \mathbb{N}; n \longmapsto n-1$.

$$h_2 : \mathbb{N} \setminus \{0\} \longrightarrow \mathbb{Z}, \begin{cases} p & \text{si } n = 2p \\ -p & \text{si } n = 2p+1 \end{cases} \quad ; \quad h_3 : \mathbb{N} \longrightarrow \mathbb{Z}, \begin{cases} f(2n) = n \\ f(2n+1) = -n-1 \end{cases}$$

- Soit I un ensemble dénombrable i.e il existe une bijection de $\mathbb{N}$ dans I et soit $A = \biguplus_{i \in I} A_i$ tel que pour tout i de I , A_i est dénombrable i.e pour tout i de I, il existe une bijection g_i de $\mathbb{N}$ dans A_i .

  1. On considère l'application $h : \mathbb{N} \times \mathbb{N} \longrightarrow A; h(n,p) = g_{f(n)}(p)$, Démontrer que h est bijective et en déduire que A est dénombrable.

Exercice 1.4.2 :

- Soit R une relation binaire sur un ensemble E. On note Δ la relation d'égalité sur E.

1. Comparer Δ et R si R est réflexive. Montrer que R^{-1} est reflexive.
2. Montrer que $R \circ R^{-1}$ est réflexive si, et seulement si, l'ensemble de définition de R est E.
3. Montrer que R est transitive si, et seulement si, $R \circ R \subset R$.
4. Montrer que si R est transitive, $R \circ R$ l'est aussi.
5. Montrer que R est antisymétrique si, et seulement si, $R \cap R^{-1} \subset \Delta$.

Exercice 1.4.3 :

- Soit $\mathcal{P}$ l'ensemble des nombres premiers strictements à 2. On considère la relation R entre deux éléments de $\mathcal{P}$ définie par :

$$pRq \Longleftrightarrow \tfrac{p+q}{2} \in \mathcal{P}.$$

1. La relation R est -elle réflexive, symétrique et transitive ?

- Dans $\mathbb{N}^*$, on définie une relation ϕ en posant pour tout $(x, y) \in \mathbb{N}^* \times \mathbb{N}^*$:

$$x\phi y \Longleftrightarrow \exists n \in \mathbb{N}^* : y = x^n.$$

1. Montrer que ϕ est une relation d'ordre partiel sur $\mathbb{N}^*$.

2. On considère dans la suite de l'exercice que $\mathbb{N}^*$ est ordonné par la relation ϕ .

Déterminer le plus grand élément et le plus petit élément de A dans les cas suivantes:

- $A = \{2, 4, 16\}$.

- $A = \{2, 3, 4, 16, 17\}$.

Exercice 1.4.4 :

- Montrer que la relation R définie sur $\mathbb{R}$ par : $x\mathcal{R}y \Longleftrightarrow xe^y = ye^x$ est une relation d'équivalence. Préciser, pour x fixé, le nombre d'éléments de la classe de x modulo R.

- On définit sur $\mathbb{R}$ une relation d'équivalence $\sim$ par : $x \sim y \Longleftrightarrow x - y \in \mathbb{Z}$.

Soit $x \in \mathbb{R}$, on note $E(x)$ la partie entière de x.

• Vérifier que $\forall x \in \mathbb{R} : x \sim x - E(x)$ et $x - E(x) \in [0, 1[$.

• Soient $x, y \in [0, 1[$, montrer que $x = y$ ou $x \nsim y$. En déduire que le cardinal de l'ensemble quotient $\mathbb{R}/\mathbb{Z}$ est infini.

- Montrer que une relation d'équivalence $\mathcal{R}$ sur un ensemble E peut être définie des manières suivantes :

1. Par la donnée d'un partition $\mathcal{P}$ de E :

$$x\mathcal{R}y \Longleftrightarrow \exists X \in \mathcal{P} \text{ tel que: } x \in X \text{ et } y \in X.$$

2. Par la donnée d'une application f de E dans un ensemble quelconque F :

$$x\mathcal{R}y \Longleftrightarrow f(x) = f(y).$$

3. Par la donnée d'une application h de $E \times E$ dans l'ensemble $U = \{z \in \mathbb{C} : |z| = 1\}$ vérifiant la condition :

$$\forall (x, y, z) \in E^3, h(x, y)h(y, z) = h(x, z).$$

en posant,

$$x\mathcal{R}y \Longleftrightarrow h(x, y) = 1.$$

- Montrer qui'il existe une application $f^* : E/R \longrightarrow F$ telle que $f = f^* \circ \pi$ où π est la projection (surjection) canonique.

- On dit qu'une relation R de E dans E est circulaire si $(x\mathcal{R}y) \wedge (y\mathcal{R}z) \Longrightarrow (x\mathcal{R}z)$ pour tous $x, y, z \in E$. Montrer qu'une relation est réflixive et circulaire si, et seulement si elle est une relation d'équivalence.

- Trouver l'erreur dans le raisonnement suivant :

Soit R une relation binaire de E dans E, symétrique et transitive. Donc, $x\mathcal{R}y$ implique que $y\mathcal{R}x$, par symétrieet comme $(x\mathcal{R}y) \wedge (y\mathcal{R}x)$, cela induit que $x\mathcal{R}x$, par transitivité. Donc R est réflexive et est par conséquent une relation d'équivalence. Donner un exemple R de relation binaire de E dans E, symétrique et transitive, mais non réflexive.

Exercice 1.4.5 :

- Soit M le monoïde donné par la table ci-contre

$\cdot$	1	x	y	t	0
1	1	x	y	t	0
x	x	x	0	0	0
y	y	t	y	t	0
t	t	t	0	0	0
0	0	0	0	0	0

1. Montrer que $A = \{x, y\}$ est un ensemble générateur de M.
2. Déterminer les quotients à gauche $y^{-1}x, y^{-1}t, x^{-1}0, t^{-1}\{0, x, t\}$.
3. Tracer Le graphe de Cayley (gauche) de M par rapport à A.

Exercice 1.4.6 :

- Soient M, N et L trois monoïdes, montrer que $(M \times N) \times L \cong M \times (N \times L)$.
- Soient M et N deux monoïdes et soit θ un morphisme de N dans $End(M)$. Montrer que, l'ensemble $M \times N$ muni de la loi interne $(m, n) \circledast (m', n') = (m\theta(n)(m'), nn')$ est un monoïde.
- Soient M et N deux monoïdes. On note par N^M l'ensemble de toutes les applications de M dans N. Montrer que, l'ensemble N^M muni de la loi interne $(\varphi\psi)(m) = (\varphi)(m)(\psi)(m)$, pour $m \in M$, est un monoïde.
- On considère l'application h de $M \times N^M$ dans N^M définie par : pour tous $m \in M, f \in N^M, h(m, f) = m \cdot f = f^m$ où pour tout $x \in M, f^m(x) = f(xm)$. Montrer que :

1. $\forall f \in N^M : 1_M \cdot f = f$.
2. $\forall f \in N^M, \forall m, m' \in M : (mm') \cdot f = m \cdot (m' \cdot f)$.
3. $\forall f, g \in N^M, \forall m \in M : (fg)^m = f^m g^m$.
4. Montrer que l'ensemble $M \times N^M$ muni de la loi interne $(m, f)(m', g) = (mm', fg^m)$ est un monoïde.

5. Montrer que l'ensemble $M^N \times N$ muni de la loi interne $(f,n)(g,n') = (fg, nn')$ est un monoïde.

Exercice 1.4.7 :

- Soit X une partie de A^*, on note

$$X^* = \{w = x_1.\ x_2\ ...x_n\ , n \in \mathbb{N}\ et\ \forall 1 \preceq i \preceq n,\ x_i \in X\} \cup \{\epsilon\}.$$

Monter que X^* est le sous monoïde de A^* engendré par X.

- Soit le monoïde $M = \{1_M, \alpha, \beta\}$, avec $\alpha^2 = \alpha\beta = \alpha$ et $\beta^2 = \beta\alpha = \beta$.

Construire la table de $(M, \cdot)$.

On considère l'application $f : \{a,b\}^* \longrightarrow M$, où $f(\varepsilon) = 1_M, f(w) = \alpha$, si $w \in a\{a,b\}^*, f(w) = \beta$, si $w \in b\{a,b\}^*$.

Montrer que f un morphisme de monoïdes.

- Soit A un alphabet fini et $L = \{a\}, a \in A$. Calculer L^+ et L^*.

- Soient les deux langages $L = \{u \in A^* : |u|$ est paire$\}$ et $K = \{u \in A^* : |u|$ est impaire$\}$.

Calculer $L + K, L.K, K.L, L.L, K.K$.

- Soit L un ensemble stable non vide de A^*, i, e, $L^2 \subset L$ tel que $X = \{u \in A^* : Lu \cap uL \cap L \neq \emptyset\}$.

Montrer que : $u \in X \Longleftrightarrow (uL \cap L \neq \emptyset$ et $Lu \cap L \neq \emptyset)$.

Exercice 1.4.8 :

- Soit $\Sigma = \{\alpha_1, \alpha_2, ..., \alpha_n\}$ un alphabet, $n \in \mathbb{N} \setminus \{0,1\}$. Et soit $\lambda : \Sigma \longrightarrow \mathbb{N},\ \alpha_i \longmapsto \lambda(\alpha_i)$. On définit $\widetilde{\lambda} : \Sigma^* \longrightarrow \mathbb{N}$ comme suit :

$$\widetilde{\lambda}(w) = \sum_{i=1}^{i=n} \lambda(\alpha_i)\, |w|_{\alpha_i}.$$

Montrer que $\widetilde{\lambda}$ est un homomorphisme de monoïdes.

- Soit $h : \Sigma^* \longrightarrow \Gamma^*$ un morphisme, on définit la relation associée à h, notée $\equiv_h$ comme suit :

$$\text{pour tous } u, v \in \Sigma^*, u \equiv_h v \Longleftrightarrow h(u) = h(v).$$

Montrer que $\equiv_h$ est une congruence sur Σ^*.

- Soit $\mathcal{R}$ une relation sur Σ^* et $h : \Sigma^* \longrightarrow \Gamma^*$ un morphisme de monoïdes qui vérifie $h(r) = h(s)$ pour tout $(r, s) \in \mathcal{R}$.

Montrer qu'il existe un unique morphisme $\psi : \Sigma^*/\overset{*}{\underset{\mathcal{R}}{\leftrightarrow}} \longrightarrow \Gamma^*$ tel que $\psi \circ p = h$ où $\overset{*}{\underset{\mathcal{R}}{\leftrightarrow}}$ est la congruence engendré par $\mathcal{R}$ et p est la surjection canonique.

Exercice 1.4.9 :

- Soit E un ensemble et $\mathcal{R}$ une relation réflexive et tansitive sur E.

1. Montrer que la relation $\equiv$ définie sur E par : $(x \equiv y) \Leftrightarrow (x\mathcal{R}y$ et $x\mathcal{R}y)$ est une relation d'équivalence.

2. Montrer que la relation $\leq$ définie sur $E/\equiv$ par : $(\overline{x} \equiv \overline{y}) \Leftrightarrow (x\mathcal{R}y)$ est un ordre.

- Soit E un ensemble, F_0 une partie non vide de E et $\mathcal{R}$ une relation sur E. Considérons la suite $(F_i)_{i\in\mathbb{N}}$ de parties de E définie par :

$$F_{i+1} = F_i \cup \{x \in E/\exists y \in F_i \text{ tel que } x\mathcal{R}y\}.$$

1. Montrer que, s'il existe $i \in \mathbb{N}$ tel que $F_{i+1} = F_i$, alors pour tout $k \in \mathbb{N}$, $F_{i+k} = F_i$.

- Soit $E = \{1, 2, 3, 4\}$ $F_0 = \{1, 3\}$ et $\mathcal{R} = \{(1, 2), (1, 3), (2, 1), (2, 3), (3, 4)\}$.
  Calculer $F = \bigcup_{i\in\mathbb{N}} F_i$.
- Calculer $\mathcal{R}^t$ la fermeture transitive de $\mathcal{R}$.

Exercice 1.4.10 :

- Soient $(S, +, 0)$ et $(M, \cdot, 1)$ deux monoïdes. Soit l'action à gauche de M sur S, $M \times S \longrightarrow S, (m, s) \longmapsto ms$, avec les conditions suivantes : pour tous $s, s_1, s_2 \in S$ et $m, m_1, m_2 \in M$

(1) $m(s_1 + s_2) = ms_1 + ms_2$,

(2) $m_1(m_2 s) = (m_1 m_2)s$,

(3) $1s = s$,

(4) $m0 = 0$.

1. Montrer que,

1. Pour tout $m \in M, \theta_m : S \longrightarrow S, s \longmapsto ms$, $\theta_m \in End(S)$.

2. Montrer que $\theta : (M, \cdot, 1) \longrightarrow (End(S), \circ, id_S), m \longmapsto \theta_m$ est un morphisme de monoïdes.

3. Monter que $S \times M$ muni de la loi suivante $(s, m) * (s', m') = (s + ms', mm')$ est un monoïde.

- Soit $K = \left\{ \begin{pmatrix} 1 & 0 \\ s & m \end{pmatrix}, s \in \mathbb{N}, m \in \mathbb{N} \right\}$, montrer que $(K, \times)$ est un monoïde.

- On considère l'application $h : (\mathbb{N} \times \mathbb{N}, *) \longrightarrow (K, \times), (s, m) \longmapsto \begin{pmatrix} 1 & 0 \\ s & m \end{pmatrix}$, montrer que h est un isomorphisme de monoïdes.

- Soit $(S, \cdot)$ un semigroupe qui vérifie $\forall a \in S, \exists x \in S : axa = a$.

1. Monter que $(xa)^2 = xa$ et $(ax)^2 = ax$.
2. Montrer que $(\forall a \in S, \exists x \in S : axa = a) \Longleftrightarrow (\forall a \in S, \exists a' \in S, aa'a = a$ et $a'aa' = a')$.

- Soient G, H deux groupe et soit $\theta : G \longrightarrow Aut(H)$ un morphisme de groupes, pour tout $g \in G$, l'automorphisme $\theta(g)$ est noté par θ_g. L'ensemble $G \times H$ muni de loi " $\cdot$ " définie comme suit :

$$(g, h) \cdot (g', h') = (gg', h\theta_g(h'))$$

1. Montrer que $(G \times H, \cdot)$ est un groupe.

Exercice 1.4.11 :

- Soit $\mathcal{R}$ une relation binaire sur un ensemble E.

1. Montrer que la fermeture transitive de $\mathcal{R}$ est $\mathcal{R}^t = \bigcup_{n=1}^{+\infty} \mathcal{R}^n$.

- Soient les deux monoïdes $(\mathbb{N}, +, 0)$ et $(\mathbb{N}, \times, 1)$

1. Monter que, pour tout $n \in \mathbb{N}, \theta_n : (\mathbb{N}, +, 0) \longrightarrow (\mathbb{N}, +, 0), x \longmapsto nx$, $\theta_n \in End(\mathbb{N})$.

2. Montrer que $\theta : (\mathbb{N}, \times, 1) \longrightarrow (End(\mathbb{N}), \circ, id_{\mathbb{N}}), n \longmapsto \theta_n$ est un morphisme de monoïdes.

3. Monter que $\mathbb{N} \times \mathbb{N}$ muni de la loi suivante $(m, n) * (m', n') = (m + nm', nn')$ est un monoïde.

- Soit $M = \left\{ \begin{pmatrix} 1 & 0 \\ m & n \end{pmatrix}, m \in \mathbb{N}, n \in \mathbb{N} \right\}$, montrer que $(M, \times)$ est un monoïde.

- On considère l'application $h : (\mathbb{N} \times \mathbb{N}, *) \longrightarrow (M, \times), (m, n) \longmapsto \begin{pmatrix} 1 & 0 \\ m & n \end{pmatrix}$, montrer que h est un isomorphisme de monoïdes.

Exercice 1.4.12 :

- Considérons le morphisme de monoïdes $\psi : \{\alpha, \beta\}^* \longrightarrow (\mathbb{Z}, +)$ défini par :

$$\psi(\alpha) = 1, \psi(\beta) = -1, \psi(\epsilon) = 0.$$

1. Déterminer $\psi(w)$ pour tout w de $\{\alpha, \beta\}^*$.
2. Montrer que ψ est surjectif. Est-ce que ψ est injectif?
3. Calculer $\psi^{-1}(\{0\})$.

Exercice 1.4.13 :

- Soit *Liste* l'ensemble défini inductivement de la façon suivante :

(i) Base : $\forall x \in \mathbb{N}, x \in Liste$

(ii) Induction : $\forall q \in Liste, x \in \mathbb{N} : x :: q \in Liste$

Soit $\lambda \in Liste$, on note $taille(\lambda)$ le nombre d'éléments de λ et $queue(\lambda)$ la queue de λ.

Exemples : $taille(2) = 1, taille(4 :: 1 :: 2) = 3, queue(2) = 2, queue(4 :: 1 :: 2) = 2$.

1. Donnez une définition inductive de *taille*.
2. Donnez une définition inductive de *queue*.

Chapitre 2

Les semi systèmes de réécriture de mots

Introduction

Ce chapitre constitue une présentation générale des semi-systèmes de réécriture sur le monoïde libre Σ^* ainsi que certaines propriétés essentielles qui les consernent. La réécriture est un moyen de calcul utilisé en informatique, en algèbre, en logique mathématique et en linguistique. Il s'agit de transformer des objets syntaxiques (mots, termes, programmes, preuves, graphes,...) en appliquant des règles bien précises Voici quelques exemples classiques d'utilisation de la réécriture :

- Simplifier une expression algébrique (calcul formel)
- Etudier la structure d'un groupe ou d'un monoïde (algèbre combinatoire).

Un semi-système de réécriture est formé d'un ensemble de règles de réécriture, c'est-à-dire de règles de la forme : $\exp_g \rightarrow \exp_d$ qui veut dire on peut remplacer l'éxpréssion gauche ($\exp_g$) par l'éxpréssion droite ($\exp_d$) dans une dérivation. Par exemple, soit le semi-système de réécriture :

$$(1)\ x + 0 \rightarrow x$$
$$(2)\ x + s(y) \rightarrow s(x + y)$$

où $s(y)$ désigne le successeur de y. Le terme $s(0 + s(0))$ sera réécrit de la façon suivante :

$s(0+s(0))$ devient $s(s(0+0))$ par application de la règle (2), puis $s(s(0))$ par application de la règle (1), et plus aucune règle ne s'applique alors. Une règle $\exp_g \rightarrow \exp_d$ s'applique

à un terme t si ce terme contient une instance de $\exp_g$, c'est à dire $t = u \exp_g v \Rightarrow u \exp_d v$ où la relation $\Rightarrow$ désigne la dérivation associée à la règle $\exp_g \rightarrow \exp_d$. Dans l'exemple ci-dessus, les substitutions qui permetent de passer du terme $x + s(y)$ au terme $0 + s(0)$ sont: $x \mapsto 0, y \mapsto 0$ obtenus à partir des règles (1) et (2) en prenant $x = 0$ et $y = 0$.

On remarquera que l'obtention d'une unique forme normale (mot irréductible) est garantie si le semi-système de réécriture est convergent (c'est à dire s'il possède les propriétés de terminaison et de confluence).

Un système de Thue est une version symétrique d'un semi-système de Thue dans laquelle on peut tout aussi bien remplacer le membre droit d'une règle par son membre gauche.

Contenu

2.1. Définitions et propriétés.
2.2. La terminaison d'un semi-système de réécriture de mots.
2.3. La confluence d'un semi-système de réécriture de mots.
2.4. Exercices.

2.1 Définitions et propriétés

Définition 2.1.1 : Un semi-système de réécriture de mots, dit aussi semi-système de Thue, est un couple $(\Sigma, \mathcal{R})$ où Σ un alphabet fini et $\mathcal{R}$ une relation binaire sur le monoïde libre Σ^*. Tout élément (α, β) de $\mathcal{R}$ est appelé règle de réécriture qu'on note $\alpha \longrightarrow \beta$, avec α est sa partie gauche et β sa partie droite. Des règles $\alpha \longrightarrow \beta_1, \alpha \longrightarrow \beta_2, ..., \alpha \longrightarrow \beta_k$ ayant mème partie gauche sont souvent notées $\alpha \longrightarrow \beta_1 \setminus \beta_2 \setminus ... \setminus \beta_k$.

Exemple 2.1.2 : Soient $\Sigma = \{\alpha, \beta, \gamma\}$ et $\mathcal{R} = \{\alpha\beta \longrightarrow \beta\alpha, \beta\alpha \longrightarrow \alpha\beta, \gamma \longrightarrow \epsilon, \epsilon \longrightarrow \gamma\}$, $(\Sigma, \mathcal{R})$ est un système de réécriture.

Définition 2.1.3 : Appliquer une règle quelconque $u \longrightarrow v$ de $\mathcal{R}$ à un mot w contenant le facteur u consiste à remplacer u par v dans w. S'il n'y a aucune règle de $(\Sigma, \mathcal{R})$ applicable à w, alors w est dit irréductible ou sous la forme normale. Etant données deux mots $w_1, w_2 \in \Sigma^*$, on dit que w_2 dérive directement de w_1, et on note $w_1 \underset{\mathcal{R}}{\Rightarrow} w_2$, si et seulement si, il existe une règle $u \longrightarrow v$ de $\mathcal{R}$ et $x, y \in \Sigma^*$ tels que: $w_1 = xuy$ et $w_2 = xvy$. On dit que w_2 dérive de w_1, et on note $w_1 \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w_2$, s'il existe une suite finie de mots $u_0, u_1, ..., u_n$ de Σ^* avec, $u_0 = w_1, u_i \underset{\mathcal{R}}{\Rightarrow} u_{i+1}, \forall 0 \leq i \leq n-1$ et $u_n = w_2$. Notons que la relation $\overset{*}{\underset{\mathcal{R}}{\Rightarrow}}$ est la fermeture réflexive et transitive de $\underset{\mathcal{R}}{\Rightarrow}$.

Exemples 2.1.4 : 1. Soient $\Sigma_1 = \{0, 1\}$ et $\mathcal{R}_1 = \{10 \longrightarrow 01\}$. Soient $w_1, w_2 \in \Sigma_1^*$, si $w_1 \overset{*}{\underset{\mathcal{R}_1}{\Rightarrow}} w_2$, alors w_2 est la permutation des lettres de w_1 où tous les 0 seront avant les 1. Ce semi-système de réécriture trie les mots constitués de même chiffres binaires.

2. Soient $\Sigma_2 = \{0, 1, +\}$ et $\mathcal{R}_2 = \{0 + 0 \longrightarrow 0, 0 + 1 \longrightarrow 1, 1 + 0 \longrightarrow 1, 1 + 1 \longrightarrow 0\}$.

Ce semi-système de réécriture calcul la somme (*modulo* 2) d'une suite d'entiers binaires.

3. Soient $\Sigma_3 = \{0, 1, +, E\}$ et $\mathcal{R}_3 = \{E \longrightarrow 0, E \longrightarrow 1, E \longrightarrow E + E\}$. Pour tout $w \in \{0, 1, +\}^*$, on a $E \overset{*}{\underset{\mathcal{R}_3}{\Rightarrow}} w$ si, et seulement si w est une expression construite avec les 0 et 1 et l'opérateur $+$.

4. Considérons le semi-système de réécriture sur la dérivation symbolique par rapport à x défini par:

$$\mathcal{R}_4 = \left\{ \begin{array}{c} D_x x \longrightarrow 1, D_x a \longrightarrow 0, D_x(y+z) \longrightarrow D_x y + D_x z, D_x(y*z) \longrightarrow z * D_x y + y * D_x z, \\ D_x(y-z) \longrightarrow D_x y - D_x z, D_x(-y) \longrightarrow -D_x y, D_x(y/z) \longrightarrow z * D_x y - y * D_x z)/z^2 \end{array} \right\}.$$

où a étant un symbole de constante.

5. Considérons le système de Thue donné par l'alphabet $\Sigma_5 = \{\alpha, \beta, E\}$ et la relation $\mathcal{R}_5 = \{E \longrightarrow \epsilon, \epsilon \longrightarrow E, E \longrightarrow \alpha\beta E, \alpha\beta E \longrightarrow E, \alpha\beta \longrightarrow \beta\alpha, \beta\alpha \longrightarrow \alpha\beta\}$. Par exemple on a la dérivation infini suivante: $....\underset{\mathcal{R}_5}{\Longleftrightarrow} E \underset{\mathcal{R}_5}{\Longleftrightarrow} \alpha\beta E \underset{\mathcal{R}_5}{\Longleftrightarrow} \beta\alpha E \underset{\mathcal{R}_5}{\Longleftrightarrow} ...$

Définition 2.1.5 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture . Si un mot $u \in \Sigma^*$ peut se réécrire de manière non triviale, on dit qu'il est réductible. Dans le cas contraire, il est dit irréductible. On note alors $IRR(\mathcal{R})$ l'ensemble des éléments irréductibles de Σ^*.

Exemples 2.1.6 : Soient $\Sigma = \{\alpha, \beta\}$ et $\mathcal{R} = \{\alpha\beta \longrightarrow \beta\alpha\}$. L'ensemble des mots irréductibles est

$$IRR(\mathcal{R}) = \{\beta^n \alpha^m : n, m \in \mathbb{N}\}.$$

Les mots irréductibles sont en effet les éléments de Σ^* qui ne contiennent pas $\alpha\beta$.

Notation 2.1.7 : Pour un semi-système de réécriture $(\Sigma, \mathcal{R})$, on note $D(\mathcal{R}) = \{u \in \Sigma^*, \exists v \in \Sigma^* : u\mathcal{R}v\}$

Proposition 2.1.8 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. On a:

$$IRR(\mathcal{R}) = \Sigma^* - \Sigma^* D(\mathcal{R}) \Sigma^*.$$

Démonstration : On va montrer que $\Sigma^* - IRR(\mathcal{R}) = \Sigma^* D(\mathcal{R})\Sigma^*$:

Soit $w \in \Sigma^*$. On a

$w \in \Sigma^* - IRR(\mathcal{R}) \Leftrightarrow$ il existe $(u, v) \in \mathcal{R}$ tel que u soit un sous-mot de w

$\Leftrightarrow$ il existe $u \in D(\mathcal{R})$ tel que $w \in \Sigma^* u \Sigma^* \Leftrightarrow w \in \Sigma^* D(\mathcal{R}) \Sigma^*$. $\square$

Définition 2.1.9 : Soient $S_1 = (\Sigma_1, \mathcal{R}_1)$ et $S_2 = (\Sigma_2, \mathcal{R}_2)$ deux semi-systèmes de réécriture. Un morphisme de semi-systèmes de réécriture de S_1 vers S_2 est une application $\psi : \Sigma_1^* \longrightarrow \Sigma_2^*$ qui faiblement compatible avec les relations de réductions, c'est-à-dire telle que $\psi(\underset{\mathcal{R}_1}{\Rightarrow}) \subseteq \overset{*}{\underset{\mathcal{R}_2}{\Rightarrow}}$, ou encore telle que, pour tous $x, y \in \Sigma_1^*$ vérifiant $x \underset{\mathcal{R}_1}{\Rightarrow} y$, on a $\psi(x) \overset{*}{\underset{\mathcal{R}_2}{\Rightarrow}} \psi(y)$.

Un tel morphisme est dit:

- Non contractant si $\psi(\underset{\mathcal{R}_1}{\Rightarrow}) \subseteq \overset{+}{\underset{\mathcal{R}_2}{\Rightarrow}}$, ce qui équivaut à dire si $x \underset{\mathcal{R}_1}{\Rightarrow} y$, alors $\psi(x) \overset{+}{\underset{\mathcal{R}_2}{\Rightarrow}} \psi(y)$.
- Strict si $\psi(\underset{\mathcal{R}_1}{\Rightarrow}) \subseteq \underset{\mathcal{R}_2}{\Rightarrow}$, ce qui équivaut à dire si $x \underset{\mathcal{R}_1}{\Rightarrow} y$, alors $\psi(x) \underset{\mathcal{R}_2}{\Rightarrow} \psi(y)$.

2.2 La terminaison d'un semi-système de réécriture de mots.

Nous allons présenter ici la propriété de terminaison d'un semi-système de réécriture, si un semi-système de réécriture termine, cela signifie qu'il est impossible, partant d'un certain élément, d'effectuer une infinité d'opérations de dérivation, quel que soit le point de départ, on arrivera, après un nombre fini d'opérations, à un élément sur lequel on ne peut plus agir, c'est-à-dire un élément x tel qu'il n'existe pas de y vérifiant $x \underset{\mathcal{R}}{\Rightarrow} y$.

Définition 2.2.1 : On dit qu'un semi-système de réécriture $(\Sigma, \mathcal{R})$ termine ou qu'il est noethérien s'il n'existe pas de chaine de réécriture infini $w_0 \underset{\mathcal{R}}{\Rightarrow} w_1 ... \underset{\mathcal{R}}{\Rightarrow} w_n \underset{\mathcal{R}}{\Rightarrow} ...$

Exemples 2.2.2 : Le semi-système de réécriture $(\Sigma_1, \mathcal{R}_1)$, avec $\Sigma_1 = \{\alpha, \beta\}$ et la relation $\mathcal{R}_1 = \{\alpha \longrightarrow \alpha\beta\}$, est non noethérien car la dérivation $\alpha \underset{\mathcal{R}_1}{\Rightarrow} \alpha\beta \underset{\mathcal{R}_1}{\Rightarrow} \alpha\beta\beta \underset{\mathcal{R}_1}{\Rightarrow} \alpha\beta\beta\beta \underset{\mathcal{R}_1}{\Rightarrow} ...$ est bien infini dans $(\Sigma_1, \mathcal{R}_1)$. De même le semi-système de réécriture $(\Sigma_2, \mathcal{R}_2)$, avec $\Sigma_2 = \{\alpha, \beta\}$ et la relation $\mathcal{R}_2 = \{\alpha \longrightarrow \beta, \beta \longrightarrow \alpha\}$, est non noethérien car la dérivation $\alpha \underset{\mathcal{R}_2}{\Rightarrow} \beta \underset{\mathcal{R}_2}{\Rightarrow} \alpha \underset{\mathcal{R}_2}{\Rightarrow} \beta \underset{\mathcal{R}_2}{\Rightarrow} ...$ est bien infini dans $(\Sigma_2, \mathcal{R}_2)$. Par contre le semi-système de réécriture $(\Sigma_3, \mathcal{R}_3)$, avec $\Sigma_3 = \{\alpha, \beta\}$ et la relation $\mathcal{R}_3 = \{\alpha\beta \longrightarrow \alpha\}$, est noethérien, car la seule règle $(\alpha\beta, \alpha)$ est applicable à un mot w de Σ_3^* si, et seulement si, $|w|_{\alpha\beta} \neq 0$, et si $|w|_\beta = k$, alors pas plus de k dérivations le processus de substitution termine.

Remarques 2.2.3 : 1. On remarque que si $(\Sigma, \mathcal{R})$ est un semi-système de réécriture tel qu'il existe un $u \in \Sigma$ avec $u\mathcal{R}u$, alors il ne termine pas. En particulier, tout semi-système de réécriture dont la relation est réflexive ne termine pas, c'est le cas de $(\mathbb{N}, \geq)$. En revanche, le système de réécriture $(\mathbb{N}, >)$ termine.

2. La terminaison implique que tout élément possède au moins une forme normale, on dit que $(\Sigma, \mathcal{R})$ est normalisant. La normalisation est une propriété plus faible que la terminaison, même dans le cas où la forme normale de chaque élément est unique, par exemple considérons un ensemble réduit à deux éléments, notés u et v, que l'on munit de la relation $\mathcal{R} = \{u \longrightarrow u, u \longrightarrow v\}$. alors v est l'unique forme normale de u et de v; cependant, comme nous l'avons déjà remarqué, le fait que u vérifie $u\mathcal{R}u$ empêche $\mathcal{R}$ de terminer.

Définition 2.2.4 : Soient $(\Sigma, \mathcal{R})$ un semi-système de réécriture et deux mots $u, v \in \Sigma^*$. S'il existe $w \in \Sigma^*$ tel que $u \underset{\mathcal{R}}{\Rightarrow} w$ et $v \underset{\mathcal{R}}{\Rightarrow} w$, alors on dit que u et v sont joignables. On désignera

par $u{\downarrow}v$ l'ensemble de tous les éléments $w \in \Sigma^*$ tels que $u \underset{\mathcal{R}}{\Rightarrow} w$ et $v \underset{\mathcal{R}}{\Rightarrow} w$. (Ainsi, u et v sont joignables si, et seulement si $u{\downarrow}v \neq \emptyset$.

Exemple 2.2.5 : soient $\Sigma = \{\alpha, \beta\}$ et $\mathcal{R} = \{\alpha\beta \longrightarrow \beta\alpha\}$, $u = \alpha\beta\beta\alpha$ et $v = \beta\alpha\alpha\beta$, on a $\alpha\beta\beta\alpha \underset{\mathcal{R}}{\Rightarrow} \beta\alpha\beta\alpha$ et $\beta\alpha\alpha\beta \underset{\mathcal{R}}{\Rightarrow} \beta\alpha\beta\alpha$, donc u et v sont joignables et $\beta\alpha\beta\alpha \in u{\downarrow}v$.

Théorème 2.2.6 : Soit $(\Sigma_1, \mathcal{R}_1)$ un semi-système de réécriture. Les assertions suivantes sont équivalentes :

1. $(\Sigma_1, \mathcal{R}_1)$ est noethèrien.
2. Il existe un ordre strict $>$ sur Σ_1^* tel que $(\Sigma_1^*, >)$ termine et tel que si $x \underset{\mathcal{R}_1}{\Rightarrow} y$ alors $x > y$.
3. Il existe un autre semi-système de réécriture $(\Sigma_2, \mathcal{R}_2)$ qui termine ainsi qu'un morphisme de systèmes de réécriture non contractant $\psi : (\Sigma_1, \mathcal{R}_1) \longrightarrow (\Sigma_2, \mathcal{R}_2)$.

Démonstration : 1 $\Rightarrow$ 2. On montrer que $\overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}}$ est un ordre strict qui termine sur Σ_1^*.

Pour l'anti réflexivité, on suppose qu'il existe $x \in \Sigma_1^*$ tel que $x \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} x$, par définition, il existe un chemin de longueur l no nulle de x à x dans $(\Sigma_1, \mathcal{R}_1)$, ce qui équivaut à dire, il existe $l \in \mathbb{N} - \{0\}$ tel que $(x, x) \in \left(\underset{\mathcal{R}_1}{\Rightarrow}\right)^l$ en mettant bout à bout une infinité de copies de ce chemin, on obtient un chemin infini (une dérivation infini) dans $(\Sigma_1, \mathcal{R}_1)$, ce qui contredit l'hypothèse.

Pour la transitivité, on suppose que x, y, et z sont trois éléments de Σ_1^* tels que $x \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} y$ et $y \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} z$: il existe donc un chemin de longueur non nulle dans $(\Sigma_1, \mathcal{R}_1)$ de x à y et un autre de y à z; en les recollant, on obtient un chemin de longueur non nulle de x à z dans $(\Sigma_1, \mathcal{R}_1)$, ce qui donne $x \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} z$.

Enfin, pour la terminaison, on suppose qu'il existe une suite $(x_n)_{n\in\mathbb{N}}$ d'éléments de Σ_1^* tels que $x_n \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} x_{n+1}$ pour tout n, il existe donc, pour tout n, un chemin de longueur non nulle de x_n à x_{n+1} dans $(\Sigma_1, \mathcal{R}_1)$, mis bout à bout, tous ces chemins donnent un chemin infini partant de x_0 dans $(\Sigma_1, \mathcal{R}_1)$, ce qui contredit encore l'hypothèse.

Il ne reste plus qu' à montrer que si $x \underset{\mathcal{R}_1}{\Rightarrow} y$ alors $x \overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}} y$. Ce qui est, par définition de la fermeture transitive $\overset{+}{\underset{\mathcal{R}_1}{\Rightarrow}}$ de $\underset{\mathcal{R}_1}{\Rightarrow}$.

2 $\Rightarrow$ 3. On prend $\Sigma_2 = \Sigma_1$ et la relation $\mathcal{R}_2$ est l'ordre strict $>$ et ψ est l'application identique de Σ_1^* et on le note $id_{\Sigma_1^*}$. Par hypothèse, $(\Sigma_1, >)$ termine et ψ est un morphisme strict, donc non contractant, de $(\Sigma_1, \mathcal{R}_1)$ vers $(\Sigma_1, >)$.

2 ⇒ 3. Supposons qu'il existe une suite $(x_n)_{n\in\mathbb{N}}$ d'éléments de Σ_1^* tels que $x_n \underset{\mathcal{R}_1}{\Rightarrow} x_{n+1}$ pour tout n. Alors, le morphisme ψ nous donne une suite $(\psi(x_n))_{n\in\mathbb{N}}$ d'éléments de Σ_2^* tels que $\psi(x_n) \overset{+}{\underset{\mathcal{R}_2}{\Rightarrow}} \psi(x_{n+1})$ pour tout n. Or, comme $(\Sigma_2, \mathcal{R}_2)$ termine, en appliquant **1 ⇒ 2**, on a la terminaison de $(\Sigma_2, \overset{+}{\underset{\mathcal{R}_2}{\Rightarrow}})$, ce qui contredit l'existence d'une telle suite. □

Les corollaires qui suivent sont des conséquences de l'implication **3 ⇒ 1** du théorème 2.2.6

Corollaire 2.2.7 : *Soit* $(\Sigma_1, \mathcal{R}_1)$ *un semi-système de réécriture tel que* $\mathcal{R}_1 = \{\alpha_i \longrightarrow \beta_i, 1 \leq i \leq n, n \in \mathbb{N}^*\}$. *Si* $\forall 1 \leq i \leq n, |\alpha_i| > |\beta_i|$, *alors le semi-système* $(\Sigma_1, \mathcal{R}_1)$ *est noethérien.*

Démonstration : On prend $(\Sigma_2, \mathcal{R}_2) = (\mathbb{N}, >)$, on a $(\mathbb{N}, >)$ termine et soit l'application de longueur

$\psi : (\Sigma_1, \mathcal{R}_1) \longrightarrow (\mathbb{N}, >)$, définie par $w \longmapsto |w|$.

L'application ψ est un morphisme de monoïdes et $\forall w_1, w_1 \in \Sigma_1^*$,

on a $w_1 \underset{\mathcal{R}_1}{\Rightarrow} w_2 \Longleftrightarrow \exists \alpha_i \longrightarrow \beta_i \in \mathcal{R}_1, \exists (x, y) \in \Sigma_1^* \times \Sigma_1^* : w_1 = x\alpha_i y$ et $w_2 = x\beta_i y)$.

On a $\psi(w_1) = \psi(x\alpha_i y) = |x| + |\alpha_i| + |y|$ et $\psi(w_2) = \psi(x\beta_i y) = |x| + |\beta_i| + |y|$,

comme $\forall 1 \leq i \leq n, |\alpha_i| > |\beta_i|$, alors $\psi(w_1) > \psi(w_2)$ donc $\psi(w_1) >^+ \psi(w_2)$. □

Par conséquent $(\Sigma_1, \mathcal{R}_1)$ est noethérien.

Exemple 2 2.8 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture avec $\Sigma = \{\alpha, \beta\}$ et la relation $\mathcal{R} = \{\alpha\alpha \longrightarrow \beta\}$.

On $|\alpha\alpha| = 2, |\beta| = 1$, donc $|\alpha\alpha| > |\beta|$, et par consèquent le semi-système $(\Sigma, \mathcal{R})$ termine.

Remarque 2.2.9 : L'inverse du corollaire 2.3.1 n'est pas vraie, par exemple le semi-système de réécriture $(\Sigma, \mathcal{R})$, avec $\Sigma = \{\alpha, \beta, \gamma\}$ et la relation $\mathcal{R} = \{\alpha\beta \longrightarrow \gamma\alpha\alpha\}$ est noethérien.

Corollaire 2.2.10 : *Soit* $\Sigma = \{\sigma_1, \sigma_2, ..., \sigma_n\}$ *un alphabet,* $n \in \mathbb{N} - \{0, 1\}$. *Etant donnée l'application* $\lambda : \Sigma \longrightarrow \mathbb{N}$, $\sigma_i \longmapsto \lambda(\sigma_i)$. *On définit l'application* $\widetilde{\lambda} : \Sigma^* \longrightarrow \mathbb{N}$ *comme suit:* $\widetilde{\lambda}(w) = \sum_{i=1}^{i=n} \lambda(\sigma_i) \left|w\right|_{\sigma_i}$. *L'application* $\widetilde{\lambda}$ *est bien un morphisme de monoïdes.*

Soit $(\Sigma_1, \mathcal{R}_1)$ *un semi-système de réécriture avec* $\mathcal{R}_1 = \left\{\alpha_j \longrightarrow \beta_j, 1 \leq j \leq m; m \in \mathbb{N}^*\right\}$.

Si pour tout $1 \leq j \leq m : \widetilde{\lambda}(\alpha_j) > \widetilde{\lambda}(\beta_j)$, *alors* $(\Sigma_1, \mathcal{R}_1)$ *est noethérien.*

Démonstration : On prend $(\Sigma_2, \mathcal{R}_2) = (\mathbb{N}, >)$, on a $(\mathbb{N}, >)$ termine. Et soit $\psi : (\Sigma_1, \mathcal{R}_1) \longrightarrow (\Sigma_2, \mathcal{R}_2)$, définie par $\psi(w) = \widetilde{\lambda}(w)$. L'application ψ est bien un morphisme de monoïdes.

On a:$\forall w_1, w_2 \in \Sigma_1^*, w_1 \underset{\mathcal{R}_1}{\Rightarrow} w_2 \iff \exists \alpha_j \longrightarrow \beta_j \in \mathcal{R}_1, \exists (x,y) \in \Sigma_1^* \times \Sigma_1^* : w_1 = x\alpha_j y$ et $w_2 = x\beta_j y$). Donc $\psi(w_1) = \psi(x\alpha_j y) = \widetilde{\lambda}(x) + \widetilde{\lambda}(\alpha_j) + \widetilde{\lambda}(y)$ et $\psi(w_2) = \psi(x\beta_j y) = \widetilde{\lambda}(x) + \widetilde{\lambda}(\beta_j) + \widetilde{\lambda}(y)$, comme $1 \leq j \leq m : \widetilde{\lambda}(\alpha_j) > \widetilde{\lambda}(\beta_j)$, alors $\psi(w_1) > \psi(w_2)$ donc $\psi(w_1) >^+ \psi(w_2)$. Enfin $(\Sigma_1, \mathcal{R}_1)$ est noethérien. □

Exemple 2.2.11 : Soit $\Sigma = \{\alpha, \beta, \gamma\}$ un alphabet, et soit $\lambda : \Sigma \longrightarrow \mathbb{N}$, avec $\lambda(\alpha) = 1, \lambda(\beta) = 2, \lambda(\gamma) = 3$.

Et Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture tel que $\Sigma = \{\alpha, \beta, \gamma\}$ et la relation $\mathcal{R}$ définie par $\{\beta\beta \longrightarrow \alpha\alpha, \gamma\beta \longrightarrow \alpha\beta\}$.

Comme $|\beta\beta| = |\alpha\alpha|$ et $|\gamma\beta| = |\alpha\beta|$ on utilise le Corollaire **2.3.4** au lieu du Corollaire 2.3.1 car les conditions dans ce dernier ne sont pas verifiées.

Vérifions qu'on a : $\widetilde{\lambda}(\beta\beta) > \widetilde{\lambda}(\alpha\alpha)$ et $\widetilde{\lambda}(\gamma\beta) > \widetilde{\lambda}(\alpha\beta)$.

On a $\widetilde{\lambda}(\beta\beta) = \lambda(\alpha)\left|\beta\beta\right|_\alpha + \lambda(\beta)\left|\beta\beta\right|_\beta + \lambda(\gamma)\left|\beta\beta\right|_\gamma = 1 \times 0 + 2 \times 2 + 3 \times 0 = 4$.

De même $\widetilde{\lambda}(\alpha\alpha) = \lambda(\alpha)\left|\alpha\alpha\right|_\alpha + \lambda(\beta)\left|\alpha\alpha\right|_\beta + \lambda(\gamma)\left|\alpha\alpha\right|_\gamma = 1 \times 2 + 2 \times 0 + 3 \times 0 = 2$.

Dautre part on $\widetilde{\lambda}(\gamma\beta) = \lambda(\alpha)\left|\gamma\beta\right|_\alpha + \lambda(\beta)\left|\gamma\beta\right|_\beta + \lambda(\gamma)\left|\gamma\beta\right|_\gamma = 1 \times 0 + 2 \times 1 + 3 \times 1 = 5$.

Et $\widetilde{\lambda}(\alpha\beta) = \lambda(\alpha)\left|\alpha\beta\right|_\alpha + \lambda(\beta)\left|\alpha\beta\right|_\beta + \lambda(\gamma)\left|\alpha\beta\right|_\gamma = 1 \times 1 + 2 \times 1 + 3 \times 0 = 3$.

Finalement $(\Sigma, \mathcal{R})$ est noethérien.

Une autre méthode pour montrer la terminaison d'un système de réécriture de mots $(\Sigma, \mathcal{R})$ utilise une fonction de poids $P : \Sigma^* \longrightarrow T$ où l'ensemble T est muni d'un ordre $\geq$ bien fondé, on démontre alors que cette fonction vérifie la condition suivante:
si $w_1 \underset{\mathcal{R}}{\Rightarrow} w_2$ alors $P(w_1) > P(w_2)$. L'existence d'une telle fonction implique la terminaison d'un système de réécriture et réciproquement. Puisque l'ordre sur les entiers naturels est bien fondé, une condition suffisante de terminaison est donnée par l'existence d'une fonction $P : \Sigma^* \longrightarrow \mathbb{N}$, on fait remarquer que la condition $w_1 \underset{\mathcal{R}}{\Rightarrow} w_2$ alors $P(w_1) > P(w_2)$ est décidable si P est un morphisme de *monoïde*s de $(\Sigma^*, \cdot)$ vers $(\mathbb{N}, +)$ et vérifier pour tout $u \longrightarrow v$ de $\mathcal{R}$: $P(u) > P(v)$.

Proposition 2.2.12 : *Soit* $(\Sigma, \mathcal{R})$ *un semi-système de réécriture de mots,* $\psi : (\Sigma^*, \cdot) \longrightarrow (\mathbb{N}, +)$ un morphisme de *monoïdes non trivial et la fonction* $P : \Sigma^* \longrightarrow \mathbb{N}$ définie par $P(w) = \sum_{i=1}^{i=|w|} n^i \times \psi(w(i))$, $n \in \mathbb{N} - \{0\}$ où $w(i)$ est la i-éme lettre de w.

Si $\forall u \longrightarrow v \in \mathcal{R}, \begin{cases} |u| = |v| & (C_1) \\ \text{et} & \\ P(u) > P(v) & (C_2) \end{cases}$, alors $(\Sigma, \mathcal{R})$ est noethérien.

Démonstration : Tout d'abord on montre qu'on a : $\forall x, y \in \Sigma^* : P(xy) = P(x) + n^{|x|} \times P(y)$.

On a $P(xy) = \sum_{i=1}^{i=|xy|} n^i \times \psi((xy)(i)) = \sum_{i=1}^{i=|x|} n^i \times \psi((xy)(i)) + \sum_{i=|x|+1}^{i=|x|+|y|} n^i \times \psi((xy)(i))$

$= \sum_{i=1}^{i=|x|} n^i \times \psi((x)(i)) + \sum_{i=1}^{i=|y|} n^{|x|+i} \times \psi((xy)(|x|+i))$

$= \sum_{i=1}^{i=|x|} n^i \times \psi((x)(i)) + \sum_{i=1}^{i=|y|} n^{|x|+i} \times \psi((y)(i)) = P(x) + n^{|x|} \times P(y)$.

Soient $u \longrightarrow v \in \mathcal{R}$ et $x, y \in \Sigma^*$, on montre que $P(xuy) > P(xuy)$.

On a $P(xuy) = P(x(uy)) = P(x) + n^{|x|} \times P(u\,y) = P(x) + n^{|x|}\big(P(u) + n^{|u|} \times P(y)\big)$

$= P(x) + n^{|x|} \times P(u) + n^{|x|+|u|} \times P(y)$. D'autre part, $P(xvy) = P(x(vy)) = P(x) + n^{|x|} \times P(v\,y) = P(x) + n^{|x|}\big(P(v) + n^{|v|} \times P(y)\big) = P(x) + n^{|x|} \times P(v) + n^{|x|+|v|} \times P(y)$. D'après les conditions (C_1), (C_2), décrites ci-dessus on a $P(xuy) > P(xvy)$ et par conséquent $(\Sigma, \mathcal{R})$ est noethérien. □

Exemple 2.2.13 : Soit $\Sigma = \{\alpha, \beta, \gamma\}$ un alphabet et la relation $\mathcal{R}$ définie par $\{\beta\alpha \longrightarrow \alpha\beta, \gamma\beta \longrightarrow \beta\gamma\}$.

On considère le morphisme, $\psi : \Sigma^* \longrightarrow \mathbb{N}$, avec $\psi(\alpha) = 3, \psi(\beta) = 2, \psi(\gamma) = 1$ et la fonction de poids $P : \Sigma^* \longrightarrow \mathbb{N}$, où $P(w) = \sum_{i=1}^{i=|w|} 2^i \times \psi(w(i))$.

Pour la condition (C_1), on a $|\beta\alpha| = |\alpha\beta| = 2$ et $|\gamma\beta| = |\beta\gamma| = 2$.

Pour la condition (C_2), on montre que $P(\beta\alpha) > P(\alpha\beta)$ et $P(\gamma\beta) > P(\beta\gamma)$.

On calcul $P(\beta\alpha), P(\beta\alpha) = \sum_{i=1}^{i=2} 2^i \times \psi(\beta\alpha(i)) = 2 \times \psi(\beta) + 2^2 \times \psi(\alpha) = 16$.

De même $P(\alpha\beta) = \sum_{i=1}^{i=2} 2^i \times \psi(\alpha\beta(i)) = 2 \times \psi(\alpha) + 2^2 \times \psi(\beta) = 14$, donc $P(\beta\alpha) > P(\alpha\beta)$.

On calcul $P(\gamma\beta), P(\gamma\beta) = \sum_{i=1}^{i=2} 2^i \times \psi(\gamma\beta(i)) = 2 \times \psi(\gamma) + 2^2 \times \psi(\beta) = 10$.

De même $P(\beta\gamma) = \sum_{i=1}^{i=2} 2^i \times \psi(\beta\gamma(i)) = 2 \times \psi(\beta) + 2^2 \times \psi(\gamma) = 8$, donc $P(\gamma\beta) > P(\beta\gamma)$.

Par conséquent $(\Sigma, \mathcal{R})$ est noethérien.

Proposition 2.2.14 : *Soit* $(\Sigma, \mathcal{R})$ *un semi-système de réécriture de mots,* $\psi : (\Sigma^*, \cdot) \longrightarrow (\mathbb{N}, +)$ un morphisme de *monoïdes non trivial et la fonction* $P : \Sigma^* \longrightarrow \mathbb{N}$ définie par $P(w) = \sum_{i=1}^{i=|w|} i \times \psi(w(i))$, où $w(i)$ est la i-éme lettre de w.

Si $\forall u \longrightarrow v \in \mathcal{R}$, $\begin{cases} |u| = |v| & (C_1) \\ \text{et} & \\ P(u) > P(v) & (C_2) \\ \text{et} & \\ \psi(u) > \psi(u) & (C_3) \end{cases}$, alors $(\Sigma, \mathcal{R})$ est noethérien.

Démonstration : Tout d'abord on montre qu'on a: $\forall x, y \in \Sigma^* : P(xy) = P(x) + P(y) + |x| \times \psi(y)$.

On a $P(xy) = \sum_{i=1}^{i=|xy|} i \times \psi(xy(i)) = \sum_{i=1}^{i=|x|} i \times \psi(xy(i)) + \sum_{i=|x|+1}^{i=|x|+|y|} i \times \psi(xy(i))$

$= \sum_{i=1}^{i=|x|} i \times \psi(x(i)) + \sum_{i=1}^{i=|y|} (|x| + i) \times \psi((xy)(|x| + i))$

$= \sum_{i=1}^{i=|x|} i \times \psi((x)(i)) + \sum_{i=1}^{i=|y|} (|x| + i) \times \psi((y)(i)) = P(x) + P(y) + |x| \times \psi(y)$. $\left(\sum_{i=1}^{i=|y|} \psi((y)(i)) = \psi((y) \right)$.

Soient $u \longrightarrow v \in \mathcal{R}$ et $x, y \in \Sigma^*$, on montre que $P(xuy) > P(xuy)$.

On a $P(xuy) = P(x(uy)) = P(x) + P(u\,\mathrm{y}) + |x| \times \psi(u\,\mathrm{y}) = P(x) + P(u) + P(y) + |u| \times \psi(\mathrm{y}) + |x| \times (\psi(u) + \psi(\mathrm{y})) = [P(x) + P(y) + |x| \times \psi(y)] + [P(u) + |u| \times \psi(\mathrm{y}) + |x| \times \psi(u)]$ .D'autre part, $P(xvy) = [P(x) + P(y) + |x| \times \psi(y)] + [P(v) + |v| \times \psi(\mathrm{y}) + |x| \times \psi(v)]$. D'après les conditions (C_1), (C_2), (C_1) décrites ci-dessus on a $P(xuy) > P(xvy)$ et par conséquent $(\Sigma, \mathcal{R})$ est noethérien. □

Exemple 2.2.15 : Soit $\Sigma = \{\alpha, \beta, \gamma\}$ un alphabet et la relation $\mathcal{R}$ définie par $\{\beta\alpha \longrightarrow \beta\gamma, \alpha\beta \longrightarrow \alpha\gamma\}$

On considère le morphisme, $\psi : \Sigma^* \longrightarrow \mathbb{N}$, avec $\psi(\alpha) = 2, \psi(\beta) = 1, \psi(\gamma) = 0$ et la fonction de poids $P : \Sigma^* \longrightarrow \mathbb{N}$, où $P(w) = \sum_{i=1}^{i=|w|} i \times \psi(w(i))$.

Pour la condition (C_1), on a $|\beta\alpha| = |\beta\gamma| = 2$ et $|\alpha\beta| = |\alpha\gamma| = 2$.

Pour la condition (C_2), on montre que $P(\beta\alpha) > P(\beta\gamma)$ et $P(\alpha\beta) > P(\alpha\gamma)$.

On calcul $P(\beta\alpha), P(\beta\alpha) = \sum_{i=1}^{i=2} i \times \psi(\beta\alpha(i)) = 1 \times \psi(\beta) + 2 \times \psi(\alpha) = 5$.

De même $P(\beta\gamma) = \sum_{i=1}^{i=2} i \times \psi(\beta\gamma(i)) = 1 \times \psi(\beta) + 2 \times \psi(\gamma) = 1$, donc $P(\beta\alpha) > P(\beta\gamma)$.

On calcul $P(\alpha\beta), P(\alpha\beta) = \sum_{i=1}^{i=2} i \times \psi(\alpha\beta(i)) = 1 \times \psi(\alpha) + 2 \times \psi(\beta) = 4$.

De même $P(\alpha\gamma), P(\alpha\gamma) = \sum_{i=1}^{i=2} i \times \psi(\alpha\gamma(i)) = 1 \times \psi(\alpha) + 2 \times \psi(\gamma) = 2.$, donc $P(\alpha\beta) > P(\alpha\gamma)$.

Pour la condition (C_3), on montre que $\psi(\beta\alpha) > \psi(\beta\gamma)$ et $\psi(\alpha\beta) > \psi(\alpha\gamma)$.

On a $\psi(\beta\alpha) = 3, \psi(\beta\gamma) = 1, \psi(\alpha\beta) = 3, \psi(\alpha\gamma) = 2$.

Finalement $(\Sigma, \mathcal{R})$ est noethérien.

Définition 2.2.16 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. On dit que le principe de récurrence est vrai dans $(\Sigma, \mathcal{R})$ si, pour toute propriété P définie sur Σ^*, on a,

si pour tout $x \in \Sigma^*$, le fait que $P(y)$ soit vraie pour tous les $y \in \Sigma^*$ tels que $x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$ implique que $P(x)$ est vraie, alors $P(x)$ est vraie pour tout $x \in \Sigma^*$, en symboles,

$$(\forall x \in \Sigma^*, \forall y \in \Sigma^* : \Big(P(x) \text{ et } x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y\Big) \Rightarrow P(x)) \Rightarrow (\forall x \in P(x)).$$

Exemple 2.2.17 : Dans le cas de $(\mathbb{N}, >)$, il s'agit du principe de récurrence usuelle :

si $P(0)$ est vraie et $(\forall n \in \mathbb{N}, P(m)$ est vraie tels que $n > m) \Rightarrow P(n)$ est vraie, alors $P(n)$ est vraie pour tout n, en symboles,

$$(P(0) \text{ et } (\forall n \in \mathbb{N}, \forall m \in \mathbb{N} : (P(n) \text{ et } n > m) \Rightarrow P(n))) \Rightarrow (\forall n \in P(n)).$$

Dans cet exemple en fait remarquer que comme la relation $>$ est transitive, alors sa fermeteur tansitive est lui-même.

Théorème 2.2.18 : Un semi-système de réécriture $(\Sigma, \mathcal{R})$ est noethèrien si, et seulement si le principe de récurrence est vraie dans $(\Sigma, \mathcal{R})$.

Démonstration : En fait la démonstration de l'implication si le semi-système de réécriture $(\Sigma, \mathcal{R})$ est noethérien alors, le principe de récurrence est vraie dans $(\Sigma, \mathcal{R})$, par l'absurde, supposons que $S = (\Sigma, \mathcal{R})$ est noethérien mais que le principe de récurrence n'est pas vrai dans $(\Sigma, \mathcal{R})$. Alors il existe une propriété P telle que:

• pour tout $x \in \Sigma^*$, le fait que $P(y)$ soit vraie pour tous les $y \in \Sigma^*$ tels que $x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$ implique que $P(x)$ est vraie;

• il existe x_0 dans Σ^* tel que $P(x_0)$ est fausse.

Alors, il existe x_1 dans Σ^* tel que $x_0 \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} x_1$ et $P(x_1)$ est fausse, sinon, on aurait $P(x_0)$. On recommence à partir de x_1, il existe forcément un x_2 dans Σ^* tel que $x_1 \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} x_2$ et $P(x_2)$ est fausse. On peut donc construire un chemin de réduction infini dans $(\Sigma, \mathcal{R})$ qui ne termine pas, ce qui contredit le fait que $(\Sigma, \mathcal{R})$ est noethérien.

Réciproquement**:** supposons que le principe de récurrence est vrai dans $(\Sigma, \mathcal{R})$. On note $P(x)$ la formule propositionnelle:

$P(x)$:(Il n'existe pas de chemin de réduction infini partant de x dans $(\Sigma, \mathcal{R})$).

Soit $x \in \Sigma^*$ et supposons que $P(y)$ est vraie pour tous les $y \in \Sigma^*$ tels que $x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$, en particulier, il n'existe aucun chemin infini partant d'un y tel que $x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$, ce qui exclut la possibilité d'avoir un chemin infini partant de x. Donc $S = (\Sigma, \mathcal{R})$ est noethérien. □

2.3 La confluence d'un semi-système de réécriture de mots.

Dans ce pragraphe, on introduit la propriété de confluence d'un semi-système de réécriture de mots, dans un semi-système de réécriture confluent, les choix effectués n'ont pas d'importance.

Définition 2.3.1 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. Un branchement de $(\Sigma, \mathcal{R})$ est un triplet (x, y, z) d'éléments de Σ^* tels que $x \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} y$ et $x \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} z$, x est appelé la source d'un tel branchement. On dit qu'un branchement (x, y, z) est local si $x \underset{\mathcal{R}}{\Rightarrow} y$ et $x \underset{\mathcal{R}}{\Rightarrow} z$. Un branchement (x, y, z) est dit confluent s'il existe un $w \in \Sigma^*$ tel que $y \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w$ et $z \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w$. On dit d'un tel w qu'il ferme le branchement(x, y, z).

Définition 2.3.2 : On dit qu' un semi-système de réécriture est confluent (resp. localement confluent) si tous ses branchement(resp. branchements locaux) sont confluents. On dit aussi que la relation binaire $\mathcal{R}$ est (localement) confluente.

Exemple 2.3.3 : On considère le semi-système de réécriture $(\Sigma, \mathcal{R})$ où $\Sigma = \{\alpha, \beta\}$, $\mathcal{R} = \{\alpha\beta \longrightarrow \alpha, \beta\alpha \longrightarrow \beta\}$.

Ce système n'est pas confluent, car on a:

$\alpha\beta\alpha \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} \alpha\beta \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} \alpha$ et $\alpha\beta\alpha \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} \alpha\alpha$, mais les mots $\alpha\alpha, \alpha$ sont en formes normaux.

Proposition 2.3.4 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture.

1. Pour tout (x, y) de $\mathcal{R}$, le branchement (x, y, y) est confluent

2. Si (x, y, z) est un un branchement confluent, alors

• pour tout mot u de Σ^*, les deux branchement $(ux, u\,\mathrm{y}, uz)$ et (xu, yu, zu) sont confluents.

• pour tous mots u, v de Σ^*, le branchement $(uxv, u\,\mathrm{y}v, uzv)$ est confluent.

3. Pour tous $(x, y), (z, t)$ de $\mathcal{R}$, le branchement (xz, yz, xt) est confluent. Dans ce cas on dit que les réductions (xz, yz) et (xz, xt) sont disjointes.

Définition 2.3.5 : Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. Un branchement est critique si il n'est pas de la forme $(ux, u\,\mathrm{y}, uz)$ ou (xu, yu, zu) où (x, y, z) un branchement, u est un mot non vide et ses réductions ne sont ni égales ni disjointes.

Proposition 2.3.6 : Un branchement critique dans un semi-système de réécriture $(\Sigma, \mathcal{R})$ est nécessairement d'une de ces deux formes :

• Un chevauchement (uxv, yv, uz) où $(ux, y), (xv, z) \in \mathcal{R}$ et $u, x, v \neq \epsilon$.

• Une inclusion (uxv, uyv, z) où $(x, y), (uxv, z) \in \mathcal{R}$ et $ux, xv \neq \epsilon$.

Proposition 2.3.7 : Si tous les branchements critiques d'un semi-système de réécriture $(\Sigma, \mathcal{R})$ sont confluents, alors tous les branchements le sont.

Démonstration

Soit $(\Sigma, \mathcal{R})$ un système de réécriture dont tous les branchements critiques sont confluents. Un branchement (x, y, z) est forcément dans un des cas suivants:

• si (x, y, z) est critique, alors, par hypothèse, il est confluent.

• si $y = z$, on a vu que le branchement (x, y, y) est confluent.

• si les réductions $x \underset{\mathcal{R}}{\overset{*}{\Rightarrow}} y$ et $x \underset{\mathcal{R}}{\overset{*}{\Rightarrow}} z$ sont disjointes, on a vu que le branchement (x, y, z) est confluent.

• Si $(x, y, y) = (ux', uy', uz')$ ou $(x, y, z) = (x'u, y'u, z'u)$, où (x', y', z') est un branchement et u un mot différent de ϵ, on peut supposer que $(x', y', z') \neq (vx", vy", vz")$ et $(x', y', z') \neq (x"v, y"v, z"v)$ où $(x", y", z")$ est un branchement et v un mot différent de ϵ. Donc le branchement (x', y', z') entre forcément dans un des cas précédement. Ainsi, (x', y', z') est confluent. Alors, on a vu que les branchements (ux', uy', uz') et $(x'u, y'u, z'u)$ sont confluents aussi. Donc (x, y, z) est confluent. □

Exemple 2.3.8 : Soient $\Sigma = \{\alpha, \beta\}$ et $\mathcal{R} = \{\alpha\beta \longrightarrow \epsilon, \beta\alpha \longrightarrow \epsilon\}$, on a donc deux branchements critiques confluents $(\alpha\beta\alpha, \alpha, \alpha)\,, (\beta\alpha\beta, \beta, \beta)$.

Définition 2.3.9 : Un semi-système de réécriture $S = (\Sigma, \mathcal{R})$ est dit convergent ou complet s'il termine et s'il est confluent. On dit aussi que la relation binaire $\mathcal{R}$ est convergente.

Exemples 2.3.10 : 1. Soient $\Sigma = \{0, 1\}$ et $\mathcal{R}_1 = \{01 \longrightarrow 1, 11 \longrightarrow 1\}$, le système $(\Sigma, \mathcal{R})$ est convergent, mais si on remplace la règle $11 \longrightarrow 1$ par la règle $11 \longrightarrow 11$, le système $(\Sigma, \mathcal{R})$ non convergent.

2. Soient $\Sigma_2 = \{\alpha, \beta\}$ et $\mathcal{R}_2 = \{\alpha\beta \longrightarrow \epsilon\}$, le système $(\Sigma, \mathcal{R})$ est convergent, car on a, $\forall u, v \in \Sigma^* : u\alpha\beta v \underset{\mathcal{R}_2}{\Rightarrow} uv$, et on distingue deux cas :

· si $u \neq \alpha$ ou $v \neq \beta$ alors le mot uv est irréductible.

· si $u = \alpha$ et $v = \beta$, on $\alpha\alpha\beta\beta \underset{\mathcal{R}_2}{\Rightarrow} \alpha\beta \underset{\mathcal{R}_2}{\Rightarrow} \epsilon$.

3. Soient $\Sigma_3 = \{\alpha, \beta, \gamma\}$ et $\mathcal{R}_3 = \{\alpha\beta \longrightarrow \epsilon, \beta\gamma \longrightarrow \epsilon\}$, le système n'est pas confluent, car on a $\alpha\beta\gamma \underset{\mathcal{R}_3}{\Rightarrow} \alpha$ et $\alpha\beta\gamma \underset{\mathcal{R}_3}{\Rightarrow} \gamma$, mais les mots α et γ sous formes irréductibles.

Théorème 2.3.11 : Un semi-système de réécriture $(\Sigma, \mathcal{R})$ est convergent si et seulement s'il termine et s'il est localement confluent.

Démonstration

Supposons que $(\Sigma, \mathcal{R})$ est un semi-système de réécriture convergent, par définition, il termine et est confluent. Or, il est évident que la confluence implique la confluence locale puisque les branchements locaux sont aussi des branchements.

Réciproquement: Supposons que $(\Sigma, \mathcal{R})$ termine et qu'il est localement confluent. Comme $S = (\Sigma, \mathcal{R})$ termine, on peut utiliser le principe de récurrence pour monter que tout branchement est confluent. Plus précisément, on pose $P(x)$ la formule propositionnelle $P(x)$: (Tout branchement de source x est confluent).

Si x est une forme normale, alors le seul branchement de source x est (x, x, x) qui est toujours confluent: on a $x \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} x$ pour tout x. A présent, considérons un x dans Σ^* et supposons que $P(y)$ est vraie pour tous les $y \in \Sigma^*$ tels que $x \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$. Soit (x, y, z) un branchement. Distinguons deux cas:

1. Si $x = y$ (ou $x = z$), le branchement est confluent car $y = x \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} z$ et $z \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} z$.

2. Sinon, il existe $y^{'}$ et x' dans Σ^* tels que $x \underset{\mathcal{R}}{\Rightarrow} y' \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} y$ et $x \underset{\mathcal{R}}{\Rightarrow} z' \overset{+}{\underset{\mathcal{R}}{\Rightarrow}} z$: on a un branchement local $(x, y^{'}, z^{'})$ donc, comme $(\Sigma, \mathcal{R})$ est localement confluent, il existe $w \in \Sigma^*$ tel que $y' \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w$ et $z^{'} \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w$.

On obtient un branchement $(y, y^{'}, w)$: come $x \underset{\mathcal{R}}{\Rightarrow} y^{'}$, on peut appliquer l'hypothèse de récurrence à $y^{'}$ et en déduire qu'il existe $u \in \Sigma^*$ tel que $y \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} u$ et $w \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} u$. Puisque $x \underset{\mathcal{R}}{\Rightarrow} z^{'}$, on applique de nouveau l'hypothèse de récurrence à $z^{'}$ pour conclure que le branchement $(z^{'}, u, z)$ est confluent, c'est-à-dire qu'il existe $v \in \Sigma^*$ tel que $u \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} v$ et $z \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} v$. Le branchement (x, y, z) est donc confluent et $P(x)$ est vérifiée. □

Corollaire 2.3.12 : Si un semi-système de réécriture est noethérien et si tous les branchements critiques sont confluents, alors il est convergent.

2.4 Exercices

Exercice 2.4.1 :

- *Soit* $(\Sigma, \mathcal{R})$ *un semi-système de réécriture de mots,* $\psi : (\Sigma^*, \cdot) \longrightarrow (\mathbb{N}, +)$ un morphisme de *monoïdes et la fonction* $P : \Sigma^* \longrightarrow \mathbb{N}$ définie par :
  $P(w) = \sum_{i=1}^{i=|w|} n^i \times \psi(w(i)), n \in \mathbb{N} - \{0\}$ où $w(i)$ est la i-éme lettre de w.

1. Montrer que $\forall x, y \in \Sigma^* : P(xy) = P(x) + n^{|x|} \times P(y)$.
2. Calculer $P(xu\,\mathrm{y})$ pour tous $x, y, u \in \Sigma^*$.
3. Montrer que, si $\forall\,(u, v) \in \mathcal{R}, \begin{cases} |u| = |v| \\ \text{et} \\ P(u) > P(v) \end{cases}$, alors $(\Sigma, \mathcal{R})$ est noethérien.

- Soit $\Sigma = \{\alpha, \beta, \gamma\}$ un alphabet et la relation $\mathcal{R}$ définie par $\{(\beta\alpha, \alpha\beta), (\gamma\beta, \beta\gamma)\}$.

On considère le morphisme, $\psi : \Sigma^* \longrightarrow \mathbb{N}$, avec $\psi(\alpha) = 3, \psi(\beta) = 2, \psi(\gamma) = 1$ et la fonction de poids $P : \Sigma^* \longrightarrow \mathbb{N}$, où $P(w) = \sum_{i=1}^{i=|w|} 2^i \times \psi(w(i))$.

1. Vérifier que $(\Sigma, \mathcal{R})$ est noethérien.

Exercice 2.4.2 :

- *Soit* $(\Sigma, \mathcal{R})$ *un semi-système de réécriture de mots,* $\psi : (\Sigma^*, \cdot) \longrightarrow (\mathbb{N}, +)$ un morphisme de *monoïdes non trivial et la fonction* $P : \Sigma^* \longrightarrow \mathbb{N}$ définie par :
  $P(w) = \sum_{i=1}^{i=|w|} i \times \psi(w(i))$, où $w(i)$ est la i-éme lettre de w.

1. Montrer que, $\forall x, y \in \Sigma^* : P(xy) = P(x) + P(y) + |x| \times \psi(y)$.
2. Calculer $P(xu\,\mathrm{y})$ pour tous $x, y, u \in \Sigma^*$.
3. Montrer que, si $\forall u \longrightarrow v \in \mathcal{R}$, $\begin{cases} |u| = |v| & (C_1) \\ \text{et} & \\ P(u) > P(v) & (C_2) \\ \text{et} & \\ \psi(u) > \psi(u) & (C_3) \end{cases}$, alors $(\Sigma, \mathcal{R})$ est noethérien.

- Soit $\Sigma = \{\alpha, \beta, \gamma\}$ un alphabet et la relation $\mathcal{R}$ définie par $\{\beta\alpha \longrightarrow \beta\gamma, \alpha\beta \longrightarrow \alpha\gamma\}$.

On considère le morphisme, $\psi : \Sigma^* \longrightarrow \mathbb{N}$, avec $\psi(\alpha) = 2, \psi(\beta) = 1, \psi(\gamma) = 0$ et la fonction de poids $P : \Sigma^* \longrightarrow \mathbb{N}$, où $P(w) = \sum_{i=1}^{i=|w|} i \times \psi(w(i))$.

1. Montrer que $(\Sigma, \mathcal{R})$ est noethérien.

Exercice 2.4.3 :

- Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. Montrer que

1. Pour tout (x, y) de $\mathcal{R}$, le branchement (x, y, y) est confluent
2. Si (x, y, z) est un un branchement confluent, alors

• pour tout mot u de Σ^*, les deux branchement $(ux, u\,\mathrm{y}, uz)$ et (xu, yu, zu) sont confluents.

- pour tous mots u, v de Σ^*, le branchement $(uxv, u\,\mathrm{y}v, uzv)$ est confluent.

3. Pour tous $(x, y), (z, t)$ de $\mathcal{R}$, le branchement (xz, yz, xt) est confluent.

Exercice 2.4.4 :

- Soit $(\Sigma, \mathcal{R})$ un semi-système de réécriture. La congruence $\overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}}$ engendrée par $\mathcal{R}$ est définie par:

- $w \underset{\mathcal{R}}{\Longleftrightarrow} w'$, s'il existe u, v de Σ^* et $r \longrightarrow s \in (\mathcal{R} \cup \mathcal{R}^{-1})$ tels que $w = urv, w' = usv$.
- $w \overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}} w'$, s'il existe une suite finie de mots $u_0, u_1, ..., u_n$ de Σ^* avec,

$$u_0 = w, u_i \underset{\mathcal{R}}{\Longleftrightarrow} u_{i+1}, \forall 0 \leq i \leq n-1 \text{ et } u_n = w'.$$

1. Déterminer le monoïde quotient $\Sigma^*/\overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}}$ dans les cas suivants :

- Soient $\Sigma = \{\alpha, \beta\}$ et $\mathcal{R} = \{\alpha\beta \longrightarrow \beta\alpha\}$.
- Soientt $\Sigma = \{\alpha\}$, et $\mathcal{R} = \{\alpha^2 \longrightarrow \epsilon\}$.

Exercice 2.4.5 :

- *Soit* $(\Sigma, \mathcal{R})$ *un semi-système de réécriture de mots.* A tout langage Γ sur Σ, on associe le langage :

$$L(\Gamma, \mathcal{R}) = \left\{ w \in \Sigma^*, \exists \gamma \in \Gamma : \gamma \overset{*}{\underset{\mathcal{R}}{\Rightarrow}} w \right\}.$$

1. Déterminer le langage $L(\Gamma, \mathcal{R})$ dans les cas suivants :

- Soient $\Sigma_1 = \{\alpha, \beta\}, \mathcal{R}_1 = \{(\alpha\beta, \alpha\alpha\beta\beta), (\beta\alpha, \beta\beta\alpha)\}$ et $\Gamma_1 = \{\alpha\beta, \beta\alpha\}$.
- Soient $\Sigma_2 = \{A, B, C\}, \mathcal{R}_2 = \{(A, BA), (A, C)\}$ et $\Gamma_2 = \{A\}$,

Chapitre 3

Automates et langages rationnels

Introduction

Dans ce chapitre, on donne les notions et les propriétés de base des automates finis. Un automate fini est une représentation du comportement d'un système fini, un automate est formé par l'ensemble des différents états possibles du système, reliés entre eux par des conditions : le système passe d'un état dans un autre quand une conditiondonnée est vérifiée. Les automates finis permettant de représenter d'une manière fini certaines ensembles (les ensembles rationnels) du monoïde libre Σ^*.

Contenu

3.1. Notations et définitions.
3.2. Langages rationnels et automates finis.
3.3. Automate minimal.
3.4. Exercices

3.1 Notations et définitions.

Définition 3.1.1 : Un automate est un 5-uplet $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ où,

- Σ est un ensemble fini dit l'alphabet d'entrée.
- Q est un ensemble dit l'ensemble des états.
- q_0 est l'etat initial.
- $F \subseteq Q$ est l'ensemble des états finals.

$\delta \subseteq Q \times \Sigma \times Q$ est l'ensemble des transitions.

Un automate fini est un automate $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ dont l'ensemble d'états Q est fini.

Nous représentons un automate fini $\mathcal{A} = (Q, I, F, \Sigma, \delta)$ de la manière suivante :

les états de $\mathcal{A}$ sont les sommets d'un graphe orienté et sont représentés par des cercles.

Si $\delta(q, \alpha) = q', q, q' \in Q, \alpha \in \Sigma$, alors on trace un arc orienté de q vers q' et d'étiquette $\alpha, q \underset{\longrightarrow}{\alpha} q'$. Les états finals sont représentés à un double cercle et l'état initial est désigné par une flèche entrante sans étiquette. Enfin si deux lettres α, β sont telles que $\delta(q, \alpha) = q'$ et $\delta(q, \beta) = q'$, on s'autorise à dessiner un unique arc portant deux étiquettes séparés par une virgule $q \underset{\longrightarrow}{\alpha, \beta} q'$. Cette convention s'adapte à plus de deux lettres.

Exemple 3.1.2 : Soit l'automate $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ où $\Sigma = \{\alpha, \beta\}, Q = \{1, 2\}$, $q_0 = 1, F = \{2\}, \delta = \{(1, \alpha, 1), (1, \beta, 2), (2, \beta, 2)\}$.

Remarque 3.1.3 : Soit l'automate $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$. On étend naturellement la fonction de transition δ à $Q \times \Sigma^*$ de la manière suivante : $\delta(q, \epsilon) = q$ et $\delta(q, \alpha w) = \delta(\delta(q, \alpha), w)$, $\alpha \in \Sigma, w \in \Sigma^*, q \in Q$.

Définition 3.1.4 : Soit $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ un automate fini. Un mot c de δ^* est un chemin dans $\mathcal{A}$ ssi il s'écrit,

$c = (q_1, x_1, q_1')(q_2, x_2, q_2') \ldots (q_i, x_i, q_i') \ldots (q_n, x_n, q_n')$ et il vérifie la condition $c = \epsilon$ ou $\forall i \in [1, n-1], q_i' = q_{i+1}$.

- n est la longueur du chemin.
- Ce chemin mêne de l'état q_1 à l'état q_n'.
- On convient que ϵ mêne de q à q, pour tout $q \in Q$.
- On appelle trace le morphisme $h : \delta^* \longrightarrow \Sigma^*$ tel que: $\forall (q, x, q') \in \delta, h((q, x, q')) = x$.
- Un mot $w \in \Sigma^*$ est accepté (ou reconnu) par $\mathcal{A}$ ssi il existe un chemin c dans $\mathcal{A}$ menant d'un état q_0 à un état $q_f \in F$ tel que $h(c) = w$.

• On appelle langage accepté (ou reconnu) par $\mathcal{A}$, noté $L_{\mathcal{A}}$, l'ensemble des mots acceptés par $\mathcal{A}$ .i. e,

$$L_{\mathcal{A}} = \{w \in \Sigma^*, \exists q_f \in F : \delta^*(q_0, w) = q_f\}.$$

Définition 3.1.5 : Un automate fini $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ est automate fini déterministe ssi δ est une fonction de transition : $\delta : Q \times \Sigma \longrightarrow Q$ (d'un état donné, il part au plus une seule flèche étiquetée par une lettre donnée). Un automate fini déterministe est complet ssi δ est une fonction définie sur l'ensemble $Q \times \Sigma$ tout entier. Dans un automate fini non déterministe (A.F.N) il peut y avoir le choix entre plusieurs chemins lors de la lecture d'un mot.

Examples 3.1.6 : 1. Soit l'automate $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$ où $\Sigma = \{0, 1\}, Q = \{q_0, q_1\}$, $\delta = \{(q_0, 0, q_1), (q_0, 1, q_0)\}$, q_0 est l'état initial, $F = \{q_1\}$. Ici, $\mathcal{A}$ reconnait le langage décrit par l'expression 1^*0.

2. Soit l'automate $\mathcal{B} = (Q, q_0, F, \Sigma, \delta)$ où $\Sigma = \{0, 1\}, Q = \{q_0, q_1\}$, $\delta = \{(q_0, 0, q_1), (q_0, 1, q_0), (q_1, 1, q_1)\}$, q_0 est l'état initial, $F = \{q_1\}$. Ici, $\mathcal{B}$ accepte les mots du langage décrit par l'expression régulière 1^*01^*.

3. Soit l'automate $\mathcal{C} = (Q, q_0, F, \Sigma, \delta)$ où $\Sigma = \{0, 1\}, Q = \{q_0, q_1, q_2\}$, $\delta = \{(q_0, 0, q_1), (q_0, 1, q_0), (q_1, 0, q_2), (q_1, 1, q_1), (q_2, 0, q_2), (q_2, 1, q_2)\}$, q_0 est l'état initial, $F = \{q_1\}$. Ici, $\mathcal{C}$ accepte les mots du langage décrit par l'expression régulière 1^*01^*. $\mathcal{C}$ est la version complétée de $\mathcal{B}$.

Définition 3.1.7 : Soit l'automate $\mathcal{A} = (Q, q_0, F, \Sigma, \delta)$. La fonction de transitions δ définit un morphisme Ψ de Σ^* dans le monoïde Q^Q de toutes les applications de Q dans Q: pour tout $w \in \Sigma^*, \Psi(w)$ est l'application qui à tout $q \in Q$ associe l'élément $q' \in Q$ ssi $\delta^*(q, w) = q'$. Le sous monoïde $\Psi(\Sigma^*)$ de Q^Q est appelé le monoïde de transitions de $\mathcal{A}$.

Exercice 3.1.8 : Construit l'automate fini déterministe qui reconnaît le langage L suivant:

$$L = \{w \in \{0, 1\}^* : |w|_1 \equiv 0\,[4]\}.$$

Proposition 3.1.9 : Soit $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ un automate fini déterministe.

1. La relation $\mathcal{R}_{\mathcal{A}}$ définie sur Σ^* par $w_1 \mathcal{R}_{\mathcal{A}} w_2 \iff \delta^*(q_0, w_1) = \delta^*(q_0, w_2)$, est une relation d'équivalence.

2. Pour tout langage L de Σ^*, la relation $\mathcal{R}_L$ définie sur Σ^* par :
$w_1 \mathcal{R}_L w_2 \Longleftrightarrow (\forall x \in \Sigma^* : w_1 x \in L \Longleftrightarrow w_2 x \in L)$ est une relation d'équivalence.

3. Les relations $\mathcal{R}_{\mathcal{A}}$ et $\mathcal{R}_L$ sont invariantes à droite.

4. La relation $\mathcal{R}_{\mathcal{A}}$ est d'indice fini.

Démonstration : Il est claire que les deux relations $\mathcal{R}_{\mathcal{A}}$ et $\mathcal{R}_L$ sont des relations d'équivalence. Pour (3), supposons que $w_1 \mathcal{R}_{\mathcal{A}} w_2$ et montrons que pour tout $u \in \Sigma^*, w_1 u \mathcal{R}_{\mathcal{A}} w_2 u$. On a $\delta^*(q_0, w_1 u) = \delta^*(\delta^*(q_0, w_1), u) = \delta^*(\delta^*(q_0, w_1), u) = \delta^*(q_0, w_1 u)$. Donc $w_1 u \mathcal{R}_{\mathcal{A}} w_2 u$.

supposons que $w_1 \mathcal{R}_L w_2$ et montrons que pour tout $u \in \Sigma^*, w_1 u \mathcal{R}_L w_2 u$. On a $w_1 \mathcal{R}_L w_2 \Longleftrightarrow (\forall x \in \Sigma^*, w_1 x \in L \Longleftrightarrow w_2 x \in L)$. Soit $x \in \Sigma^*$ tel que $(w_1 u) t \in L$, on a $(w_1 u) t \in L \Longleftrightarrow w_1 (ut) \in L \Longleftrightarrow w_2 (ut) \in L \Longleftrightarrow (w_2 u) t \in L$. Donc $w_1 u \mathcal{R}_L w_2 u$.

Pour (4), on a les classes d'équivalence de $\mathcal{R}_{\mathcal{A}}$ sont en bijection avec les états de $\mathcal{A}$ car : $\forall w \in \Sigma^*, [w]_{\mathcal{R}_{\mathcal{A}}} = \{x \in \Sigma^* : \delta^*(q_0, w) = \delta^*(q_0, x)\}$.

Théorème 3.1.10 (Myhill-Nerode): Soit L un langage sur un alphabet Σ. Les assertions suivantes sont équivalentes :

1. L est accepté par un automate fini déterministe.

2. L est la réunion de classes d'équivalence d'une relation invariante à droite d'indice fini.

3. La relation $\mathcal{R}_L$ est d'indice fini.

Démonstration : 1. $\Longrightarrow$ 2. Soit $\mathcal{A}$ un automate fini déterministe acceptant L. Alors la relation $\mathcal{R}_{\mathcal{A}}$ est invariante à droite et L est réunion des classes de $\mathcal{R}_{\mathcal{A}}$ correspondant aux états terminaux de $\mathcal{A}$.

2. $\Longrightarrow$ 3. Supposons que L est réunion de classes de $\mathcal{R}$ d'indice fini. Montrons que $x \mathcal{R} y \Longrightarrow x \mathcal{R}_L y$. Si $x \mathcal{R} y$, alors $xz \mathcal{R} yz$ (invariante à droite). Les mots xz et yz sont donc dans la même classe de $\mathcal{R}$. Soit cette classe est une des classes qui constitue L, auquel cas xz et yz sont tous deux dans L, soit ce n'est pas le cas et ni xz ni yz ne sont pas dans L. Dans tous les cas, on a quel que soit z, $xz \in L \Longleftrightarrow yz \in L$. Donc $x \mathcal{R}_L y$. On en déduit qu'une classe de $\mathcal{R}$ est incluse de $\mathcal{R}_L$, chaque classe de $\mathcal{R}_L$ est donc réunion de classe de $\mathcal{R}$. $\mathcal{R}_L$ est donc d'indice fini.

3. $\Longrightarrow$ 1. Supposons que $\mathcal{R}_L$ est d'indice fini. Notons c_0 la classe du mot vide ϵ et $c_1, ..., c_n$ les autres classes. Soit $\sigma \in \Sigma$, définissons $\delta(c_i, \sigma)$, pour cela, soit x un mot de c_i, posons

$\delta(c_i, \sigma)$ = la classe de $x\sigma$. Il faut vérifier que cette classe ne dépend pas du choix de x. ceci résulte de l'invariance à droite de $\mathcal{R}_L$. La fonction δ est donc définie. Disons maintenant que c_k est terminal si tous les mots de c_k sont dans L (si un élément de c_k est dans L, alors tous le sont, car $\mathcal{R}_L$ est invariante à droite par ϵ. On a ainsi définit un automate fini $\mathcal{A}$. On établit par récurrence sur $|x|$ que $\delta(c_0, x)$ = la classe de x. Montrons que $\mathcal{A}$ accepte L. On a $u \in L_{\mathcal{A}} \Longleftrightarrow \delta(c_0, u) = c_k$ terminal $\Longleftrightarrow u \in L$.

Théorème 3.1.11 : Si un langage est reconnu par un automate fini, alors il est également reconnu par un automate fini déterministe.

Démonstration : Si l'automate fini du départ $\mathcal{A}$ est déterministe, c'est évident. Si l'automate de départ n'est pas déterministe, on se propose de construire un automate fini déterministe $\mathcal{D}$ qui intègre les choix existant dans l'automate de départ. Soit un AFN $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$, on construit l'automate $\mathcal{D} = (Q', q'_o, F', \Sigma, \delta')$, Q' sera inclus dans $p(Q)$, $q'_o = \{q_0\}, F' = \{X \in Q' : X \cap F \neq \emptyset\}$, $\delta'(X, \sigma) = X'$ tel que $X' = \{x' : \exists x \in X, \delta(x, \sigma) = x'\}$ pour tous $\sigma \in \Sigma, X \in Q'$.

Exemple 3.1.12 : Soit l'AFN $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ où, $\Sigma = \{0, 1\}, Q = \{q_0, q_1\}, \delta = \{(q_0, 0, q_0), (q_0, 0, q_1), (q_0, 1, q_1), (q_1, 1, q_1)\}$, q_0 est létat initial, $F = \{1\}$.

On construit $\mathcal{D} = (Q', q'_o, F', \Sigma, \delta')$, on a $q'_o = \{q_0\}$, la fonction de transition δ' est définie comme suit :

δ'	0	1
$\{q_0\}$	$\{q_0, q_1\}$	$\{q_1\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_1\}$
$\{q_1\}$		$\{q_1\}$

$F' = \{\{q_0, q_1\}, \{q_1\}\}$.

Définition 3.1.12 : Soient $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ et $\mathcal{A}' = (Q', q'_o, F', \Sigma, \delta')$ deux automates finis déterministes complets. Un morphisme de $\mathcal{A}$ dans $\mathcal{A}'$ est une application $h : Q \longrightarrow Q'$ vérifiant les conditions suivantes :

1. $h(q_o) = q'_o$,
2. $\forall q \in Q, \forall \sigma \in \Sigma, h(\delta(q, \sigma)) = \delta'(h(q))$,
3. $q \in Q \Longleftrightarrow h(q) \in F'$.

3.2 Langages rationnels et automates finis

Définition 3.2.1 : L'ensemble des langages rationnels est le plus petit ensemble R contenant les langages finis et clos par union, intersection et étoile.

définition inductive :

(B)

$$\emptyset \in R$$
$$\{\epsilon\} \in R$$
$$\{\sigma\} \in R, \text{ pour tout } \sigma \in \Sigma$$

(I) Si $L, M \in R$ alors

$$L \cup M \in R$$
$$L.M \in R$$
$$L^* \in R$$

Définition 3.2.2 : Soit Σ un alphabet, les langages reconnaissables sur Σ, noté $\mathrm{Re}\,c\,(\Sigma^*)$ sont les langages reconnus par un automate fini déterministe.

Théorème 3.2.3 (Kleene) : Un langage sur un alphabet Σ est régulier si et seulement si il reconnu par un automate fini.

Théorème 3.2.4 : 1. Les langages reconnaissables sont clos par complémentation.

2. Les langages reconnaissables sont clos par union ensembliste.
3. Les langages reconnaissables sont clos par intersection ensembliste.
4. Les langages reconnaissables sont clos par miroir.
5. Les langages reconnaissables sont clos par concaténation.
6. Les langages reconnaissables sont clos par étoile.

Démonstration : voire (François Yvon [3]).

Définition 3.2.5 : Soit $L \subseteq \Sigma^*$ un langage, si w est un mot sur Σ, on note par $w^{-1}L$ l'ensemble des mots qui concaténés avec w appartiennent à L, i.e, $w^{-1}L = \{u \in \Sigma^* : wu \in L\}$.
On appelle résiduel du langage L, tout langage de la forme $w^{-1}L$, on note $Q\,(L) = \{w^{-1}L, w \in \Sigma^*\}$.
Pour tout langage $L \subseteq \Sigma^*$, on peut définir une relation sur Σ^*, notée $\sim_L$ comme suit :

$$w_1 \sim_L w_2 \Longleftrightarrow w_1^{-1}L \sim_L w_2^{-1}L.$$

En d'autre termes, $w_1 \sim_L w_2$ si, et seulement si, pour tout $u \in \Sigma^*, w_1u \in L \Longleftrightarrow w_2u \in L$.

Exemple 3.2.6 : Soient $\Sigma = \{a, b\}$ et $L = \{w \in \{a, b\}^* : |w|_a \equiv 0\,[3]\}$, pour ce langage, on a par exemple $b \nsim_L ab$ car pour $u = aa$, $bu \notin L$ et $abu \in L$. Par contre $a \sim_L ababaa$ car $a^{-1}L = (ababaa)^{-1}\,L = \{w \in \{a, b\}^* : |w|_a \equiv 2\,[3]\}$.

Exemple 3.2.7 : Soient Σ un alphabet et $L = \left\{w \in \Sigma^* : \exists\sigma \in \Sigma, \exists i \in \mathbb{N}, u = \sigma^{2^i}\right\}$. On pour tous $i, j \in \mathbb{N}$ avec $i \neq j$ $\sigma^{2^i}\sigma^{2^i} = \sigma^{2^{i+1}}$ et $\sigma^{2^j}\sigma^{2^i} = \sigma^{2^i+2^j} \notin L$, donc il ya une infinité de classes d'équivalence modulo $\sim_L$ et par conséquent L n'est pas rationnel.

3.3 Automate minimal

Dans cette section, nous donnons une caractérisation des langages reconnaissables, à partir de laquelle nous introduisons la notion d'automate canonique (minimal) (en nombre d'états) d'un langage. Nous présentons ensuite un algorithme pour construire l'automate canonique d'un langage reconnaissable représenté par un automate fini déterministe quelconque.

Définition 3.3.1 : Soient Σ un alphabet et L un langage rationnel sur Σ. On définit l'automate minimal de L $\mathcal{A}_L = (Q_L, q_o, F_L, \Sigma, \delta_L)$ comme suit :

$Q_L = \{w^{-1}L, w \in \Sigma^*\}$,

$q_o = \epsilon^{-1}L = L$,

$F_L = \{w^{-1}L, w \in L\}$

$\delta_L\,(q, \sigma) = \sigma^{-1}q$, pour tous $q \in Q_L, \sigma \in \Sigma$.

La fonction de transition δ_L s'étend à $Q_L \times \Sigma^*$ par $\delta_L\,(q, w) = w^{-1}q, \forall q \in Q_L, \forall w \in \Sigma^*$.

Remarque 3.3.2 : Au vu de la définition de $\sim_L$, il est clair que l'ensemble des états de $\mathcal{A}_L$, $\{w^{-1}L, w \in \Sigma^*\}$, est en bijection avec l'ensemble quotient $\Sigma^*/\sim_L = \left\{[w]_{\sim_L} : w \in \Sigma^*\right\}$. En effet, à chaque classe d'équivalence $[w]_{\sim_L}$ modulo $\sim_L$ correspond un état $w^{-1}L$ de l'automate minimal $\mathcal{A}_L$ et réciproquement. C'est pour cette raison que on trouve une définition de l'automate minimal en termes des classes d'équivalence de $\sim_L$. Ainsi, on aurait pu définir l'automate minimal comme suit :

$Q_L = \left\{[w]_{\sim_L}, w \in \Sigma^*\right\}$,

$q_o = [\epsilon]_{\sim_L}$,

$F_L = \left\{[w]_{\sim_L}, w \in L\right\}$

$\delta_L\left([w]_{\sim_L}, \sigma\right) = [w\sigma]_{\sim_L}$.

Exemple 3.3.3 : Soit le langage $L = \{w \in \{a,b\}^* : |w|_a \equiv 0\,[3]\}$, la congruence de Nerode possède trois classes déquivalence $[\epsilon]_{\sim_L}, [a]_{\sim_L}, [aa]_{\sim_L}$, donc l'automate minimale $\mathcal{A}_L$ de L a trois états $L, a^{-1}L, (aa)^{-1}L, q_o = L, F_L = \{L\}$. La fonction de tansition est définie par :

δ_L	a	b
L	$a^{-1}L$	L
$a^{-1}L$	$(aa)^{-1}L$	$a^{-1}L$
$(aa)^{-1}L$	L	$(aa)^{-1}L$

Proposition 3.3.4 : L'automate minimal d'un langage $L \subseteq \Sigma^*$ accepte L.

Démonstration : En effet, soit $w \in \Sigma^*$,

$w \in L_{\mathcal{A}_L} \Longleftrightarrow \delta_L(L,w) \in F_L \Longleftrightarrow w^{-1}L \in F_L \Longleftrightarrow w \in L$.

Définition 3.3.5 : Soit $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ un automate fini déterministe complet, on définit sur Q la suite de relation d'équivalence $(\mathcal{R}_k)_{k\in\mathbb{N}}$ comme suit :

$(q\mathcal{R}_0 q') \Longleftrightarrow (q \in F \Longleftrightarrow q' \in F)$,

$\forall k \in \mathbb{N}, (q\mathcal{R}_{k+1}q') \Longleftrightarrow ((q\mathcal{R}_k q')$ et $\forall \sigma \in \Sigma, \delta(q,\sigma)\,\mathcal{R}_k\,\delta(q',\sigma))$.

La relation d'équivalence sur Q notée $\approx$ associe à la la suite de relation $(\mathcal{R}_k)_{k\in\mathbb{N}}$ est définie par :

$\forall k \in \mathbb{N}, \mathcal{R}_{k+1} = \mathcal{R}_k \Longrightarrow \approx = \mathcal{R}_k$.

On considère l'automate quotient $\mathcal{A}/\approx$ de $\mathcal{A}$, $\mathcal{A}/\approx = (Q_\approx, q_o, F_\approx, \Sigma, \delta_\approx)$ où,

$Q_\approx = \{[q]_\approx, q \in Q\}$,

$q_o = [q_0]_\approx$,

$F_\approx = \{[q]_\approx, q \in F\}$,

$\delta_\approx([q]_\approx, \sigma) = [\delta(q,\sigma)]_\approx$.

L'automate quotient $\mathcal{A}/\approx$ est l'automate minimal de $\mathcal{A}$.

Exemple 3.3.6 : Soit l'automate $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ où, $Q = \{1,2,3,4,5,6,7\}$,
$\Sigma = \{a,b\}, q_0 = \{1\}, F = \{3,4,5\}$ et la fonction de transition δ est définie par :

δ	a	b
1	2	4
2	3	7
3	7	2
4	5	7
5	7	6
6	5	7
7	7	7

On a $(q\mathcal{R}_0q') \Longleftrightarrow (q \in F \Longleftrightarrow q' \in F)$, donc $Q/\mathcal{R}_0 = \{\{1,2,6,7\},\{3,4,5\}\}$, la relation $\mathcal{R}_1$ est définie comme suit :

$(q\mathcal{R}_1q') \Longleftrightarrow ((q\mathcal{R}_0q')$ et $\forall\sigma \in \Sigma, \delta(q,\sigma)\,\mathcal{R}_0\delta(q',\sigma))$, i.e,

$((q\mathcal{R}_0q')$ et $\delta(q,a)\,\mathcal{R}_0\delta(q',a)$ et $\delta(q,b)\,\mathcal{R}_0\delta(q',b))$.

Donc $Q/\mathcal{R}_1 = \{\{1\},\{2,6\},\{7\},\{3,5\},\{4\}\}$. La relation $\mathcal{R}_2$ est définie comme suit :

$(q\mathcal{R}_2q') \Longleftrightarrow ((q\mathcal{R}_1q')$ et $\forall\sigma \in \Sigma, \delta(q,\sigma)\,\mathcal{R}_1\delta(q',\sigma))$, i.e,

$((q\mathcal{R}_1q')$ et $\delta(q,a)\,\mathcal{R}_1\delta(q',a)$ et $\delta(q,b)\,\mathcal{R}_1\delta(q',b))$.

On a $\mathcal{R}_2 = \mathcal{R}_1$ et par conséquent $Q/\mathcal{R}_2 = Q/\mathcal{R}_1$. Finalement la relation $\approx$ est égale à la relation $\mathcal{R}_1$.

L'automate minimal de $\mathcal{A}$ est l'automate $\mathcal{A}/\approx = (Q_\approx, q_o, F_\approx, \Sigma, \delta_\approx)$ où,

$Q_\approx = \{[q]_\approx, q \in Q\} = \{\{1\},\{2,6\},\{7\},\{3,5\},\{4\}\}$,

$q_o = [q_0]_\approx = \{\{1\}\}$,

$F_\approx = \{[q]_\approx, q \in F\} = \{\{2,6\},\{3,5\},\{4\}\}$,

La fonction de transition $\delta_\approx$ est définie par :

$\delta_\approx$	a	b
$\{1\}$	$\{2,6\}$	$\{4\}$
$\{2,6\}$	$\{3,5\}$	$\{7\}$
$\{7\}$	$\{7\}$	$\{7\}$
$\{3,5\}$	$\{7\}$	$\{2,6\}$
$\{4\}$	$\{3,5\}$	$\{7\}$

3.4 Exercices

Exercice 3.4.1 :

Soit l'AFD $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ où $Q = \{1, 2, 3\}, \Sigma = \{a, b\}, q_o = 1, F = \{3\}$ et où la fonction de transition est donnée par :

δ	a	b
1	1	2
2	3	2
3	3	1

1. Tracer le diagramme d'états de $\mathcal{A}$.

2. Donner l'éxécution de $\mathcal{A}$ sur les mots suivants : $abba, bbbabb, bababa, bbbaa$.

3. Quel est le langage accepté par $\mathcal{A}$ (en donner une expression régulière)?

Exercice 3.4.2 :

Soit $\Sigma = \{a, b\}$. Construire un AFD acceptant le langage suivant :

1. $\{w \in \Sigma^* : |w| \equiv 0\,[3]\}$,

2. $\{w \in \Sigma^* : |w|_a \equiv 0\,[3]\}$,

3. $\{w \in \Sigma^* : |w| \equiv 1\,[3]\}$,

4. $\{w \in \Sigma^* : |w|_a \leq 4\}$,

5. $\{w \in \Sigma^* : |w|_a > 4\}$.

Exercice 3.4.3 :

Soit $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ un AFD, montrer que,

1. Pour tout $q \in Q$ l'ensemble $Stab(q) = \{w \in \Sigma^* : \delta(q, w) = q\}$ est un sous monoïde de $(\Sigma^*, \cdot, \epsilon)$.

2. Pour tous $q, q' \in Q$, pour tout $w \in \Sigma^*$, si $q' = \delta(q, w)$ et $q = \delta(q', \widetilde{w})$ où $\widetilde{w}$ est l'image miroir de w, alors $w \cdot Stab(q') \cdot \widetilde{w} \subseteq Stab(q)$ et $\forall u \in \Sigma^* : u \in Stab(q) \Longrightarrow \widetilde{w}uw \in Stab(q')$.

3. Pour tout $q \in Q$, la relation notée R_q définie sur Σ^* par :
$uR_qv \Longleftrightarrow Stab(q) \cdot u = Stab(q) \cdot v$, est une relation d'équivalence

4. $uR_qv \Longleftrightarrow \exists x, y \in Stab(q) : u = x \cdot v$ et $v = y \cdot u$.

5. L'application $\varphi : \Sigma^*/R_q \longrightarrow O_q$ définie par $\varphi(\overline{u}) = \delta(q, u)$ est bien définie, où $O_q = \{\delta(q, w), w \in \Sigma^*\}$ et $\overline{u}$ désigne la classe d'équivalence de u modulo R_q.

6. La relation ρ définie sur Q par : $q\rho q' \Longleftrightarrow \exists w \in \Sigma^* : q' = \delta(q, w)$ et $q = \delta(q', \widetilde{w})$, est une relation d'équivalence.

Exercice 3.4.4 :

Soit $L = \{ab, aab, aba, ba, bb, aaa\}$.

1. Quels sont les différents ensembles de la forme $w^{-1}L, w \in \{a, b\}^*$.

2. En déduire l'automate minimal de L.

Exercice 3.4.5 :

Soit $L = \{w \in \{a, b\}^* : |w|_a \equiv 0\,[2] \text{ et } |w|_b \equiv 0\,[2]\}$.

Construire un automate fini $\mathcal{A}$ tel que $L_{\mathcal{A}} = L$.

Exercice 3.4.6 :

1. Soit Q un ensemble fini. Montrer que l'ensemble $Q^Q = \{f : Q \longrightarrow Q,\ f \text{ est une fonction}\}$ muni de l'opération de composition de fonctions est un monoïde.

2. Soit $\mathcal{A} = (Q, q_o, F, \Sigma, \delta)$ un AFD. On pose, pour tout $w \in \Sigma^*$,
$f_w : Q \longrightarrow Q, q \longmapsto f_w(q) = \delta(q, w)$. Montrer que $M_t = \{f_w, w \in \Sigma^*\}$ est un sous monoïde de $\left(Q^Q, \circ\right)$.

Exercice 3.4.7 :

On considère le langage L sur l'alphabet binaire $\Sigma = \{0, 1\}$ décrit par l'expression régulière suivante :

$$E = 0^*1\,(10^*1 + 0)^*.$$

- Construisez l'automate minimal $\mathcal{A}$ reconnaissant le langage L par la méthode des résiduels à gauche.

Chapitre 4

Langages algébriques

Introduction

Les chapitres précédents nous ont donnés un apercçu assez complet des langages réguliers. Nous avons constaté, que des langages comme $\{a^n b^n, n \in \mathbb{N}\}$ sur l'alphabet $\{a, b\}$, pourtant "relativement simples" d'un point de vue syntaxique, n'était pas réguliers. Dans ce chapitre, nous allons présenter une famille de langages qui sont générés par des méthodes plus riches que les expressions régulières. Nous allons introduire la notion de grammaire hors contexte. Un langage généré par une telle grammaire sera dit algébrique (ou hors contexte). Historiquement, ces langages ont été introduits par N. Chomsky dans le but initial de formaliser les propriétés grammaticales de langues naturelles.

Contenu

4.1. Notations et définitions.
4.2. Grammaires et langages réguliers.
4.3. Exercices

4.1 Notations et définitions

Définition 4.1.1 : Soient V et Σ deux alphabets finis (supposés disjoints). Une grammaire hors contexte, ou grammaire algébrique, est la donnée d'un 4-uple $G = (V, \Sigma, P, S)$ où $P \subseteq V \times (V \cup \Sigma)^*$ est un ensemble fini, appelé l'ensemble des règles de dérivation (ou productions) de G et $S \in V$ est le symbole initial de G. Les éléments de l'alphabet V sont appelés variables (ou symboles non terminaux) et les éléments de l'alphabet Σ sont les symboles terminaux. Nous prendrons généralement la convention de représenter les symboles non terminaux par des lettres majuscules et les symboles terminaux par des minuscules.

Exemple 4.1.2 : Soit la grammaire hors contexte $G = (V, \Sigma, P, S)$ où $V = \{S, A\}, \Sigma = \{a, b\}$ et les productions de G données par $P = \{(S, AA), (A, AAA), (A, bA), (A, Ab), (A, a)\}$.

Notation 4.1.3 : Soient $G = (V, \Sigma, P, S)$ une grammaire, $A \in V$ une variable, $w \in (V \cup \Sigma)^*$ un mot et $(A, w) \in P$ une règle de dérivation. On dit que A (resp. w) est le premier (resp. second) membre de la production (A, w).

Si $A \in V$ est une variable et $(A, w_1), ..., (A, w_n) \in P$ sont des productions ayant A pour premier membre et où $w_1, ..., w_n \in (V \cup \Sigma)^*$, alors on note $A \longrightarrow w_1/w_2/.../w_n$.

Définition 4.1.4 : Soit $G = (V, \Sigma, P, S)$ une grammaire. Si w peut s'écrire xAy avec $A \in V$ et $x, y \in (V \cup \Sigma)^*$, alors on note $w \Longrightarrow_G z$ lorsque $z = xu$ y avec $(A, u) \in P$.On note $\Longrightarrow_G^*$ la fermeture réflexive et transitive de $\Longrightarrow_G$. Ainsi, $w \Longrightarrow_G^* z$ si $z = w$ ou s'il existe des mots $w_1, ..., w_n \in (V \cup \Sigma)^*, n \geq 0$ tels que

$$w \Longrightarrow_G w_1 \Longrightarrow_G w_2 \Longrightarrow_G ... \Longrightarrow_G w_n \Longrightarrow_G z.$$

Définition 4.1.5 : Soit $G = (V, \Sigma, P, S)$ une grammaire. Le langage généré par G est l'ensemble des mots sur Σ qui s'otiennent par dérivation à partir du symbole initial S, i.e., $L(G) = \{w \in \Sigma^* : S \Longrightarrow_G^* w\}$.

Un langage $L \subseteq \Sigma^*$ est algébrique ou hors contexte s'il existe une grammaire hors contexte $G = (V, \Sigma, P, S)$ telle que $L = L(G)$. Deux grammaires G et H sont équivalentes si elles génèrent le même langage , i.e., si $L(G) = L(H)$.

Exemple 4.1.6 : Soit la grammaire hors contexte $G = (V, \Sigma, P, S)$ où $V = \{S, A\}, \Sigma = \{a, b\}$ et les productions de G données par : $S \longrightarrow AA, A \longrightarrow AAA/bA/Ab/a$. Le mot

$ababaa$ appartient à $L(G)$ car $S \Longrightarrow_G AA \Longrightarrow_G aA \Longrightarrow_G aAAA \Longrightarrow_G abAAA \Longrightarrow_G abaAA \Longrightarrow_G ababAA \Longrightarrow_G ababaA \Longrightarrow_G ababa$.

Le mot $ababaa$ est obtenu à partir de S par une dérivation de longueur 8. La suite des règles appliquées donnant lieu à un mot donné n'est pas nécessairement unique. En effet, on peut générer le mot $ababaa$ à partir du symbole initial S de diverses façons :

$S \Longrightarrow_G AA \Longrightarrow_G AAAA \Longrightarrow_G aAAA \Longrightarrow_G abAAA \Longrightarrow_G abaAA \Longrightarrow_G ababAA \Longrightarrow_G ababaA \Longrightarrow_G ababaa$.

$S \Longrightarrow_G AA \Longrightarrow_G Aa \Longrightarrow_G AAAa \Longrightarrow_G AAbAa \Longrightarrow_G AAbaa \Longrightarrow_G AbAbaa \Longrightarrow_G Ababaa \Longrightarrow_G ababaa$.

$S \Longrightarrow_G AA \Longrightarrow_G aA \Longrightarrow_G aAAA \Longrightarrow_G aAAa \Longrightarrow_G abAAa \Longrightarrow_G abAbAa \Longrightarrow_G ababAa \Longrightarrow_G ababaa$.

Exemple 4.1.7 : La grammaire ci-dessous génère exactement le langage $\{a^n b^n, n \in \mathbb{N}\}$. Considérons $G = (V, \Sigma, P, S)$ où $V = \{S\}, \Sigma = \{a, b\}$ et les productions de G données par $S \longrightarrow aSb/\epsilon$.

Exemple 4.1.7 : (Langage de Dyck). On considére la grammaire $G = (V, \Sigma, P, S)$ où $V = \{S, T\}, \Sigma = \{a_1, \overline{a_1}, ..., a_n, \overline{a_n}\}$ et les productions de P données par $S \longrightarrow ST/\epsilon, T \longrightarrow a_1 S\overline{a_1}/.../a_n S\overline{a_n}$.

4.2 Grammaires et langages réguliers

Le but de cette section est de montrer que l'ensemble des langages réguliers est un sous ensemble strict de l'ensemble des langages algébriques. En montrant que la famille des langages algébriquescontient le langage vide, les langages $\{\sigma\}, \sigma \in \Sigma$, et est stable pour l'union, la concaténation et l'étoile de Kleene.

Proposition 4.2.1 : La grammaire $G = (\{S\}, \Sigma, P, S)$ où l'unique règle est $S \longrightarrow \sigma, \sigma \in \Sigma$, génère le langage $\{\sigma\}$. De même, si $P = \emptyset$, le langage généré est $\emptyset$.

Proposition 4.2.2 : L'ensemble des langages algébriques est stable pour l'union.

Démonstration : Soit $G_1 = (V_1, \Sigma, P_1, S_1)$ (resp. $G_2 = (V_2, \Sigma, P_2, S_2)$) une grammaire générant L_1 (resp. L_2). On peut supposer que $V_1 \cap V_2 = \emptyset$ et que $S \notin V_1 \cup V_2$. La grammaire $G = (\{S\} \cup P_1 \cup P_2, \Sigma, P, S)$ où P contient $P_1 \cup P_2$ est la règle $S \longrightarrow S_1/S_2$, génère exactement $L_1 \cup L_2$.

Proposition 4.2.3 : L'ensemble des langages algébriques est stable pour la concaténation.

Démonstration : Avec les mêmes natations que dans la preuve précédente, il suffit de considérer la règle supplémentaire $S \longrightarrow S_1 S_2$ pour générer le langage $L_1 L_2$.

Proposition 4.2.4 : L'ensemble des langages algébriques est stable pour l'étoile de Kleene.

Démonstration : Encore une fois en utilisant les mêmes natations, il suffit de considérer la grammaire $G = (\{S\} \cup V_1, \Sigma, P, S)$ où P contient P_1 et la règle supplémentaire $S \longrightarrow SS_1/\epsilon$ pour générer le langage L_1^*.

Exemple 4.2.5 : Soit la grammaire $G = (V, \Sigma, P, S)$ où $V = \{S, A, B\}, \Sigma = \{a, b\}$ et où les productions sont $S \longrightarrow aB/\epsilon, B \longrightarrow bS/bA, A \longrightarrow aA/\epsilon$. Le langage généré par G est exactement $\{\epsilon\} \cup (ab)^* ab (a)^*$ qui est régulier.

Définition 4.2.6 : Une grammaire hors contexte $G = (V, \Sigma, P, S)$ est régulière (à gauche) si toute production de G possède une des trois formes suivantes : $A \longrightarrow a, A \longrightarrow Ba, A \longrightarrow \epsilon$ où $A, B \in V$ et $a \in \Sigma$.

Une grammaire hors contexte $G = (V, \Sigma, P, S)$ est régulière (à droite) si toute production de G possède une des trois formes suivantes : $A \longrightarrow a, A \longrightarrow aB, A \longrightarrow \epsilon$ où $A, B \in V$ et $a \in \Sigma$.

Proposition 4.2.7 : Un langage est régulier si et seulement si il est généré par une grammaire régulière à gauche (resp. à droite).

4.3 Exercices

Exercice 4.3.1 :

Décrire le langage $L(G)$ engendré par la grammaire G dans les cas suivants :

1. $S \longrightarrow aS/bA$ et $A \longrightarrow bA/c$,
2. $S \longrightarrow aaS/aa$,
3. $S \longrightarrow aaS/a/b$,
4. $S \longrightarrow aS$ et $A \longrightarrow bS/a$.

Exercice 4.3.2 :

- Soit la grammaire $G = (\{S\}, \Sigma, P, S)$ où $\Sigma = \{a, b, c\}$ et les productions de P données par :

$S \longrightarrow aSa/aSb/bSa/bSb/c$.

1. Montrer que le mot $aabcbbb$ est engendré par G.

2. Démontrer par induction que $L(G) \subseteq \{ucv : u, v \in \{a, b, c\} \text{ et } |u| = |u|\}$.

3. Démontrer que le langage $\{ucv : u, v \in \{a, b, c\} \text{ et } |u| = |u|\}$ n'est pas rationnel.

Exercice 4.3.3 :

- On note $nbocc : \Sigma^* \times \Sigma \longrightarrow \mathbb{N}$ l'application qui à un mot w de Σ^* et une lettre σ de Σ associe le nombre d'occurences de σ dans w.

1. Soit $L_1 = \{w \in \{a, b\}^* : nbocc(w, a) = nbocc(w, b)\}$. Donner une grammaire algébrique G_1 engendrant L_1 et démontrer par induction que $L(G_1) = L_1$.

2. Soit la grammaire $G_2 = (\{S, T\}, \Sigma, P, S)$ où $\Sigma = \{a, b\}$ et les productions de P données par :

$S \longrightarrow aS/T, T \longrightarrow bT/\epsilon$. Donner un automate équivalent à G_2.

3. Déterminiser l'automate obtenu à la question précédente.

4. Déterminer une expression rationnelle dénotant $L_2 = L(G_2)$.

5. Montrer, en utilisant le lemme de l'étoile, que le langage $L_3 = \{a^n b^n : n \in \mathbb{N}\}$ n'est pas rationnel.

6. Trouver le langage $L = L_1 \cap L_2$. En déduire que L_1 n'est pas rationnel.

Exercice 4.3.4 :

Voici quatre grammaires, suivies de six propositions de langages. Rendez son langage à chacune des grammaires :

$$G_1 = (\{S, T, A, B\}, \{0, 1, 2\}, P_1, S), P_1 = \left\{ \begin{array}{c} S \longrightarrow 0S/0T \\ T \longrightarrow 1A/B2 \\ A \longrightarrow 2/T2 \\ B \longrightarrow 1/1T \end{array} \right\}.$$

$$G_2 = (\{S, T, A, B\}, \{0, 1, 2\}, P_2, S), P_2 = \left\{ \begin{array}{c} S \longrightarrow 0T/1S/2S \\ T \longrightarrow 0T/1A/2S \\ A \longrightarrow 0T/1S/2B \\ B \longrightarrow 0B/1B/2B/\epsilon \end{array} \right\}.$$

$$G_3 = (\{S,T\},\{0,1,2\},P_3,S), P_3 = \left\{ \begin{array}{c} S \longrightarrow 0ST/012 \\ 2T \longrightarrow T2 \\ 1T2 \longrightarrow 1122 \end{array} \right\}.$$

$$G_4 = (\{S\},\{0,1,2\},P_4,S), P_4 = \left\{ \begin{array}{c} S \longrightarrow 0/0S \\ S \longrightarrow 1SS/S1S/SS1 \\ S \longrightarrow 2SS/S2S/SS2 \end{array} \right\}.$$

En considèrant les langages suivants :

L_1 = ensemble des mots qui contiennent le facteur 012.

$L_2 = \{w \in \{0,1,2\}^* : |w|_0 > |w|_1 + |w|_2\}$.

L_3 = ensemble des mots qui ont autant de 0 que de 1 et que de 2.

$L_4 = \{0^n 1^m 2^m : n > 0, m > 0\}$.

L_5 = ensemble des mots qui contiennent pas le facteur 012.

$L_6 = \{0^n 1^n 2^n : n > 0\}$.

Chapitre 5

Calculabilité, Complexité des algorithmes

Introduction

Une question fondamentale de l'informatique théorique est de déterminer si un problème donné peut ou non être résolu au moyen d'un ordinateur i.e., par une procédure effective. En particulier se pose la question de déterminer si une fonction donnée à arguments et valeurs entiers peut être calculée au moyen d'un algorithme. Nous présentons dans cette section, une classe de fonctions de $\mathbb{N}^p$ à valeurs dans $\mathbb{N}$ pour lesquelles une telle procédure de calcul existe toujours.

Contenu

5.1. Fonctions primitives récursives.
5.2. Complexité d'un algorithme.
5.3. L'indécidabilité.
5.4. Exercices.

5.1 Fonctions primitives récursives

Définition 5.1.1 : Soit p un entier. On note $\mathcal{F}_p$ l'ensemble des applications de $\mathbb{N}^p$ dans $\mathbb{N}$. Dans le cas où $p = 0$, les éléments de $\mathcal{F}_0$ sont identifiés avec les éléments de $\mathbb{N}$. On pose de plus $\mathcal{F} = \bigcup_{p\in\mathbb{N}} \mathcal{F}_p$.

Définition 5.1.2 : Soient i et p deux entiers tels que $1 \leq i \leq p$. La fonction de projection $\mathrm{p}_{i,p}$ appartenant à $\mathcal{F}_p$ est définie par

$$\mathcal{P}_{i,p} : \mathbb{N}^p \longrightarrow \mathbb{N}, (x_1, ..., x_p) \longmapsto x_i.$$

Par exemple, $\mathcal{P}_{1,1} : \mathbb{N} \longrightarrow \mathbb{N}$ est la fonction identité.

Définition 5.1.3 : La fonction successeur σ appartenant à $\mathcal{F}_1$ est définie par $\sigma : \mathbb{N} \longrightarrow \mathbb{N}, x \longmapsto x + 1$.

Définition 5.1.4 : La fonction *zéro* **0** appartenant à $\mathcal{F}_0$ est simplement la constante 0, c'est une fonction sans argument qui a toujours la valeur 0.

Définition 5.1.5 : Soient $f_1, ..., f_n$ des fonctions de $\mathcal{F}_p$ et g une fonction de $\mathcal{F}_n$. La fonction composée $h = g(f_1, ..., f_n)$ est la fonction de $\mathcal{F}_p$ définie par :

$$h(x_1, ..., x_p) = g(f_1(x_1, ..., x_p), ..., f_n(x_1, ..., x_p)).$$

Définition 5.1.6 : Soient $g \in \mathcal{F}_p$ et $h \in \mathcal{F}_{p+2}$. On définit une fonction $f \in \mathcal{F}_{p+1}$ telle que pour tous $x_1, ..., x_p, n \in \mathbb{N}$

$$\begin{cases} f(x_1, ..., x_p, 0) = g(x_1, ..., x_p), \\ f(x_1, ..., x_p, n+1) = h(x_1, ..., x_p, n, f(x_1, ..., x_p, n)). \end{cases}$$

On dit que f est définie par récursion primitive à partir de g et de h.

Définition 5.1.6 : L'ensmble $\mathcal{PR}$ des fonctions primitives récursives est le plus petit sous ensemble de $\mathcal{F}$ qui

▶ Contient la fonction *zéro* **0**,

▶ Contient les fonctions de projection $\mathcal{P}_{i,p}$ quels que soient les entiers i et p tels que $1 \leq i \leq p$,

▶ Contient la fonction successeur σ,

► Est stable pour la composition, i.e., si n et p sont des entiers, $f_1, ..., f_n$ des fonctions de $\mathcal{F}_p$ qui appartiennent à $\mathcal{PR}$ et g une fonction de de $\mathcal{F}_n$ qui appartient à $\mathcal{PR}$, alors la fonction composée $g(f_1, ..., f_n)$ apparient encore à $\mathcal{PR}$,

► Est stable par récursion primitive, i.e., si p est un entier, g une fonction de $\mathcal{F}_p$ appartenant à $\mathcal{PR}$ et h une fonction de $\mathcal{F}_{p+2}$ appartenant à $\mathcal{PR}$, alors la fonction définie par récursion primitive à partir de g et de h appartient encore à $\mathcal{PR}$.

Exemples 5.1.7 : 1. Les fonctions constantes de $\mathcal{F}_1$ sont primitives récursives. En effet, la fonction constante $\mathbf{1}$ s'obtient en composant la fonctio *zéro* et la fonction successeur qui sont toutes les deux primitives récursive $\mathbf{1} = \sigma \circ \mathbf{0}$. D'une manière générale, la fonction constante $\mathbf{n+1}$ est la composée de la fonction successeur et de la fonction $\mathbf{n}$ qui, par hypothèse de récurrence est primitive récursive.

2. Les fonctions constantes de $\mathcal{F}_p$ sont primitives récursives. Soit n un naturel, on note $\mathbf{n}_p$ la fonction de $\mathcal{F}_p$ définie par :

$$\mathbf{n}_p : \mathbb{N}^p \longrightarrow \mathbb{N}, (x_1, ..., x_p) \longmapsto n.$$

Au vu de l'exemple précédent, nous savons déja que pour tout $n \in \mathbb{N}$, la fonction de $\mathcal{F}_1$ $\mathbf{n}$ est primitive récursive. Procédons par récurrence sur p. Supposons que $\mathbf{n}_p$ est une fonction primitive récursive et montrons que $\mathbf{n}_{p+1}$ l'est encore. Il est clair que

$$\begin{cases} \mathbf{n}_{p+1}(x_1, ..., x_p, 0) = \mathbf{n}_p(x_1, ..., x_p) \\ \mathbf{n}_{p+1}(x_1, ..., x_p, m+1) = \mathcal{P}_{p+2,p+2}(x_1, ..., x_p, m, \mathbf{n}_{p+1}(x_1, ..., x_p, m)). \end{cases}$$

Ainsi, $\mathbf{n}_{p+1}$ s'obtient par récursion primitive à partir des fonctions $\mathbf{n}_p$ et $\mathcal{P}_{p+2,p+2}$ qui sont toutes deux primitives récursives. Par conséquent, $\mathbf{n}_{p+1}$ appartient aussi à $\mathcal{PR}$.

3. Pour tout entier $p \geq 1$, la fonction d'addition $\Sigma_p : (x_1, ..., x_p) \longmapsto \sum_{i=1}^{p} x_i$ appartenant à $\mathcal{F}_p$ est primitive récursive. On procède par récurrence sur p. Pour $p = 1$, $\Sigma_1 = \mathcal{P}_{1,1}$ est une fonction primitive récursive. Supposons le résultat acquis pour p et vérifions-le pour $p+1$ en se ramenant au schéma de définition par récursion primitive :

$$\begin{cases} \Sigma_{p+1}(x_1, ..., x_p, 0) = \Sigma_p(x_1, ..., x_p) \\ \Sigma_{p+1}(x_1, ..., x_p, n+1) = \sigma(\mathcal{P}_{p+2,p+2}(x_1, ..., x_p, n, \Sigma_{p+1}(x_1, ..., x_p, n))). \end{cases}$$

Définition 5.1.8 : La fonction d'Ackermann $\mathcal{A} \in \mathcal{F}_2$ est définie par :

$$\begin{cases} \mathcal{A}(0,m) = m+1, \\ \mathcal{A}(m+1,0) = \mathcal{A}(m,1), \\ \mathcal{A}(m+1,n+1) = \mathcal{A}(m,\mathcal{A}(m+1,n)). \end{cases}$$

Théorème 5.1.9 : La fonction d'Ackermann n'est pas primitive récursive.

5.2 Complexité d'un algorithme

L'objet de la théorie de la complexité est :

• Pour les algorithmes d'évaluer le nombre d'opérations élémentaires (complexité temporelle) et l'espace mémoire nécessaire (complexité saptiale) pour leur résolution;

• Pour les problèmes (de décision), de les calssifier suivants leur niveau de difficulté.

Définition 5.2.1 : Soient f et g deux fonctions à variable entière et à valeurs positives.

$f(n) = O(g(n))$ s'il existe une constante $c > 0$ et un entier n_0 tels que
$\forall n \geq n_0, 0 \leq f(n) \leq cg(n)$. On dit que g domine f ou que f ne croit pas plus vite que g multiplié par une constante.

Définition 5.2. 2 : Soit n un entier, on note D les données de taille inférieur ou égale n, $C(D)$ le coût associé à l'exécution de la donnée D par un algorithme et $C(n)$ la complexité de l'algorithme. Alors on peut définir $C(n)$ de trois façons différentes:

1. Complexité dans le pire des cas**:** $C(n) = Max\{C(D)/D$ donnée de taille inférieur ou égale $n\}$.

2. Complexité en moyenne **:** $C(n) = \sum_{D \text{ donnée de taille inférieur ou égale } n} P(D)\,C(D)$.

où $P(D)$ est la probabilité d'obtenir la donnée D.

3. Complexité dans le meilleur des cas **:**

$C(n) = Min\{C(D)/D$ donnée de taille inférieur ou égale $n\}$.

Définition 5.2.3 : Un algorithme est à complexité polynômiale si sa complexité dans le pire des cas est de la forme $O\left(n^k\right)$ où k est une constante. Si sa complexité ne peut être majorée par un polynôme, on dit que l'algorithme est à complexité exponetielle**.** Un algorithme est à quasi exponetielle si sa complexité est une fonction de la forme $\exp^{O(n)}$.

5.3 L'indécidabilité

Un problème P de variable x est décidable s'il existe un algorithme qui pour chaque x dit "*oui*" ou "*non*" à la question : " Est-ce que $P(x)$ est vrai?". Ce problème avait été posé la première fois par David Hilbert**,** au Congré International des Mathématiciens qui avait eu lieu à Paris en 1900. L'indécidabilité est la négation de la décidabilité, ceci signifie qu'il ne peut exister d'algorithme se terminant toujours en un temps fini permettant de résoudre ce problème.

On introduit quelques problèmes indécidables :

Définition 5.3.1 : (Le problème de Poste)

Données : Deux suites finis X et Y de mots sur un alphabet $\Sigma, X = u_1, u_2, ..., u_n$ et $Y = v_1, v_2, ..., v_k$.

Le problème : Existe-t-il une suite $i_1, i_2, ..., i_m$ telle que $u_{i_1}u_{i_2}...u_{i_m} = v_{i_1}v_{i_2}...v_{i_m}$?.

Exemple 5.3.2 : Sur l'alphabet $\Sigma = \{0, 1\}$, le problème avec $X = 1, 10111, 10, Y = 111, 11, 011$ est décidable : $u_2u_1u_1u_3 = v_2v_1v_1v_3$. Mais le problème avec $X = 10, 011, 101, Y = 101, 11, 011$ n'a pas de solution.

Définition 5.3.3 : (Le problème du mot dans un monoïde libre (Thue))

Données : Etant donnés un monoïde Σ^* librement et finiment engendré par un ensemble Σ, et une congruence définie sur ce monoïde engendrée elle-même par une relation finie $\mathcal{R}$.

Le problème : Le problème du mot consiste à reconnaitre si deux éléments du quotient, définis par deux représentants dans Σ^*, sont distincts ou non.

Remarque 5.3.4 : On peut chercher des conditions suffisntes pour que ce problème devienne décidable, cest le cas si dans toute classe d'équivalence on sait trouver un représentant canonique. Un idée simple est de prendre comme représentant un mot de plus courte longueur dans sa classe. Une des méthodes de décision consiste à associer à $\mathcal{R}$ un semi-système $(\Sigma, \mathcal{R})$ de règles de réécriture, tel que deux mots $w_1$et w_2 sont équivalents modulos $\overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}}$ si, et seulement si il existe une suite de réécritures conduisant w_1 et w_2 sur le même mot. Dans le cas où on peut trouver un tel semi-système complet et fini, il est alors claire que le problème du mot est résoluble.

Exemple 5.3.5 : Soit $\Sigma = \{\alpha\}$, et $\mathcal{R} = \{\alpha^2 \longrightarrow \epsilon\}$, ϵ est le mot vide. $\forall w \in \Sigma^*$, on distingue deux cas :

- Si $|w|$ est paire alors $w \overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}} \epsilon$.
- Si $|w|$ est impaire alors $w \overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}} \alpha$.

Finalement $\Sigma^* / \overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}} = \left\{ [\alpha]_{\overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}}}, [\epsilon]_{\overset{*}{\underset{\mathcal{R}}{\Longleftrightarrow}}} \right\}$ et le problème du mot dans cet exemple est résoluble.

Définition 5.3.6 : (Le problème du sac à dos (Hellman))

Données : Une suite d'entiers positifs $a_1, a_2, ..., a_n$ et S.

Le problème : Existe-t-il un sous ensemble $K \subseteq I = \{1, 2, ..., n\}$ tel que $\sum_{k \in K} a_k = S$.

Exemple 5.3.7 : Hellman a démontré que 516 ne peut s'écrire comme somme des entiers de $\{14, 28, 56, 82, 90, 132, 197, 284, 341, 455\}$. 515 peut s'écrire de 3 façons différentes à l'aide de ces entiers.

Proposition 5.3.8 : Soit $(a_i)_{1 \leq i \leq n}$ une suite d'entiers positifs. Si la suite $(a_i)_{1 \leq i \leq n}$ est supercroissante, c'est-à-dire $a_i > \sum_{k<i} a_k$ $(i > 1)$, alors le problème du sac à dos est facile à résoudre.

Preuve :

Soit a_m le plus grand élément de $(a_i)_{1 \leq i \leq n}$ inférieur ou égale à S. Alors $m+1, m+2$ ....ne sont pas dans K car sinon la somme $\sum_{k \in K} a_k$ serait supérieur à S. m appartient à K car sinon la somme $\sum_{k \in K} a_k$ serait inférieur à S. On recommence avec $S - a_m$ à la place de S.

5.4 Exercices

Exercice 5.4.1 :

- Vérifier que les fonctions suivantes sont primitives récursives :

1. Pour tout entier $p \geq 1$ la fonction produit définie par $(x_1, ..., x_p) \longmapsto \prod_{i=1}^{p} x_i$.
2. La fonction puissance de $\mathcal{F}_p$ définie par $p : (x, n) \longmapsto x^n$.
3. La fonction factorielle $n!$.

Bibliographie

[1] J. M. Autebert. "Théorie des langages et automates", Masson, (1994).

[2] M. Benois. "Application de l'étude de certaines congruences à un problème de décidabilité", Séminaire Dubreil , n^o 7, (1972).

[3] P. Berlioux , Mnacho. Echenim et Michel Lévy. "Théorie des langages", Ecole nationale supérieure d'informatique et de mathématiques appliquées de France, (2009).

[4] J. Berstel. "Automates et grammaires", Université de Marne-la-Vallée, (2005).

[5] J. Berstel. "Theory of codes", Academic Press, (1984).

[6] J. Berstel. "Congruences plus que parfaites et langages algébriques", Séminaire d'informatique théorique, (1977).

[7] R. V. Book and H. N. Liu. "Rewriting Systems and Word Problems in a Free Partially Commutative Monoid", Information Processing Letters n^o 26, pp. 29-32, (1987).

[8] T. Bourdier. "Mathématiques Discrètes 1 et Informatique Théorique, Ecole Supiérieure d'informatique et Applications de Lorraine.

[9] R. Cori et D. Perrin. "Automates et Commutations Partielles", RAIRO-Informatique théorique, tome19, n^o 1, pp. 21-32, (1985).

[10] N. Dershowitz. "Termination of rewriting", Journal of Symbolic Computation, tome 3, pp. 69-116(1987).

[11] H. Dubois. "Systèmes de règles de production et calcul de réécriture", Thèse de doctorat en informatique, Université Henri Poincaré-Nancy 1, (2001).

[12] S. Eilenberg. "Automata languages and machines. Vol A, Académic Press, New-York, (1974).

[13] M. Eytan et G. TH. Guilbaud. "Présentation de quelques monoïdes finis", Mathématiques et sciences humaines, vol 7, pp. 3-10, (1964).

[14] R. Floyd et R. Beigel, Traduction de D. Krob. "Le langage des machines", International Thomson France, paris, (1995).

[15] N. Ghadbane and D. Mihoubi."On the termination problem for string rewrite systems ", International Journal of Computer Applications, Vol. 146, n^o 6, (2016).

[16] N. Ghadbane and D. Mihoubi. "A Construction of Some Group Codes, International Journal of Electronics and Information Engineering, Vol. 4, (2016).

[17] N. Ghadbane. "A Construction and Representation of some Variable length Codes, Annals Computer Science Series Journal, Vol. 15, n^o 2, (2017).

[18] N. Ghadbane, "Etude sur les groupes syntaxiques de petits degrés, Mémoire de Magister, Université de M'sila, (2010).

[19] D. Guin et T. Hausberger. "Algèbre 1, Groupes, Corps et Théorie de Galois", EDP Sciences, (2008).

[20] Y. Guiraud. "Présentations d'opérades et systèmes de réécriture", Thèse de Doctorat, Institut de Mathématiques et de Modélisation de Montpellier, (2004).

[21] A. K. "Knuth-Bendix procedure and Buchberger Algorithm A Synthesis", Université Cadi Ayyad , Marrakech, Morocco, (1989).

[22] D. Kapur and P. Narendran. "A Finite Thue System With Decidable Word Problem and Without Equivalent Finite Canonical System", Theoretical Computer Science, vol 35, pp. 337-344, (1985).

[23] Y. Lafont. "Réécritue et problème du mot", Gazette des Mathématiciens 120. Laboratoire de Mathématiques Discrètes de Luminy, Marseille, France, (2009).

[24] Y. Lafont. "A new finiteness condition for monoids presented by complete rewrting systms", Laboratoire de Mathématiques Discrètes de Luminy, Marseille, France, (1995).

[25] S. Lombardy. "Approche structurelle de quelques problèmes de la théorie des automates, Thèse de Doctorat, Ecole doctorale d'Informatique, Télécommunications et d'Electronique de Paris, (2001).

[26] S. Marcel."Langage formels et monoïdes finis", Séminaire Dubreil. Algèbre et théorie des nombres, vol. 23, no. 2, pp. 1-3, (1970).

[27] M. Marchand. "Outils mathématiques pour l'informaticien", De Boeck, (2005).

[28] Y. Metiver. "Calcul de longueurs de chaines de réécriture dans un monoïde libre", U.E.R, de Mathématiques et informatique, Université de Bordeaux 1, France (1983).

[29] D. Mihoubi. "Modes de Reconnaissance et Equités dans les ω-Automates a Pic", Thèse de doctorat, Université Paris-Nord, (1989).

[30] M. Nivat. "Sur le noyau d'un homomorphisme du monoïde libre", Séminaire Schutzenberger, tom 1, exp. n^o 4, pp. 1-6, (1970).

[31] M. Nivat. "Congruence parfaites et quasi-parfaites", Séminaire Dubreil Algèbre, tom 1, exp, n^o 7, pp. 1-9, (1970).

[32] M. Nivat. "Eléments de la théorie général des codes, Université de Paris, (1965-1966).

[33] H. Phan et Philippe Guillot. "Preuves de sécurité des schémas cryptographiques", université Paris 8, (2013).

[34] D. Perrin. "Le degré minimal du groupe d'un code bipréfixe fini", Journal of Combinatorial Theory, Series A, vol. 25, no. 2, pp. 749-759, (2003).

[35] L. Pierre. "Les systèmes de réécriture", Université de Nice Sophia-Antipolis,(2007).

[36] P. Rannou. "Réécriture de diagrammes et de Σ-diagrammes", Thèse de doctorat, Université d'Aix-Marseille, (2013).

[37] M. Rigo. "Théorie des automates et langages formels", Université de Liège, 2009.

[38] G. Rindone. "On syntactic groups", Bulletin Mathématique de Belgique, (2003-2004).

[39] B. Robin. "Correspondances entre les algorithmes de Knuth-Bendix et de Buchberger", Université de Claude Bernard Lyon.

[40] J. Rouyer. "Preuves de terminaison de systèmes de réécriture fondées sur les interprétations polynomiales", CRIN, Nancy, (1989).

[41] A. Salomma. "Jewels of formal language theory", University of Turku, (1977).

[42] S. Julia. "Automates et Langages", (2015/2016).

[43] H. J. Shyr. "Free monoids and languages", Department of Mathematicsn, Soochow University Taipei,Taiwan R.O.C,(1979).

[44] F. Yvon et A. Demaille, "Théorie des Langages Rationnels, (2010)

[45] H. Zantema. "Termination of String Rewriting Proved Automatically, TU Eindhoven, (2004).