REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

Université Mohamed Boudiaf de M'sila

Faculté des Mathématiques et de l'Informatique

Département des Mathématiques

Mémoire de Master

Domaine : Mathématiques et Informatique
Filière : Mathématiques
Option : Algèbre et Mathématiques Discrètes

Thème

Les types de produit de monoïdes et leur applications aux semi automates

---

Présentée par :
RAYANE Somia

Devant le jury composé de :

MIHOUBI Douadi	Prof. Université de Msila	**Président.**
GHADBANE Nacer	MCA. Université de Msila	**Encadreur.**
HEBOUB Lakhdar	MAA. Université de Msila	**Examinateur.**

Année universitaire 2019/2020

Dédicace

À celui qui m’a appris le sens de la vie et qui était avec moi dans mes premiers pas à ceux qui étaient impatients d’atteindre ce jour lors de ma remise de diplôme à mon père,qui Dieu ait pitié de lui et de ma chère mère.

A tous ma famille,

A tous mes amies,

Je dédie ce mémoire.

Remerciements

Je remercie Dieu qui m'a aidé à faire ce travail, et j'ai eu la chance d'avoir le succès de Dieu Tout-Puissant.

Quand après je vous remercie, mon professeur encadré :Mr.N.GHADBANE pour sa coopération et son assistance,que Dieu le récompense le mieux.

Je remercie également le jury pour son évaluation de mon humble travail,aussi que pour tous les conseils qu'il ont prodigués.

Et je remercie également tous les professeurs de la faculté des mathématiques et de l'informatique pour ce qu'ils fournissent afin de transrmettre pleinement des information précieusses à tous les étudiant.

Enfin,je remercie tous ceux qui ont contribué à la réalisation de ce travail de près ou de loin.

Les types de produit de monoïdes et leur applications aux semi automates

Rayane Somia

2019/2020

Table des matières

Notations

Σ : alphabet fini.
Σ^* : monoïde libre.
$|w|$: la langueur du mot w.
$|w|_a$: le nombre d'occurence de la lettre a dans le mot w.
$\equiv$: relation de congruence.
$\equiv_E$: congruence syntaxique.
$M/\equiv$: monoïde quotient.
L : langage sur alphabet Σ.
L^* : l'étoile de kleene de L.
$Rat(\Sigma^*)$: la famille des langages rationelles.
$S = (Q, \Sigma, \delta)$: semi automate.
$A = (Q, I, F, \Sigma, \delta)$: un automate.
AFD : automate fini déterministe.
$\mathcal{L}(A)$: le langage reconnu par un automate A.
$\operatorname{Re} c(A)$: l'ensemble des langages reconnaissables sur A.
A_L : l'automate minimale d'un langage L.
$M(L)$: monoïde syntaxique.
$\zeta(A)$: monoïde de transitions.
M^N : l'esemble de touts les applications de N dans M.
$M \rtimes_\theta N$: produit en couronne des monoïdes par rapport à θ.
SWS' : produit en couronne des semi automates.

Introduction

La théorie des automates qui s'est développée ces quatre-vingt dix dernières années, a eu une influence considérable, non seulement sur les systèmes informatique, mais aussi sur la biologie, la biochimie, etc. La richesse et la diversité des idées est surprenante, surtout si l'on accepte de considérer le sujet au sens large avec des liens allât de la linguistique à l'algèbre, en passant par l'électronique et l'informatique.

Dans ce mémoire, nous allons étudier les monoïdes, semi-automates et la décomposition d'un automate fini. Plus précisément : monoïde syntaxique, monoïde de transitions, produit direct, semi direct des monoïdes, et produit de semi-automates.

Ce travail est composé de trois chapitres :

Le premier chapitre, consiste à un rappel des notions élémentaires sur les monoïdes et les semi-automates.

Dans le second chapitre, nous allons étudier le monoïde syntaxique et monoïde de transitions ainsi que certaines propriétés, aussi on présente la relation entre les deux.

Dans le troisième chapitre, on fait une étude sur les types des produits des monoïdes et leurs applications en semi-automates.

Chapitre 1

Notions élémentaires sur les monoïdes, semi-automates.

Ce premier chapiter contient les définitions et les propriétés des outils que nous utiliserons par la suite : monoïdes, langages et semi-automates.

1.1 Généralités sur les monoïdes

Définition 1.1

Un monoïde est un ensemble muni d'une loi intérne, i.e, d'une application $\cdot : M \times M \to M$, qui satisfait aux conditions suivantes :

- L'opération " $\cdot$ " est associative :

$$\forall x, y, z \in M, x \cdot (y \cdot z) = (x \cdot y) \cdot z\,;$$

- Il existe un élément neutre (unique) $1_M \in M$ tel que :

$$\forall x \in M, x \cdot 1_M = 1_M \cdot x = x.$$

Un monoïde M est dit simplifiable à gauche, ou encore régulier à gauche, (respectivement à droite) si :$\forall (x, y, z) \in M^3, x \cdot y = x \cdot z \Longrightarrow y = z$ (respectivement $y \cdot x = z \cdot x) \Longrightarrow y = z$).

Exemple 1.1

1) $(\mathbb{N}, +, 0), (\mathbb{R}, \times, 1)$ et $(\mathbb{N} \cup \{+\infty\}, \min, +\infty)$ sont des monoïdes, où $+$ et $\times$ dénotent respectivement l'addition et la multiplication usuelles.

2) L'ensemble des parties d'un ensemble E, muni de l'union d'ensembles $(P(E), \cup)$ est un monoïde dont l'ensemble vide $\emptyset$ est l'élément neutre. Le même ensemble muni de l'intersection d'ensembles $(P(E), \cap)$ est un monoïde, dont l'ensemble E est l'élément neutre.

3) L'ensemble des application d'un ensemble Q vers lui même $Q^Q = \{f : Q \to Q\}$ muni de la composition des applications $(Q^Q, \circ)$ est un monoïde d'élément neutre l'application identité id_Q.

Proprietés d'un monoïde

1. Un élément $m' \in M$ est dit le symétrique de l'élément $m \in M$ si $m \cdot m' = 1_M$.
2. Un monoïde est dit commutatif si ses éléments sont permutables, c'est à dire si : $\forall (x, y) \in M^2, x \cdot y = y \cdot x$.

Remarque 1.1

Un monoïde $(M, \cdot, 1_M)$ qui est tel que tout élément de M possède un symétrique est un groupe.

Définition 1.2

Soit un monoïde $(M, \cdot, 1_M)$. Un sous monoïde est un triplet $(M', \cdot, 1_{M'})$ tels que :

1. $M' \subseteq M$;
2. $1_M = 1_{M'}$;
3. $\forall x, y \in M', x \cdot y \in M'$.

Proposition 1.1

Tout intersection de sous monoïde est un sous monoïde.

Exemple 1.2

1. L'ensemble $2\mathbb{Z}$ des entiers paires est un sous monoïde de $(\mathbb{Z}, +)$.
2. Pour tout monoïde $(M, \cdot, 1_M)$, les parties $\{1_M\}$ et M sont des sous monoïdes dit "triviaux".

Définition 1.3

Soit Σ un alphabet fini, les éléments de Σ sont appelés lettres ou symboles. Un mot sur Σ est une suite finie des lettres. La longueur d'un mot w est le nombre de lettres constituant ce mot, qu'on note $|w|$.
L'unique mot de longueur 0 est le mot correspondant à la suite vide, ce mot s'appelle le mot vide on le note ε. On note par Σ^* le monoïde libre engendré par Σ, d'élément neutre le mot vide ε, muni de l'opération concaténation des mots.

Exemple 1.3

Pour $\Sigma = \{a, b\}, \Sigma^* = \{\varepsilon, a, b, aa, ab, ba.....\}$.

$w = ab$ et $w^2 = abab$ sont des mots sur $\Sigma = \{a, b\}$.

Si $w = ab$, alors $|w|_a = |w|_b = 1$.

Définition 1.4

La concaténation de deux mots $u = u_1...u_m$ et $v = v_1...v_n$ est le mot noté $u \cdot v$ ou uv et égal à $u_1...u_m v_1...v_n$ obtenu par simple juxtaposition. C'est une opération associative dont le mot vide ε est l'élément neutre . C'est pour cette raison que le mot vide est parfois noté 1.

Exemple 1.4

Si $u = abaa$ et $v = bab$, on a $uv = abaabab$ et $vu = bababaa$. La concaténation n'est pas commutative.

Proprietés

Soit Σ un alphabet quelconque. Le monoïde Σ^* possède les deux propriétés suivantes :

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

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

Proposition 1.2

Soit Σ un alphabet.

1. L'ensemble Σ^* est infini.

2. L'ensemble Σ^* est dénombrable.

Preuve.

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 example, 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 example si $|u| = 5$, on a les 5 premiers nombres 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 $u = \alpha\beta\gamma\alpha\alpha$, 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 1.5

Si $(M, \cdot)$ et (N, Δ) sont deux monoïdes, avec neutres 1_M et 1_N, respectivement, alors la fonction $f : M \to N$ est un morphisme de monoïdes de $(M, \cdot)$ vers (N, Δ) si :

i) $f(x \cdot y) = f(x) \Delta f(y)$ pour tout $x, y \in M$;

ii) $f(1_M) = 1_N$.

Remarque 1.2

Un isomorphisme de monoïdes est simplement un morphisme de monoïde bijective.

Exemple 1.5

• L'application longueur $|\cdot| : \Sigma^* \to \mathbb{N}$ est un morphisme de monoïdes entre $(\Sigma^*, \cdot, \varepsilon)$ et $(\mathbb{N}, +, 0)$. En effet, $\forall u, v \in \Sigma^*, |u \cdot v| = |u| + |v|$ et $|\varepsilon| = 0$.

• Considérons l'alphabet $\Sigma = \{a, b, c\}$ et le morphisme $\varphi : \Sigma^* \to \Sigma^*$ défini par $\varphi(a) = abc, \varphi(b) = ac, \varphi(c) = b$. En effet, pour définir un tel morphisme, on remarquera qu'il suffit de se donner l'image des lettres. On a, par exemple,

$\varphi(abbc) = \varphi(a)\varphi(b)\varphi(b)\varphi(c) = abcacacb.$

Proposition 1.3

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

Preuve. L'existence : Posons
$\widetilde{\mu}(\varepsilon) = 1_M$ et $\widetilde{\mu}(\alpha_1 ... \alpha_n) = \mu(\alpha_1)...\mu(\alpha_n), n \in \mathbb{N}, \alpha_i \in \Sigma, 1 \leq i \leq n.$
Il est facile de voir que $\widetilde{\mu}$ est bien homomorphisme.
L'unicité : Si $\widetilde{\mu}$ et $\widetilde{\lambda}$ sont deux homomorphisme de Σ^*dans M tels que :
$\forall \alpha \in \Sigma, \widetilde{\mu}(\alpha) = \widetilde{\lambda}(\alpha)$, alors $\widetilde{\mu}(\varepsilon) = \widetilde{\lambda}(\varepsilon) = 1_M$ et pour tout mot $w = \alpha_1 ... \alpha_n \in \Sigma^*$, on a $\widetilde{\mu}(w) = \widetilde{\mu}(\alpha_1 ... \alpha_n) = \mu(\alpha_1)...\mu(\alpha_n) = \widetilde{\lambda}(\alpha_1 ... \alpha_n) = \widetilde{\lambda}(w)$. ■

Définition 1.6

Soit M un monoïde où l'opération interne est implicite. On rappelle qu'une relation d'équivalence $\equiv$ sur M est une relation de congruence si :

$(\forall u_1, u_2, v_1, v_2 \in M)(u_1 \equiv v_1 \wedge u_2 \equiv v_2 \Longrightarrow u_1 u_2 \equiv v_1 v_2).$

Pour un sous-ensemble E de M, on définit la relation de congrunce syntaxique de E sur M, notée $\equiv_E$, par :

$x \equiv_E y \Longleftrightarrow (\forall u, v \in M)[uxv \in E \Leftrightarrow uyv \in E].$

Exemple 1.6

Le monoïde syntaxique de L_{paire} consiste en classe des mots de taille paire, et la classe des mots de taille impaire, la premiére classe étant l'identité du monoïde.

L_{paire} : Le langage des mots de longueur paire sur $\{a, b\}$.

Définition 1.7

La classe de congruence contenant l'élément $m \in M$ est l'ensemble

$[m] = \{x \in M : x \equiv m\}.$

Si $\equiv$ est une relation de congruence sur le monoïde $(M, \cdot)$, l'ensemble de quotient $M/\equiv\, = \{[m] : m \in M\}$ est un monoïde appelé le monoïde quotient de M sur $\equiv$.

Exemple 1.7

Soit le monoïde $(\mathbb{N}, +)$ et soit la relation $\equiv$ définie par $x \equiv y$ si, et seulement si, x et y ont même partié. 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 1.8

Soit $\equiv$ une congruence sur un monoïde M.

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

Définition 1.9

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 $y^{-1}X = \bigcup_{x \in X} y^{-1}x$.

1.2 Langages

Définition 1.10

Soit Σ un alphabet. On appelle langage formel tout sous-ensemble $L \subseteq \Sigma^*$.

Remarque 1.3

Σ^* est le plus grand langage sur Σ au sens de l'inclusion.

Exemple 1.8

1. $L^0 = \{\varepsilon\}$ est le langage contenant le seul mot ε.

2. $L = \emptyset$ est le langage vide.

3. Considérons l'alphabet $\Sigma = \{a, b, c\}$. L'ensemble $L = \{\varepsilon, a, aa, bbc, ccca, ababab\}$ est un langage fini.

4. L'ensemble L_{2a} des mots sur Σ comprenant un nombre pair de a est aussi un langage (infini), $L_{2a} = \{\varepsilon, b, c, aa, bb, cb, cb, cc, aab, aac, aba, aca, ..., abaacaaa, ...\}$.

Définition 1.11

Soient $L_1 \subseteq \Sigma^*$ et $L_2 \subseteq \Sigma^*$. On appelle concaténé des deux langages le langage : $L_1 \cdot L_2 = \{x.y, x \in L_1 \wedge y \in L_2\}$.

Définition 1.12

Pour tout $L \subseteq \Sigma^*$, L^+ dénote le langage :

$L^+ = L \cup L^2 \cup ... \cup L^n \cup ...$

De plus, on définit l'étoile (dite de Kleene) de L par :

$L^* = \bigcup_{n \geq 0} L^n = L^+ \cup \{\varepsilon\}$.

Exemple 1.9

Avec $\Sigma = \{a, b\}$, posons $L_1 = \{a, b\}$ et $L_2 = \{ab\}$, on a alors $L_1 L_2 = \{aab, bab\}$ et $L_2^* = \{\epsilon, ab, abab, ...\}$.

Proposition 1.4

Soient $L_1 \subseteq \Sigma^*$ et $L_2 \subseteq \Sigma^*$, et leur concaténé $L_1 \cdot L_2$.

- $L_1 \cdot L_2 \subseteq \Sigma^*$.
- $L^* \subseteq \Sigma^*$ pour tout monoïde $L^* = \langle L, \cdot \rangle$.
- $L_1 \cdot (L_2 \cup L_3) = L_1 \cdot L_2 \cup L_1 \cdot L_3$ et $(L_2 \cup L_3) \cdot L_1 = L_2 \cdot L_1 \cup L_3 \cdot L_1$ (distributivité de la concaténation par rapport à l'union).
- $L^n = \{w_1 w_2 ... w_i, i \in \{1, 2...n\}, w_i \in L\}$ la puissance n-iéme de langage $L, n > 0$.

Définition 1.13

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

Remarque 1.4

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

Exemple 1.10

1 $\{\varepsilon\}$ est un langage rationnel, car on a $\{\varepsilon\} = \emptyset^*$.

2 Σ est rationnel : $\Sigma = \bigcup\limits_{\sigma \in \Sigma} \{\sigma\}$.

3 Σ^* est rationnel.

4 Si $w \in \Sigma^*$, $\{w\}$ est rationnel. Si $w = \sigma_1 ... \sigma_n$, on a $\{w\} = \{\sigma_1\} ... \{\sigma_n\}$.

5 Tout langage fini est rationnel. En effet, si L est fini, on a $L = \bigcup\limits_{w \in L} \{w\}$.

Proposition 1.5

Le produit de deux langages rationnels est un langage rationel.

Définition 1.14

Soit Σ un alphabet fini, $h : \Sigma^* \longrightarrow M$ un morphisme de monoïdes et $L \subset \Sigma^*$ un langage. On dit que h reconnaîte L s'il existe une partie P de M telle que $L = h^{-1}(P)$. Par extension, on dit également dans ce cas M reconnaîte par L.

Définition 1.15

Un langage est dit reconnaissable s'il est reconnu par un monoïde fini.

1.3 Les semi -automates

Définition 1.16

Un semi automate est un triplet $S = (Q, \Sigma, \delta)$, où Σ et Q sont des ensembles finis et δ est une fonction de $Q \times \Sigma$ vers Q. Un semi automate $S = (Q, \Sigma, \delta)$ est dit complet si δ est une application.

Définition 1.17

Un semi-automate est souvent décrit au moyen d'une table (également appelée table d'états suivante ou table de transition).

Encore une autre représentation des semi automates est extrêmement utile. Il utilise un graphe orienté. Les sommets du graphique représentent les étates de S et sont représentés par des cercles, et pour tout $\binom{q_i}{q_j} \in \delta_{\sigma_i}$, une flèche étiquetée σ_i mène de q_i à q_j, $q_i \xrightarrow{\sigma i} q_j$.

Exemple 1.11

1. Soit $S = (Q, \Sigma, \delta)$ un semi-automate avec $Q = \{p, q, r\}; \Sigma = \{a, b, c\}$ et δ est définie comme suit :

δ	a	b	c
p	q	q	r
q	p	p	ϕ
r	q	r	ϕ

2. $Q = \{1, 2, 3\}, \Sigma = \{a, b\}$

$1.a = 2$ $\quad$ $2.a = 3$ $\quad$ $3.a = 3.$

$1.b = 1$ $\quad$ $2.b = 1$ $\quad$ $3.b = 3.$

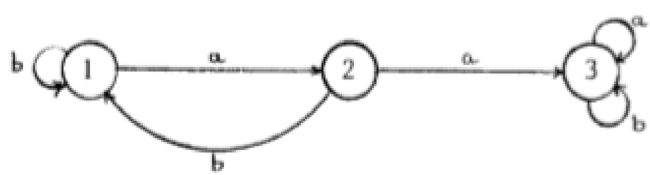

FIGURE V.1 Réprésentation d'un semi automate.

Remarque 1.5

La fonction de transition peut être étendue naturellement aux secondes de symboles d'entrée, en laissant

$\delta(q, w\sigma) = \delta(\delta(q, w), \sigma)$ et $\delta(q, \varepsilon) = q$, pour tout $w \in \Sigma^*, \sigma \in \Sigma$ et $q \in Q$.

Définition 1.18

S semi automate $S' = (Q', \Sigma', \delta')$ est un sous semi automate de la semi automate $S = (Q, \Sigma, \delta)$ si : $Q' \subseteq Q, \Sigma' \subseteq \Sigma$, et $\sigma_i' \subseteq \sigma_i$ pour tout $\sigma_i \in \Sigma'$.

Définition 1.19

Soit $S = (Q, \Sigma, \delta)$ est un semi automate complet, pour tout $\sigma \in \Sigma$, on définit l'application $\delta_\sigma : Q \to Q$ par $\delta_\sigma(q) = \delta(q, \sigma)$, et si $w \in \Sigma^*$ avec $w = \sigma_1 ... \sigma_n$, alors $\delta_w(q) = (\delta_{\sigma_n} \circ ... \delta_{\sigma_1})(q)$.

L'application $h : (\Sigma^*, \cdot) \to (Q^Q, \circ)$ définie par $h(w) = \delta_w$, est un morphisme de monoïdes.

La relation $\Re$ définie sur Σ^* par : $w \Re w' \Longleftrightarrow h(w) = h(w')$ est une congruence.

Le monoïde quotient $(\Sigma^*/\Re, \odot)$ est dit le monoïde de semi automate $S = (Q, \Sigma, \delta)$ avec " $\odot$ " est le produit de concaténation des classe.

Remarque 1.6

$(Q^Q, \circ)$ est le monoïde de toutes les fonctions de Q vers Q muni de la composition des applications, et d'élément de l'unité la fonction identité notée id_Q.

Définition 1.20

On appelle transition vide ou ε-transition toute transition d'étiquette ε. Cette transition joue un rôle particulier. En effet, si l'on essaie de comprendre le fonctionnement d'un automate, on peut associer à toute transition $q_i \xrightarrow{\sigma_i} q_j$ l'assertion suivante $\ll$ si l'on est en q_i et que l'on lit un σ_i, on arrive en q_j $\gg$. Une ε-transition $q_i \xrightarrow{\varepsilon} q_j$ est donc une transition qui permet de passer de l'état q_i à l'état q_j sans avoir besoin de lire une lettre.

Définition 1.21

Soit $S = (Q, \Sigma, \delta)$ et $S' = (Q', \Sigma', \delta')$ deux semi-automates, un morphisme d'automates est une surjection $\phi : Q \longrightarrow Q'$ tel que pour tout $q \in Q$ et $\sigma \in \Sigma$,

$\phi(\delta(q, \sigma)) = \delta'(\phi(q), \sigma)$.

Définition 1.22

Un automate est un 5-uplet $A = (Q, I, F, \Sigma, \delta)$ où,

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

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

Exemple 1.12

Soit $A = (Q, I, F, \Sigma, \delta)$ un automate défini par :

$\Sigma = \{a, b\}, Q = \{0, 1, 2\}, I = \{0\}, F = \{1, 2\}, \delta = \{(0, a, 1), (1, a, 1), (1, b, 2)\}$.

Remarque 1.7

Un automate A est dit fini si l'ensemble des états Q est fini.

Un automate fini $A = (Q, I, F, \Sigma, \delta)$ est un automate fini déterministe et l'on note AFD si δ est une fonction et $|I| = 1$.

Proposition 1.6

Tout automate est équivalent à un automate déterministe.

Définition 1.23

Soit $A = (Q, I, F, \Sigma, \delta)$ un automate fini déterministe la fonction $\delta : Q \times \Sigma \longrightarrow Q$ se prolonge en une fonction $\delta^* : Q \times \Sigma^* \longrightarrow Q$ définie par :

1. $\forall q \in Q, \delta^*(q, \varepsilon) = q$;
2. $\forall q \in Q, \forall \alpha \in \Sigma^*, \forall a \in \Sigma, \delta^*(q, a.\alpha) = \delta^*(\delta(q, a), \alpha)$.

Exemple 1.13

Si l'on reprend l'automate défini plus haut, on peut calculer l'image de $(q_0, abab)$ par δ^* :

$\delta^*(q_0, abab) = \delta^*(\delta(q_0, a), bab) = \delta^*(q_1, bab) = \delta^*(\delta(q_1, b), ab)$

$= \delta^*(q_2, ab) = \delta^*(\delta(q_2, a), b) = \delta^*(q_3, b)$

$= \delta^*(\delta(q_3, b), \varepsilon) = \delta^*(q_2, \varepsilon) = q_2$.

Définition 1.24

Le langage accepté (reconnu) par A : $\mathcal{L}(A) = \{u \in \Sigma^* | \delta(i, u) \in F\}$.

Un langage $L \subseteq \Sigma^*$ est reconnaissable, s'il existe un automate fini A tel que $L = \mathcal{L}(A)$.

On note $Rec(\Sigma^*)$ la famille des langages reconnaissables sur Σ^*.

Exemple 1.14

En prenant $q_0 = 1$ et $F = \{3\}$, on voit que l'automate de l'example 1.11(2) reconnaît le langage $L = \Sigma^* aa \Sigma^*$.

Définition 1.25

Soit L un langage reconnu par un automate fini déterministe $A = (Q, I, F, \Sigma, \delta)$. Pour tout couple d'état $(p, q) \in Q^2$, on définit l'ensemble :

$E_{p,q} = \{\alpha \in \Sigma^*, \delta^*(p, \alpha) = q\}$

ainsi que l'ensemble L_p, défini pour tout état $p \in Q$:

$L_p = \{\alpha \in \Sigma^*, \exists q \in F, \delta^*(p, \alpha) = q\}$.

Notation 1.1

L_p décrit le langage reconnu par le sous-automate $A_p = (Q, \Sigma, \{p\}, E, F)$.

Remarque 1.8

Pour tout état $p \in Q$, on note $\gamma_{p,F} = \varepsilon$ si $p \in F$ et $\emptyset$ sinon, on a

$L_p = \sum\limits_{q \in Q} E_{p,q}.L_q + \gamma_{p,F}$.

Exemple 1.15

Le langage des mots de longueur paire sur $\{a, b\}$ est rationnel.

Il s'exprime comme $L_{pair} = ((\{a\} \cup \{b\})^2)^*$.

L'automate suivante reconnaît L_{pair} :

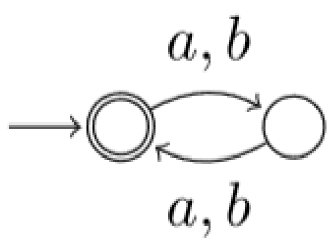

FIGURE V.2 Automate qui reconnaît L.

Définition 1.26

On définit l'automate minimal $A_L = (Q_L, q_{0,L}, F_L, \Sigma, \delta_L)$ d'un langage $L \subseteq \Sigma^*$comme suit :

- $Q_L = \{w^{-1}.L | w \in \Sigma^*\}$,
- $q_{0,L} = \varepsilon^{-1}.L = L$,

- $F_L = \{w^{-1}.L | w \in L\} = \{q \in Q | \varepsilon \in q\}$,
- $\delta_{L(q,\sigma)}) = \sigma^{-1}.q$,pour tous $q \in Q_L, \sigma \in \Sigma$.

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

Exemple 1.16

Soit $\Sigma = \{a, b\}, Q = \{1, 2, 3, 4\}$. On a le langage $L = (aa + ab + abb + aab + bb)^*$. L'automate minimal du langage L est donné ci dessous

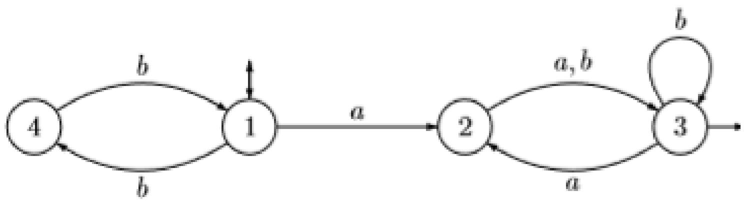

FIGURE $V.3$ Automate minimal du langage L.

Proposition 1.7

L'automate minimal d'un automate A est unique. De plus si l'automate A est déterministe et complet, alors l'automate minimal s'obtient comme le quotient de l'automate A par une relation d'équivalence consistant à identifier des sommets entre eux.

Proposition 1.8

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

Preuve.

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

$w \in L(A_L) \Longleftrightarrow \delta_L(q_0, w) \in F_L \Longleftrightarrow w^{-1}L \in F_L \Longleftrightarrow w \in L$.

On a utilisé le fait que

$\delta_L(q_{0,L}, w) = \delta_L(\varepsilon^{-1}.L, w) = w^{-1}.(\varepsilon^{-1}.L) = (\varepsilon w)^{-1}.L$. ■

Chapitre 2

Etude sur le monoïde syntaxique et le monoïde de transitions.

Ce chapitre contient la définition de monoïde syntaxique et monoïde de transitions ainsi que certaines propriétés, aussi on présente la relation entre les deux. L'étude de ces monoïdes permet de déterminer certaines propriétés combinatoires du langage par des caractéristiques algébriques du monoïde.

2.1 Monoïde syntaxique

Définition 2.1

Soit Σ un alphabet. Pour toute partie L de Σ^*, on appelle contexte de $w \in \Sigma^*$ l'ensemble $C(w) = \{(u, v) \in (\Sigma^*)^2 : uwv \in L\}$.

La relation $\equiv_L$ définie sur Σ^* par :

$w \equiv_L w' \Leftrightarrow C(w) = C(w') \Leftrightarrow \forall u, v \in \Sigma^* : (uwv \in L \Longleftrightarrow uw'v \in L)$

est appelée la congruence syntaxique de L.

Remarque 2.1

La relation $\equiv_L$ est la plus petite congruence sur Σ^* pour laquelle L soit l'union de classe d'équivalences. Une relation d'équivalence $\equiv$ est une relation de congruence à droite si :

$(\forall u, v, w \in \Sigma^*)[u \equiv v \Longrightarrow uw \equiv vw]$.

Remarque 2.2

Nous avons montré que $\equiv_L$ est une congruence à gauche et à droite. Mais en tout généralité, une congruence doit respecter la propriété suivants : si $x \equiv_L x'$ et $y \equiv_L y'$, alors $xy \equiv_L x'y'$, c'est-à- dire qu'elle doit bien se comporter par rapport au produit envisagé, à savoir ici, la concaténation. Et ceci est bien le cas pour tous $\alpha, \beta \in \Sigma^*$, il vient

$$\alpha xy\beta \in L \Longleftrightarrow \alpha x'y\beta \in L \Longleftrightarrow \alpha x'y'\beta \in L.$$

Proposition 2.1

Muni de l'opération " $\circ$ ", le produit de classes, l'ensemble quotient $M(L) = \Sigma^*/\equiv_L$ possède une structure de monoïde, appelé le monoïde syntaxique.

Preuve.

Le neutre est $[\varepsilon]$, i.e, pour tout $x \in \Sigma^*$, on a

$[x] \circ [\varepsilon] = [\varepsilon] \circ [x] = [x]$.

De plus, l'opération " $\circ$ " est associative, i.e, pour tous $x, y, z \in \Sigma^*$,

$([x] \circ [y]) \circ [z] = [x] \circ ([y] \circ [z])$. ■

Exemple 2.1

Soit L le langage formé par des mots sur $\{a, b\}$ ne contenant pas deux bb consécutifs. On remarque tout d'abord que $xay \in L \Leftrightarrow (x \in L$ et $y \in L)$.

De là, on en tire que la classe de a pour la congruence syntaxique $\equiv_L$ est de la forme $[a] = \{awa : w \in L\} \cup \{a\}$. En particulier, $\varepsilon \notin [a]$.

Nous allons voir qu'on peut munir l'ensemble quotion $\Sigma^*/\equiv_L$ i.e, l'ensemble des classes d'équivalence pour $\equiv_L$, d'une structure de monoïde.

Remarque 2.3

1. Il est évident que $[x] \circ [y] = [xy]$.
2. On remarque qu'effectivement, la concaténation de deux classes $[x].[y]$ est formé de mots équivalents mais qu'en générale, il s'agit d'un sous-ensemble strict de la classe d'équivalence $[xy]$.

   En considérant à nouveau le langage L formé des mots sur $\{a, b\}$ ne contenant pas deux bb consécutifs, on a

$[a] \circ [a] = [aa] = [a]$.

Cependant, le mot aba (ou même a) appartient bien à $[a]$ mais ne peut pas se factoriser sons la forme

$aba = xy$ avec $x, y \in [a]$.

Ceci montre bien que $[a].[a] \subsetneq [a]$.

Définition 2.2

Il est facile de vérifier qu'il s'agit d'une relation d'équivalence sur Σ^* et même d'une congruence (à droite et à gauche), i.e, pour tout $\sigma \in \Sigma$,

$u \equiv_L v \Rightarrow (u\sigma \equiv_L v\sigma$ et $\sigma u \equiv_L \sigma v)$.

On parle souvent de la congruence syntaxique $\equiv_L$et on dit que u et v sont syntaxiquement équivalents.

Proposition 2.2

- La congruence syntaxique de L est la plus grossière des congruences sur Σ^* qui sature L.
- Le monoïde syntaxique de L divise tout monoïde qui reconnaît L, pour tout monoïde M qui reconnaît L, il existe un morphisme surjectif de M sur le monoïde syntaxique de L.

Théorème 2.1

Soient $L \subseteq \Sigma^*$ un langage et w, w' deux mots sur Σ. On a $w \equiv_L w'$ si, et seulement si pour tout état q de l'automate minimal de L, $\delta_L(q, w) = \delta_L(q, w')$.

Preuve.

Supposons qu'il existe dans l'automate minimal de L, un état tel que

$$\delta_L(q, w) \neq \delta_L(q, w').$$

Puisque l'automate minimal est réduit, il existe un mot $z \in \Sigma^*$ tel que $\delta_L(\delta_L(q, w), z)$ soit final et $\delta_L(\delta_L(q, w'), z)$ ne le soit pas (ou réciproquement, mais par souci de simplification, nous supposons être dans un tel cas de figure). De plus, l'automate minimale est accessible. Cela signfie qu'il existe un mot $x \in \Sigma^*$ tel que $\delta_L(q_{0,L}, x) = q$. Schématiquement, nous avons la situation suivant reprise en figure$V$.4 par

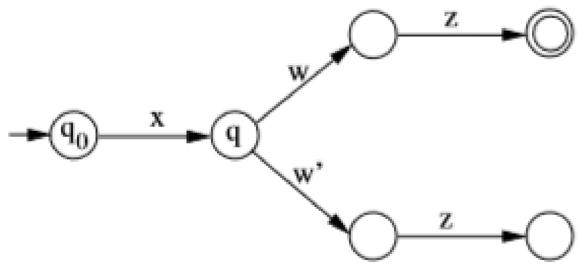

FIGURE $V.4$ Situation dans l'automate minimal.

Par conséquent, $xwz \in L$ et $xw'z \notin L$. Ainsi, les deux mots w et w' ne sont pas syntaxiquement équivalents.

Passons à la réciproque. Si pour tout état q de A_L, on a $\delta_L(q, w) = \delta_L(q, w')$, alors pour tout mot $x \in \Sigma^*$,

$$\delta_L(q_{0,L}, xw) = \delta_L(q_{0,L}, xw')$$

et dès lors, puisque l'automate minimal est déterministe, pour tout $y \in \Sigma^*$, on a

$$\delta_L(q_{0,L}, xwy) = \delta_L(q_{0,L}, xw'y).$$

Schématiquement, on a la situation représentée à la figure $V.5$ Ainsi, pour tous $x, y \in \Sigma^*$, $\quad xwy \in L \iff xw'y \in L.$

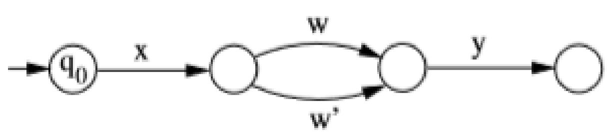

FIGURE $V.5$ Situation dans l'automate minimal. ■

Théorème 2.2

A été énoncé pour un langage L arbitraire. Dans le cas d'un langage régulier, ou obtient un monoïde syntaxique fini.

Corollaire 2.1

Un langage L est régulier si et seulement si son monoïde syntaxique est fini.

Preuve.

Si l'automate minimal A_L de L est fini, l'ensemble Q_L des états de A_L possède un nombre fini n d'éléments. Le nombre de fonctions de Q dans Q est au plus n^n et par conséquent, le monoïde syntaxique de L possède au plus n^n d'éléments. Pour la réciproque, au vu du théorème précédent, si $\delta_L(q_0, w) \neq \delta_L(q_0, w')$, alors $w \neq w'$. Par conséquent, le nombre d'états de l'automate minimal de L est majoré par le nombre de classe du monoïde syntaxique de L. De là, si $\Sigma^*/\equiv_L$est fini, alors l'automate minimal de L est fini et la langage L est régulier. ■

Proposition 2.3

Un monoïde M est dit apériodique s'il existe un $k > 0$ tel que pour tout $m \in M, m^k = m^{k+1}$, et il dit torsion si pour tout $m \in M, \langle m \rangle = \{m, m^2, m^3, ...\}$ est fini. On dira qu'un langage L est de torsion si son monoïde syntaxique est de torsion, de manière équivalente, si pour tout mot $w \in L$ il existe $k \neq p$ tel que :

$(\forall u, v \in \Sigma^*) \quad ux^kv \in L \Longleftrightarrow ux^pv \in L.$

Définition 2.3

Le monoïde syntaxique ordonné est le monoïde syntaxique muni de l'ordre induit par le préordre syntaxique $\leq_L$défini sur Σ^* par : $u \leq_L v$ si et seulement si, pour tout $x, y \in \Sigma^*, xvy \in L \Rightarrow xuy \in L$.

2.2 Monoïde de transitions

Définition 2.4

Soit $A = (Q, \Sigma, \delta, i, F)$ un automate déterministe complet.

Le monoïde de transitions de A est le sous monoïde de $(Q^Q, *)$ engendré par les application $\delta_a : Q \to Q \quad (a \in \Sigma)$ définies par $\delta_a(q) = \delta(q, a)$ et avec la loi de composition interne $f * g = g \circ f$.

Exemple 2.2

1	1	2	3
a	2	2	2
b	1	3	3
c	–	2	3
ab	3	3	3
bc	–	3	3
ca	–	2	2

$$\text{Relations :}\quad \begin{array}{lll} aa = a; & bb = b; & abc = ab \\ ac = a; & cb = bc; & bca = ca \\ ba = a; & ac = c; & cab = bc \end{array}$$

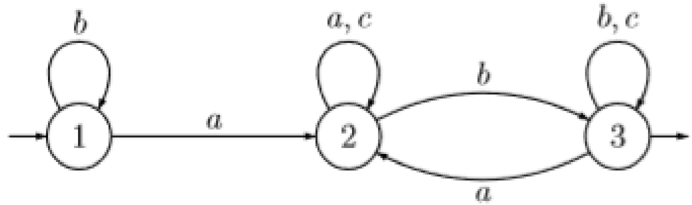

FIGURE V.6 Un automate et une présentation de son monoïde de transition.

Remarque 2.4

Si on part d'un automate fini, le monoïde obtenu est fini, puisque c'est un sous-monoïde du monoïde des fonction de Q dans Q, qui est lui-même fini.

Définition 2.5

Il existe un morphisme naturel de Σ^* dans le monoïde de transition $\zeta(A)$ d'un automate A. Ce morphisme $\varphi : \Sigma^* \longrightarrow \zeta(A)$ est défini par $\varphi(a) = a$ pour tout $a \in \Sigma$.

Autrement dit $\varphi(u)$ est la fonction de Q dans lui-même défini par u. Pour calculer l'image d'un mot par le morphisme, j'utilise la présentation du monoïde de transition obtenue.

Plus généralement deux mots u et v tels que $\varphi(u) = \varphi(v)$ ont la même action sur l'automate et ils seront donc simultanément acceptés ou rejetés. Mais il faut bien noter que la définition du monoïde de transition ne fait pas intervenir les états initiaux et finaux : la connaissance seule du monoïde ne nous permet pas de savoir si un mot est accepté ou pas.

Néanmoins, pour tout élément m du monoïde, les mots de $\varphi^{-1}(m)$ sont ou bien tous acceptés par l'automate, ou bien rejetés. Par conséquent, il suffit de connaître les éléments du monoïde dont les antécédents par φ sont acceptés par l'automate. Dans l'exemple précédent, un mot u est accepté par l'automate si et seulement si $u \in \varphi^{-1}(\varphi(ab))$.

Proposition 2.4

Si $L \subset \Sigma^*$ est reconnu par un automate, il est reconnu par le monoïde de transition de cet automate.

Preuve.

Soit A un automate reconnaissant L, muni de l'état initial q_0 et de l'ensemble des états finaux F. On not $\eta : \Sigma^* \longrightarrow \zeta(A)$ le morphisme canonique, et on pose $p = L\eta$. On va montrer que $p\eta^{-1} = L$. Soit $u \in p\eta^{-1}$: alors $u\eta \in p = L\eta$ par définition et donc $u\eta = v\eta$ pour un certain $v \in L$. Cela signifie que u et v définissent la même application de l'ensemble des états dan lui-même. En particulier $q_0.u = q_0.v$ et comme $v \in L, q_0.v \in F$. Donc $q_0.u \in F$ et on en déduit $u \in L$. On vient ainsi d'établir l'inclusion $L\eta\eta^{-1} \subset L$. L'inclusion opposée est évidente et donc $L = p\eta^{-1}$ comme annoncé. Donc $\zeta(A)$ reconnaît L. ■

Théorème 2.3

Un langage $L \subseteq \Sigma^*$, est rationnel si et seulement s'il est reconnu par un monoïde fini. En particulier, comme le monoïde syntaxique divise tout monoïde reconnaissant L, le langage est rationnel si et seulement si son monoïde syntaxique est fini.

Preuve.

Soit $L \subseteq \Sigma^*$ un langage reconnu par un monoïde fini. Il existe donc un monoïde M et un morphisme $\mu : \Sigma^* \longrightarrow M$ tel qu'il existe une partie P de M. On construit l'automate $A = (M, \Sigma, E, \{1\}, P)$ où l'ensemble de transition est $E = \{(m, a, m\mu(a)), m \in M, a \in \Sigma\}$, cet automate reconnaît L, ce qui prouve le sens indirect de l'assertion. Réciproquement si un langage L est ratinnel alors il est reconnu par un automate fini déterministe $A = (Q, \Sigma, \delta, i, T)$, il suffit alors de prendre le monoïde des transitions qui reconnaît bien L et qui est fini car R_Q est l'ensemble des parties de Q^2 qui est fini car Q est fini. ■

Théorème 2.4

Le monoïde syntaxique de $M(L)$ d'un langage L est isomorphe au monoïde de transitions $\zeta(L)$ de l'automate minimale reconnaissant L.

Preuve.

D'aprés le théorème de rationalité par morphisme, le monoïde $\zeta(L)$ de transitions de l'automate minimale de L reconnaît L. On a $\zeta(L)$ est divisible par $M(L)$.

Soient w et w' ayant des images différentes dans le monoïde des transition de l'automate minimal. Puisque cet automate déterministe, il existe un état p tel que $p.w \neq p.w'$. Soient $q = p.w, q' = p.w'$. Puisque l'automate est minimal, il existe un mot v tel que $q.v$ est final et $q.v'$ n'est pas final (quitte à considérer l'inverse). Il existe également un mot u tel que $i.u = p$. On en déduit que (u, v) appartient à $C(w)$ mais pas à $C(w')$ et que w et w' ont des images différentes dans le monoïde syntaxique. ■

Définition 2.6

Comme Le monoïde syntaxique est isomorphe au monoïde de transitions de l'automate minimale reconnaissant L. Ceci donne un moyen systématique pour calculer Le monoïde syntaxique lorsque L est un langage rationnel.

Ce monoïde est canonique, au sens où j'ai un algoritmme qui me permet de le construie : on calcule d'abord l'automate minimale du langage, puis on calcule le monoïde de transition de cet automate.

Exemple 2.3

Dans l'exemple 1.16, on donne l'automate minimale de langage

$L = (aa + ab + abb + aab + bb)^*, \Sigma = \{a, b\}$.

Le monoïde syntaxique de L est donné par :

	1	2	3	4
$*1$	1	2	3	4
a	2	3	2	0
b	4	3	3	1
$*a^2$	3	2	3	0
$*ab$	3	3	3	0
$*ba$	0	2	2	2
$*b^2$	1	3	3	4
$*aba$	2	2	2	0
$*ba^2$	0	3	3	3

Relations :

$a^3 = a \quad a^2b = ab \quad ab^2 = ab \quad bab = ba^2 \quad b^2a = aba \quad b^3 = b \quad aba^2 = ab.$

Chapitre 3

Le produit direct, semi-direct, et en couronne des monoïdes et leurs applications aux semi-automates.

Dans ce chapitre, on présente les types de produits de monoïdes : produit direct, semi-direct et le produit en couronne et leur applications aux semi automates.

3.1 Le produit direct, semi-direct, et en couronne des monoïdes

Définition 3.1

Soient $(M, *)$ et (N, Δ) deux monoïdes, on considére l'ensemble $M \times N$ le produit cartisien de M et N, et on définit la multiplication " $\cdot$ " dans $M \times N$ par :

$(m_1, n_1) \cdot (m_{2,} n_2) = (m_1 * m_2, n_1 \Delta n_2)$, tel que $m_1, m_2 \in M$ et $n_1, n_2 \in N$.

Ce résultat est un monoïde $(M \times N, \cdot)$ s'appelle le produit direct de M et N. D'où la loi sur $M \times N$ est associative et d'élément neutre $(1_M, 1_N)$.

Exemple 3.1

Soient $M = (\mathbb{N}, +)$ et $N = (\mathbb{N}, \times)$ deux monoïdes, le produit direct $M \times N$ défini par : $(m, n) \cdot (r, s) = (m + r, n \times s)$.

Proposition 3.1

Soit M, N, L trois monoïdes nous pouvons former le produit direct $(M \times N) \times L$ de même $M \times (N \times L)$ et la relation entre ces deux monoïdes est l'isomorphisme :

$(M \times N) \times L \cong M \times (N \times L)$.

Preuve.

L'isomorphisme est $h : (M \times N) \times L \longrightarrow M \times (N \times L)$ défini par :

$h((m, n), l) = (m, (n, l))$, tel que $m \in M, n \in N$et $l \in L$. ■

Proposition 3.2

Soit M, N deux monoïdes, on suppose $\theta : N \longrightarrow End(M)$, est un morphisme de monoïdes. On a $(M \times N, \cdot)$ est un monoïde avec l'opération " $\cdot$ " défini par :

$(m, n) \cdot (m', n') = (m\theta(n)(m'), nn')$ *pour tout* $m, m' \in M$ *et* $n, n' \in N$.

Preuve.

• On monter que " $\cdot$ " est associative dans $M \times N$:

soit $m, m', m" \in M, n, n', n" \in N$, on a

$$\begin{aligned}((m, n) \cdot (m', n')) \cdot (m", n") &= (m\theta(n)(m'), nn') \cdot (m", n") \\ &= (m\theta(n)(m')\theta(nn')(m"), (nn')n") \\ &= (m\theta(n)(m')\theta(n)\theta(n')(m"), (nn')n") \\ &= (m\theta(n)(m'\theta(n')(m")), (nn')n").\end{aligned}$$

Aussi on a

$$\begin{aligned}(m, n) \cdot ((m', n') \cdot (m", n")) &= (m, n) \cdot (m'\theta(n')(m"), n'n") \\ &= (m\theta(n)(n'\theta(n')(m")), n(n'n")).\end{aligned}$$

Alors " $\cdot$ " est associative dans $M \times N$.

• Pour $(m, n) \in M \times N, (m, n) \cdot (1_M, 1_N) = (m\theta(n)(1_M), n1_N) = (m1_M, n) = (m, n)$.

Aussi, on a $(1_M, 1_N) \cdot (m, n) = (1_M\theta(1_N)(m), 1_Nn) = (1_MId_M(m), n) = (m, n)$.

Alors $(M \times N, \cdot)$ est un monoïde. ■

Notation 3.1

Le monoïde $(M \times N, \cdot)$ est appelé le semi-direct produit de M et N par rapport à θ et il est noté par $M \rtimes_\theta N$.

Exemple 3.2

Soient M et N deux monoïdes. Soit $\varphi : N \longrightarrow End(M)$ un morphisme de monoïdes. Pour chaque $(m,n) \in M \times N$ on abrège ${}^n m := [\varphi(n)](m)$. On définit une loi $*$ sur $M \times N$ par :

$$* : \begin{cases} (M \times N)^2 \longrightarrow M \times N \\ (\binom{m}{n}, \binom{\mu}{v}) \to \binom{m^n\mu}{nv} \end{cases}$$

On montrer que $*$ munit $M \times N$ d'une structure de monoïde, notée $M \rtimes^{\varphi} N$.

Soient $\binom{m}{n}$ et $\binom{\mu}{v}$ dans $M \times N$. Les préservation du neutre et de la loi par φ s'écrivent ${}^1m = m$ et ${}^n({}^v m) =^{nv} m$.

De même, les préservations par le morphisme $\varphi(t)$ du neutre et de la loi se réécrivent ${}^n1 = 1$ et ${}^n(m\mu) =^n m^n\mu$.

Montrons que $\binom{1_M}{1_N}$ est neutre : cela vient des égalité

$$\begin{pmatrix}1\\1\end{pmatrix} * \begin{pmatrix}m\\n\end{pmatrix} = \binom{1^1m}{1n} = \begin{pmatrix}1m\\n\end{pmatrix} = \begin{pmatrix}m\\n\end{pmatrix}$$

et $$\begin{pmatrix}m\\n\end{pmatrix} * \begin{pmatrix}1\\1\end{pmatrix} = \binom{{}^1m1}{n1} = \begin{pmatrix}m1\\n\end{pmatrix} = \begin{pmatrix}m\\n\end{pmatrix}.$$

Montrons l'associativité : pour chaque $\binom{m}{n} \in M \times N$, on a

$$\begin{pmatrix}m\\n\end{pmatrix} * \left(\begin{pmatrix}\mu\\v\end{pmatrix} * \begin{pmatrix}r\\s\end{pmatrix}\right) = \begin{pmatrix}m\\n\end{pmatrix} * \begin{pmatrix}\mu^v r\\vs\end{pmatrix} = \begin{pmatrix}{}^nm(\mu^v r)\\n(vs)\end{pmatrix} = \begin{pmatrix}{}^nm^n\mu({}^v r)\\nvs\end{pmatrix} ...(1)$$

et $$\left(\begin{pmatrix}m\\n\end{pmatrix} * \begin{pmatrix}\mu\\v\end{pmatrix}\right) * \begin{pmatrix}r\\s\end{pmatrix} = \begin{pmatrix}m^n\mu\\nv\end{pmatrix} * \begin{pmatrix}r\\s\end{pmatrix} = \begin{pmatrix}{}^nm\mu^{nv}r\\(nv)s\end{pmatrix} = \begin{pmatrix}{}^nm^n\mu^v r\\nvs\end{pmatrix} ......(2)$$

On a $(1) = (2)$, alors $(M \times N, *)$ est un monoïde.

Proposition 3.3

Soient M et N deux monoïdes, et soit M^N l'ensemble de toutes les applications de N dans M, puis l'ensemble $M^N \times N$ est un monoïde sous la multiplication $(\varphi, n_1)(\psi, n_2) = (\varphi\psi, n_1n_2)$ où $\varphi\psi \in M^N$ est défini par :

$\varphi\psi(x) = \varphi(x)\psi(xn_1), \qquad$ *pour $x, n_1, n_2 \in N$ et $\varphi, \psi \in M^N$.*

Nous appelons le monoïde $M^N \times N$ le produit en couronne de M et N.

Preuve.

• On montrer que la multiplication est associative dans $M^N \times N$. Soit $\varphi, \psi, \eta \in M^N$ et $n_1, n_2, n_3 \in N$ alors

$((\varphi, n_1)(\psi, n_2))(\eta, n_3) = (\varphi\psi, n_1n_2)(\eta, n_3) = ((\varphi\psi)\eta, n_1n_2n_3)$.

Et $(\varphi, n_1)((\psi, n_2)(\eta, n_3)) = (\varphi, n_1)(n_2n_3, \psi\eta) = (\varphi(\psi\eta), n_1n_2n_3)$.

On montre que, $(\varphi\psi)\eta = \varphi(\psi\eta)$. Soit $x \in N$, alors

$((\varphi\psi)\eta)(x) = (\varphi(x)\psi\eta(xn_1) = \varphi(x)\psi(xn_1)\eta(xn_1n_2)$.

Soit l'application $e : N \longrightarrow M$ donné par $e(n) = 1_M$ pour tout $n \in N$.

On a $(\varphi, n)(e, 1_N) = (\varphi e, n1_N) = (\varphi e, n)$.

Où $(\varphi e)(x) = \varphi(x)e(xm) = \varphi(x)1_M = \varphi(x)$

pour $n, x \in N$ et $\varphi \in M^N$, puis $(\varphi, n)(e, 1_N) = (\varphi, n)$

Comme $(e, 1_N)(\varphi, n) = (e\varphi, 1_N n) = (e\varphi, n)$,

Où $(e\varphi)(x) = e(x)\varphi(x1_N) = 1_M\varphi(x) = \varphi(x)$.

Donc $(e, 1_N)(\varphi, n) = (\varphi, n)$. L'élément identité dans $M^N \times N$ est $(e, 1_N)$. ■

3.2 L'application de produit des monoïdes en monoïde de semi-automate

Définition 3.2

Soient $S = (Q, \Sigma, \delta)$ et $S' = (Q', \Sigma', \delta')$ deux semi automates, alors leurs produit direct est le semi automate $S \times S' = (Q \times Q', \Sigma, \delta \times \delta')$, dans le cas où $\Sigma = \Sigma'$, avec $\delta \times \delta'((q, q'), \sigma) = (\delta(q, \sigma), \delta(q', \sigma))$ pour tous $\sigma \in \Sigma, (q, q') \in Q \times Q'$.

Exemple 3.3

Soit $S = (Q, \Sigma, \delta)$ et $S' = (Q', \Sigma', \delta')$ deux semi automates, tel que $Q = Q' = \{0, 1\}, \Sigma = \Sigma' = \{\sigma\}$ et δ, δ' sont définies comme suit :

δ	σ
0	1
1	0

δ'	σ
0	1
1	1

Alors, le produit direct de S, S' est donné par : $S \times S' = (Q \times Q', \Sigma, \delta \times \delta')$ où $Q \times Q' = \{(0, 0), (0, 1), (1, 0), (1, 1)\}$ et $\delta \times \delta' : (Q \times Q') \times \Sigma \longrightarrow (Q \times Q')$ est donné par :

$\delta \times \delta'$	σ
$(0, 0)$	$(1, 1)$
$(0, 1)$	$(1, 1)$
$(1, 0)$	$(0, 1)$
$(1, 1)$	$(0, 1)$

▶Le monoïde de semi automate $S = (Q, \Sigma, \delta)$ est donné par :

.	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\epsilon]$

▶Le monoïde de semi automate $S' = (Q', \Sigma', \delta')$ est donné par :

.	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\sigma]$

▶On défini $(\delta \times \delta')_\sigma : Q \times Q' \longrightarrow Q \times Q'$ par $(\delta \times \delta')_\sigma((q, q')) = (\delta(q, \sigma), \delta'(q', \sigma))$ pour tout $(q, q') \in Q \times Q'$.

On a $(\delta \times \delta')_\sigma = \varphi = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,1) & (0,1) & (0,1) \end{pmatrix}$,

$$\varphi^2 = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,1) & (1,1) & (1,1) \end{pmatrix}, \varphi^3 = \varphi.$$

Soit $w \in \{\sigma\}^*$ est un mot, suppose $w = \sigma^n, n \in \mathbb{N}^*$ alors, on défini $(\delta \times \delta')_w$: $Q \times Q' \longrightarrow Q \times Q'$

$(\delta \times \delta')_w((q, q')) = \delta_\sigma \delta_\sigma ... \delta_\sigma((q, q'))$.

On a $(\delta \times \delta')_w = \begin{cases} Id_Q & \text{si } n = 0 \\ \varphi & \text{si } n = 2k+1, k \in \mathbb{N} \\ \varphi^2 & \text{si } n = 2k, k \in \mathbb{N}^* \end{cases}$

$\psi : (\{\sigma\}^*, .) \longrightarrow \left((Q \times Q')^{Q \times Q'}, \circ\right)$ par $\psi(w) = (\delta \times \delta')_w$.

On défini la relation R dans $\{\sigma\}^*$ par wRw' si et seulement si $\psi(w) = \psi(w')$ est une congruence. Le monoïde quotion $\{\sigma\}^*/R = \{[\epsilon], [\sigma], [\sigma^2]\}$.

▶Le monoïde de semi automate $(Q \times Q', \Sigma, \delta \times \delta')$ est donné par :

.	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$	$[\sigma^2]$
$[\sigma]$	$[\sigma]$	$[\sigma^2]$	$[\sigma]$
$[\sigma^2]$	$[\sigma^2]$	$[\sigma]$	$[\sigma^2]$

Définition 3.3

Soit $S = (Q, \Sigma, \delta)$ et $S' = (Q', \Sigma', \delta')$ deux semi automates, on défini leurs produit direct général dans le cas $\Sigma \neq \Sigma'$ par : $S \times S' = (Q \times Q', \Sigma \times \Sigma', \delta \times \delta')$ tel que

$$(\delta \times \delta')((q, q'), (\sigma, \sigma')) = (\delta(q, \sigma), \delta'(q', \sigma')) \text{ avec } (\sigma, \sigma') \in \Sigma \times \Sigma', (q, q') \in Q \times Q'.$$

Exemple 3.4

Soit $S = (Q, \Sigma, \delta)$ est un semi automate où $Q = \{0, 1\}, \Sigma = \{\sigma\}$ et $\delta : Q \times \Sigma \longmapsto Q$ est donné par :

δ	σ
0	1
1	0

et $S' = (Q', \Sigma', \delta')$ est donné par

$Q' = \{0, 1\}, \Sigma' = \{\sigma, \tau\}$ et $\delta' : Q' \times \Sigma' \longrightarrow Q'$

δ'	σ	τ
0	1	0
1	1	0

Alors, $S \times S' = (Q \times Q', \Sigma \times \Sigma', \delta \times \delta')$ où $Q \times Q' = \{(0, 0), (0, 1), (1, 0), (1, 1)\}, \delta \times \delta' : (Q \times Q') \times \Sigma \times \Sigma' \longrightarrow (Q \times Q')$ donné par :

$\delta \times \delta'$	(σ, σ)	(σ, τ)
$(0, 0)$	$(1, 1)$	$(1, 0)$
$(0, 1)$	$(1, 1)$	$(1, 0)$
$(1, 0)$	$(0, 1)$	$(0, 0)$
$(1, 1)$	$(0, 1)$	$(0, 0)$

•Le monoïde de semi automate $(Q, \{\sigma\}, \delta)$ est donné par :

$.$	$[\epsilon]$	$[\sigma]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$
$[\sigma]$	$[\sigma]$	$[\epsilon]$

•Le monoïde de semi automate $(Q', \{\sigma, \tau\}, \delta')$ est donné par :

$.$	$[\epsilon]$	$[\sigma]$	$[\tau]$
$[\epsilon]$	$[\epsilon]$	$[\sigma]$	$[\tau]$
$[\sigma]$	$[\sigma]$	$[\sigma]$	$[\tau]$
$[\tau]$	$[\tau]$	$[\sigma]$	$[\tau]$

On défini $(\delta \times \delta')_{(\sigma,\sigma)} : Q \times Q' \longrightarrow Q \times Q'$ par $(\delta \times \delta')_{(\sigma,\sigma)}((q, q')) = (\delta(q, \sigma), \delta'(q', \sigma))$ pour tout $(q, q') \in Q \times Q'$.

On a $(\delta \times \delta')_{(\sigma,\sigma)} = \eta = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,1) & (0,1) & (0,1) \end{pmatrix}$,

$$\eta^2 = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,1) & (1,1) & (1,1) \end{pmatrix}, \eta^3 = \eta.$$

Et $(\delta \times \delta')_{(\sigma,\tau)} : Q \times Q' \longrightarrow Q \times Q'$ par $(\delta \times \delta')_{(\sigma,\tau)}((q, q')) = (\delta(q, \sigma), \delta'(q', \tau))$ pour tout $(q, q') \in Q \times Q'$. On a

$$(\delta \times \delta')_{(\sigma,r)} = \mu = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,0) & (1,0) & (0,0) & (0,0) \end{pmatrix},$$

$$\mu^2 = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,0) & (0,0) & (1,0) & (1,0) \end{pmatrix}, \mu^3 = \mu.$$

•Le monoïde de semi automate $(Q \times Q', \Sigma \times \Sigma', \delta \times \delta')$ est donné par :

.	$[\epsilon]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[\epsilon]$	$[\epsilon]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[(\sigma,\sigma)]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\tau)^2]$	$[(\sigma,\tau)]$
$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$
$[(\sigma,\tau)]$	$[(\sigma,\tau)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\tau)^2]$	$[(\sigma,\tau)]$
$[(\sigma,\tau)^2]$	$[(\sigma,\tau)^2]$	$[(\sigma,\sigma)]$	$[(\sigma,\sigma)^2]$	$[(\sigma,\tau)]$	$[(\sigma,\tau)^2]$

Définition 3.4

Soit $S = (Q, \Sigma, \delta)$ et $S' = (Q', \Sigma', \delta')$ deux semi automates. On défini le produit en couronne de S et S' par $SWS' = (Q \times Q', \Sigma \times \Sigma', \delta^W)$ où $\delta^W((q,q'),(f,\sigma')) = (\delta(q, f(q')), \delta'(q', \sigma'))$ pour $\sigma' \in \Sigma'$, $f \in \Sigma^{Q'}$, $(q,q') \in Q \times Q'$.

Exemple 3.5

Soit $S = (Q, \Sigma, \delta)$ est un semi automate où $Q = \{0,1\}, \Sigma = \{\sigma, \tau\}$ et $\delta : Q \times \Sigma \longmapsto Q$ est donné par :

δ	σ	τ
0	1	0
1	1	0

et $S' = (Q', \Sigma', \delta')$ est donné par $Q' = \{0,1\}, \Sigma' = \{\sigma\}$ et $\delta' : Q' \times \Sigma' \longmapsto Q'$ est donné par :

δ	σ
0	1
1	0

.

On note les quatre éléments de $\Sigma^{Q'}$ par f_1, f_2, f_3, f_4 où

$f_1(0) = f_1(0) = \sigma$.

$f_2(0) = \sigma, f_2(1) = \tau$.

$f3(0) = \tau, f_3(1) = \sigma$.

$f_4(0) = f_4(1) = \tau$.

Alors, le semi automate SWS' a le table

δ^w	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$
(f_1,σ)	$(1,1)$	$(1,0)$	$(1,1)$	$(1,0)$
(f_2,σ)	$(1,1)$	$(0,0)$	$(1,1)$	$(0,0)$
(f_3,σ)	$(0,1)$	$(1,0)$	$(0,1)$	$(1,0)$
(f_4,σ)	$(0,1)$	$(0,0)$	$(0,1)$	$(0,0)$

On défini $(\delta^W)_{(f_1,\sigma)} : Q\times Q' \longrightarrow Q\times Q'$ par$(\delta^W)_{(f_1,\sigma)}((q,q')) = (\delta(q, f_1(q')), \delta'(q',\sigma))$ pour tout $(q,q') \in Q \times Q'$. On a

$$(\delta^W)_{(f_1,\sigma)} = \alpha = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (1,0) & (1,1) & (1,0) \end{pmatrix},$$

$$(\delta^W)_{(f_2,\sigma)} = \beta = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (1,1) & (0,0) & (1,1) & (0,0) \end{pmatrix},$$

$$(\delta^W)_{(f_3,\sigma)} = \gamma = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (1,0) & (0,1) & (1,0) \end{pmatrix},$$

$$(\delta^W)_{(f_4,\sigma)} = \lambda = \begin{pmatrix} (0,0) & (0,1) & (1,0) & (1,1) \\ (0,1) & (0,0) & (0,1) & (0,0) \end{pmatrix}.$$

Conclusion

Dans ce mémoire, on a fait une étude sur les types de produit des monoïdes et quelque applications dans les semi-automates.

Nous avons présenté dans le premier chapitre les définitions et quelques propriétés sur les monoïdes, langages et semi automates.

Ensuite nous avons fait une etude sur le monoïde syntaxique et le monoïde de transitions.

Finalement nous avons etudé aussi les produits (produit direct, semi direct et en couronne) des monoïdes et leurs applications aux semi-automates.

Bibliographie

[1] **Abraham Ginzbureg**, *Algebraic Theory of Automata*, Academic Press New York London, 1968.

[2] **Dominique Perrin,** Sous-monoïdes et automates, Secrétariat Mathématique, Paris, 1969-1970.

[3] **J.E.Pin**, *Automate Réversibles : Combinatoire,Algèbre et Topologie*, 2006.

[4] **J.E.Pin**, *Introduction aux Langages Reconnaissables*, Publication du Département de Mathématique de Lyon.,1984.

[5] **J-F.Perrot**, *Varietes de Langage et Operations*, Université Paris et Laboratoire D'informatique Théorique et Programation, 1977-1978.

[6] **Louis Frécon**, *Eléments de Mathématiques Discrètes*, Collection des Siences Appliquées de l'NSA de Lyon.

[7] **Marc Sage,** Groupes,Monoïdes(reliquat), 28 *Février* 2018.

[8] **Messouda Djaija**, *Memoire de Fin D'etude, Semi-groupe de Transformation et Décomposition d'un Automate Fini, Universite Mohamed Boudiafe de M'sila,* 2017-2018.

[9] **Michaël Cadlihac**, *Thèse Comme Requiss Par L'examen Pré-doctoral, Autour de L'automate de Parikh, Octobre* 2010.

[10] **Michel Rigo,** Théorie des Automates et Langages Formels, Univesité de liége, 2009-2010.

[11] **Michelle Strumila**, TQFTs,Contact Geometry and Automata, School of Mathematics Monash University, Semester 2,2015.

[12] **Nacer Ghadbane and Douadi Mihoubi,** Products of Monoids and Its Application on the Monoids of State Machines, Departement of Mathematices,*University of M'sila,Algeria, 2018.*

[13] **Nacer Ghadbane,** Cours Master 1ére Année, Semi Groupes et Automates Finis, Département de Mathématiques,*Université de M'sila, 2017-2018.*

[14] **Oliver Carton,** Langages Formels Calculabilité et Complexité, Juin 2014.

[15] **Oliver Carton et Jean-Éric Pin,** MPRI,Fondation Mathématiques de la Théorie des Automates, 13 March 2013.

[16] **Paul Gastin,** Langages Formels, Magistère 2006.

[17] **Paul Mercat,** Semi Groupes Fortement Automatiques, Université Paris-Sud, 17 Janvier 2013.

[18] **Tony Bourdier,** Mathématiques Discrètes 1 et Informatique Théorique, École Supérieure d'Informatique et Application de Lorraine.

Abstract :

This thesis is part of the theory of semi automata.In this work,we will follow the following steps :

* Elementary notions on monoids and semi automata.

* The syntactic monoid of a langage and the transitions monoide of a semi automata.

* The direct, semi direct, and crown product of monoids and their applications to semi automata.

Key words :

Monoid, morphisme of monoide, words and langage, semi automata, syntactic monoide, monoid of transitions, direct product, semi direct product, crown product.

Résumé :

Ce mémoire sinscrit dans le cadre de la théorie des semi-automates. Dans ce travail, nous suivrons les étapes suivantes :

* Notions élémentaires sur les monoïdes et semi automate.

* Monoïde syntaxique d’un langage et le monoïde de transitions d’un semi-automate.

* Produit direct, semi-direct, et le produit en couronne des monoïdes et leurs applications aux semi-automates.

Mots clés :

Monoïde, morphisme de monoïdes, mots et langage, semi-automate, monoïde syntaxique, monoïde de transitions, produit direct, produit semi direct, produit en couronne.

ملخص:

هذه الاطروحة جزء من نظرية شبه اتوماتيكية.في هذا العمل نتبع الخطوات التالية:

* مفاهيم اولية لنصف زمرة و شبه الة.

* نصف زمرة النحوية للغة و الانتقالات احادية الصيغة شبه الالية.

* الجداء المباشر، وشبه المباشر، و الجداء التاجي للاحادية وتطبيقاتها على شبه الة.

الكلمات المفتاحية:

نصف زمرة،تماثل نصف زمرة،كلمات و لغات،شبه الة، نصف زمرة نحوية، احادي الانتقالات، الجداء المباشر،الجداء شبه المباشر، الجداء التاجي.