Rapport : Transducteurs et machines de Moore et Mealy

19 décembre 2024

Table des matières

1 Généralités sur les transducteurs finis

Un transducteur fini est une généralisation des automates finis, implémentant non pas des langages mais plutôt des relations entre des ensembles de mots. Cela se fait à l'aide d'étiquettes supplémentaires sur chaque transition entre deux états.

1.1 Définition des transducteurs

Définition 1. *Un transducteur fini est un 6-uplet $T = (\Sigma, \Gamma, Q, I, F, \delta)$ où*
— Σ *est l'alphabet d'entrée (fini)*
— Γ *l'alphabet de sortie (fini)*
— Q *l'ensemble des états (fini)*
— $I \subset Q$ *l'ensemble des états initiaux*
— $F \subset Q$ *l'ensemble des états finaux*
— $\delta : Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \longrightarrow \mathcal{P}(Q)$ *est la fonction de transition non déterministe*

En d'autres termes, T est un automate fini non déterministe sur l'alphabet $\Sigma_\varepsilon \times \Gamma \cup \Sigma \times \Gamma_\varepsilon$

On peut comme pour les automates définir $\delta^* : Q \times \Sigma^* \times \Gamma^* \longrightarrow \mathcal{P}(Q)$ par induction :

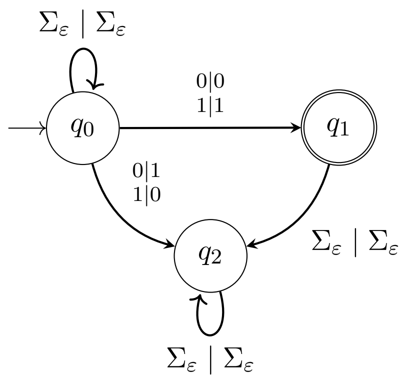
$$\delta^*_{|Q\times\Sigma_\varepsilon\times\Gamma_\varepsilon} = \delta$$
$$\forall (a,b) \in \Sigma_\varepsilon \times \Gamma_\varepsilon, \forall (w,v) \in \Sigma^* \times \Gamma^*, \forall q \in Q, \delta^*(q, wa, vb) = \bigcup_{p\in\delta^*(q,w,v)} \delta(p,a,b)$$

On définit ici la relation reconnue par un transducteur fini, de manière similaire que les langages reconnus par les automates.

Définition 2. *On défini* $[T]$ *la relation reconnue par un transducteur fini* T*, définie comme suit :*

$$\forall (u,v) \in \Sigma^* \times \Gamma^*, u[T]v \Leftrightarrow \exists q \in I, \exists r \in F, r \in \delta^*(q,u,v)$$

FIGURE 1 – Un exemple de transducteur fini reconaissant la relation de congruence modulo 2
Ici $\Sigma = \Gamma = \{0,1\}$ et les transitions sont étiquettées par lettre d'entrée, lettre de sortie



Proposition 1 (Généralisation)**.** *On peut s'autoriser des fonctions de transition* $Q \times \Sigma^* \times \Gamma^* \longrightarrow P(Q)$ *tant qu'il n'y a qu'un nombre fini de triplets* $(q,w,v) \in Q \times \Sigma^* \times \Gamma^*$ *tels que* $\delta(q,v,w) \neq \emptyset$*. En effet, ajouter cette transition correspond à ajouter les états* $q = q_0, q_1, \dots q_{|w|} = q'_0, q'_1, \dots, q'_{|v|-1}$ *avec les transitions* $\delta(q_i, w_i, \varepsilon) = \{q_{i+1}\}$*, les* $\delta(q'_i, \varepsilon, v_i) = \{q'_{i+1}\}$*, et enfin* $\delta(q_{|v|-1}, \varepsilon, v_{-1}) = \delta(q,w,v)$*.*

1.2 Relations rationnelles

Soient Σ et Γ des alphabets finis. Nous allons ici étudier une classe particulière de relations entre Σ^* et Γ^*, que l'on peut voir comme des parties de $\Sigma^* \times \Gamma^*$, les relations rationelles. Généralisons pour cela les opérations usuelles sur les langages

Définition 3. *Soient* $R, R' \subset \Sigma^* \times \Gamma^*$ *des relations. On pose*
- $R \cdot R' = \{(xx', yy') \mid (x,y) \in R, (x',y') \in R'\}$*, la concaténation ou produit*
- $R^* = \bigcup_{n \geq 0} R^n$ *où* R^n *est la concaténation répétée, et* $R^0 = \{(\varepsilon, \varepsilon)\}$

Définition 4. *L'ensemble des relations rationelles est défini comme le plus petit ensemble de relations stable par étoile, concaténation et union, et contenant* $\varnothing$ *et les singletons.*

Pour être plus général, une relation rationelle est une partie rationnelle du monoïde produit $\Sigma^* \times \Gamma^*$. Comme pour les automates on a bien équivalence entre les relations reconnues par des transducteurs finis et les relations rationelles.

1.3 Propriétés des relations rationnelles et transducteurs finis

De manière analogue au cas des automates finis et des langages rationnels, on dispose du théorème suivant reliant les transducteurs finis aux relations rationelles :

Théorème 1. *Une relation* R *est rationelle si et seulement si elle est reconnue par un transducteur fini* T

Démonstration. La preuve est exactement du même acabi que celle pour les automates finis et langages rationnels. □

Cette équivalence permet de voir plus simplement certaines propriété de stabilité des transducteurs finis ou des relations rationnelles.

Proposition 2. *Les relations rationnelles sont stables par inverse. De plus les projections d'une relation rationelle sont des langages rationnels.*

Démonstration. Si R est une relation rationelle sur $\Sigma^* \times \Gamma^*$, on note R^{-1} la relation inverse et

$$R_\Sigma = \{w \in \Sigma^* \mid \exists u \in \Gamma^*, (w,u) \in R\}$$

et de même manière R_Γ, les projections à droite et à gauche de R. Soit T un transducteur fini reconaissant R.
Si l'on considère le transducteur où les étiquettes d'entrée et de sorties sont inversées, ce dernier reconnait bien la relation R^{-1}.

□

Proposition 3. *Soit $L \subset \Sigma^*$ rationnel. L'image par R une relation rationnelle de L est également régulière.*

Démonstration. Soit $\mathcal{A} = (\Sigma, Q_L, I_L, F_L, \delta_L)$ un automate reconaissant L. On en fait un transducteur $T_L = (\Sigma, \Gamma, Q_L, I_L, F_L, \delta'_L)$ où

$$\forall q \in Q_L, \forall a \in \Sigma, \forall b \in \Gamma_\varepsilon, \delta'_L(q,a,b) = \delta_L(q,a)$$

(cela revient à coller Γ_ε en étiquette de sortie de chaque transition de $\mathcal{A}$). Si on considère maintenant T_R un transducteur fini reconaissant R, et que l'on considère le transducteur produit de T_R et T_L, ayant pour états finaux les produits des états finaux, ce dernier reconait la relation

$$R' = \{(u,v) \in R \mid u \in L\}$$

En considérant la projection à droite, on obtient alors le langage $R(L) = \{v \in \Gamma^* \mid \exists u \in L, (u,v) \in R\}$ qui est bien l'image de L par R. Donc par la proposition 2, $R(L)$ est rationel. □

Proposition 4 (Composition)**.** *La composition de deux relations rationnelles est rationnelle.*

Démonstration. Si $T_1 = (\Sigma, \Gamma, Q_1, I_1, F_1, \delta_1)$ et $T_2 = (\Gamma, \Lambda, Q_2, I_2, F_2, \delta_2)$ sont des transducteurs implémentant ces deux relations, on construit un transducteur
$T = (\Sigma_\varepsilon \times \Gamma \cup \Sigma \times \Gamma_\varepsilon, \Lambda, Q_1 \times Q_2, I_1 \times I_2, F_1 \times F_2, \delta_3)$ où δ_3 est définie pour $(q_1, q_2) \in Q_1 \times Q_2$:

$$\begin{aligned}
&\forall c \in \Lambda_\varepsilon, \forall q'_2 \in \delta_2(q_2, \varepsilon, c), (q_1, q'_2) \in \delta_3((q_1,q_2),(\varepsilon,\varepsilon),c)\\
&\forall a \in \Sigma, \forall q'_1 \in \delta_1(q_1, a, \varepsilon), (q'_1, q_2) \in \delta_3((q_1,q_2),(a,\varepsilon),\varepsilon)\\
&\forall a \in \Sigma_\varepsilon, \forall b \in \Gamma, \forall c \in \Lambda_\varepsilon, \forall q'_1 \in \delta_1(q_1,a,b), \forall q'_2 \in \delta_2(q_2,b,c), (q'_1,q'_2) \in \delta_3(q_1,q_2,(a,b),c)
\end{aligned}$$

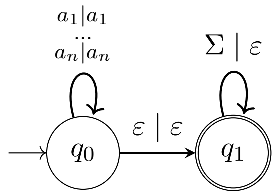
L'idée derrière T est d'effectuer le calcul de T_1, et à chaque fois qu'une lettre est produite en sortie, on donne cette dernière en entrée à T_2 et l'on fait ainsi avancer le calcul de T_2

Notons que, si l'on regarde uniquement le premier élément du couple des états, on retrouve exactement le comportement de T_1. Si on regarde uniquement les deuxièmes éléments des couples des états, on trouve le deuxième transducteur, qui lit chaque lettre du deuxième élément du couple de sortie au moment de la transition étiquetée en sortie par cette lettre.

Ce transducteur implémente donc la relation binaire $[T]$ qui contient $((u,v),w)$ si et seulement si (u,v) est dans la première relation, et (v,w) dans la deuxième. De la même manière que dans la proposition 3, on peut en intersectant $[T]$ avec $(\Sigma^* \times \{\varepsilon\}) \times \Lambda^*$ obtenir la relation de $\Sigma^* \times \Lambda^*$ en identifiant $\Sigma^* \times \{\varepsilon\}$ à Σ^* (rationelle par la construction de la proposition précedente) $[T_1] \circ [T_2]$. □

Un exemple d'application de ces propriétés de stabilité est une preuve rapide de la rationnalité de $\mathrm{Pref}(L)$, l'ensemble des préfixes d'un langage rationnel L : la relation Pref sur $\Sigma^* \times \Sigma^*$ qui met en relation w avec l'ensemble de ses préfixes est rationnelle, reconnue par le transducteur figure 2. Par la propriété 3, on obtient immédiatement la rationnalité de $\mathrm{Pref}(L)$

FIGURE 2 – Le transducteur reconaissant la relation Pref avec $\Sigma = \{a_1, ..., a_n\}$



Notons toutefois que certaines propriétés de stabilité des automates finis ne se transmettent pas aux relations rationelles : c'est le cas par exemple de l'intersection et du complémentaire. On se place sur les alphabets $\Sigma = \{a\}$ et $\Gamma = \{a, b\}$. On se donne les relations rationelles suivantes :

$$R_l = \{(a^n, w) \mid w \in \Gamma^*, n \in \mathbb{N}, |w| = 2n\}$$
$$R_b = \{(a^n, w) \mid w \in \Gamma^*, n \in \mathbb{N}, |w|_b = n\}$$

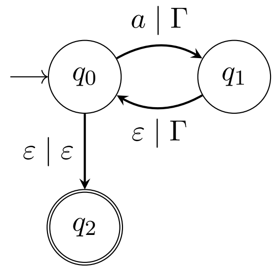
R_l étant reconnue par le transducteur figure 3, et R_b par 4.
On a alors que

$$\begin{aligned} R &= \{(a^n, a^n b^n) \mid n \in \mathbb{N}\} \\ &= R_l \cap R_b \cap (\mathrm{Pref})^{-1} \end{aligned}$$

alors que R n'est clairement pas rationelle par la proposition 2 car la projection droite de R n'est pas rationelle.

On en déduit également que des passages au complémentaires de relations ne sont généralement pas rationnelles. En effet, si c'était le cas, $(R_1^C \cup R_2^C)^C = R_1 \cap R_2$ serait rationnelle puisque l'union l'est (proposition 2).

FIGURE 3 – Le transducteur reconaissant la relation R_l

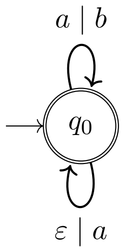


1.4 Quelques problèmes associés aux transducteurs finis

Une question naturelle que l'on peut se poser est celle de déterminer si deux transducteur T_1 et T_2 implémentent la même relation rationelle. De manière assez surprenante, étant donnés les résultats connus pour les automates finis, ce problème est, dans le cas général, indécidable [1] :

Théorème 2. *Le problème de l'équivalence de deux transducteurs est indécidable.*

Figure 4 – Le transducteur reconaissant la relation R_b

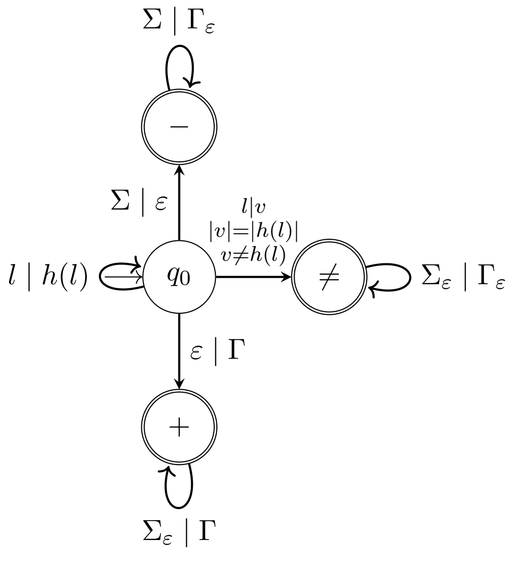


Démonstration. Cette preuve réduit le problème d'équivalence de Post vers l'équivalence de deux transducteurs.
On rappelle l'énoncé du problème d'équivalence de Post, indécidable :

Définition 5 (Problème d'équivalence de Post)**.** *Étant donné deux listes finies de mots dans* $\Gamma^* \backslash \{\varepsilon\}$ $\alpha_1, \ldots, \alpha_n, \beta_1, \ldots, \beta_n$*, existe-t-il une suite* $(i_k)_{1 \leq k \leq N}$ *telle que* $\alpha_{i_1}\alpha_{i_2}\ldots\alpha_{i_N} = \beta_{i_1}\beta_{i_2}\ldots\beta_{i_N}$ *?*

En posant $g(x_i) = \alpha_i$ et similairement $h(x_i) = \beta_i$ sur un alphabet $\Sigma = \{x_1, \ldots, x_n\}$ et $h(ab) = h(a)h(b)$, le problème devient : existe-t-il un mot $w \in \Sigma^* \backslash \{\varepsilon\}$ tel que $g(w) = h(w)$?
On construit le transducteur de la figure 5 qui à un mot x associe $\Gamma^* \backslash \{h(x)\}$.
Ici, l'état marqué $-$ reçoit les mots qui ont moins de lettre que $h(x)$, $+$ ceux qui ont plus de lettres que $h(x)$, et $\neq$ ceux qui ont au moins un caractère différent.
Similairement, on construit le transducteur qui à x associe tous les mots sauf $g(x)$. Enfin, par la propriété 2, on peut construire l'automate qui reconnaît l'union. L'union de ces deux ensembles est d'ailleurs $\Sigma^* \times \Gamma^*$ sauf s'il existe un x tel que $f(x) = g(x)$: l'équivalence de ce transducteur avec le transducteur reconnaissant tout $\Sigma^* \times \Gamma^*$ permet donc de conclure.

Figure 5 – Le transducteur qui à un mot x associe $\Gamma^* \backslash \{h(x)\}$



□

On peut également se poser la question du caractère fonctionnel d'un transducteur : étant donné un transducteur fini T, est-il possible de déterminer si ce dernier est fonctionnel ou non, c'est à dire que la relation qu'il implémente est en réalité une fonction ? Ce problème est décidable [5] et l'est même en temps polynômial en la taille de T : la preuve utilise des notions algébriques sur les monoïdes pour aboutir à une caractérisation de la fonctionnalité de T.

2 Machines de Moore

Une machine de Moore est un cas particulier d'un transducteur fini : d'abord une telle machine est déterministe et implémente donc non pas une relation rationelle mais une fonction, qui sera appelée rationelle. La contrainte supplémentaire sur ces machines est que la lettre de sortie lue ne dépend que de l'état actuel de la machine.

2.1 Définition et exemple

On peut évidemment définir une machine de Moore en partant de la définition de transducteur et en posant des contraintes sur la fonction de transition. Il est toutefois plus commode de la définir ainsi

Définition 6. *Une machine de Moore $\mathcal{M}$ est un 7-uplet $(\Sigma, \Gamma, Q, q_0, F, \delta, \lambda)$ où*
- — Σ *est l'alphabet d'entrée*
- — Γ *l'alphabet de sortie*
- — Q *l'ensemble fini des états*
- — q_0 *l'état initial*
- — $F \subset Q$ *les états finaux*
- — $\delta : Q \times \Sigma \to Q$ *la fonction de transition déterministe*
- — $\lambda : Q \to \Gamma$ *la fonction de sortie*

Un calcul de $\mathcal{M}$ sur un mot w est défini de la même manière que pour un automate déterministe. On ajoute toutefois la notion de mot produit par ce calcul, qui est le mot obtenu en concaténant les lettres obtenues par application de λ sur chaque état pris lors du calcul de $\mathcal{M}$ sur w. Plus formellement

Définition 7. *On définit pour $q \in Q$ et $w \in \Sigma^*$ $q \star w$ récursivement :*

$$\forall a \in \Sigma, q \star a = \lambda(\delta(q, a)) \text{ et } \forall u \in \Sigma^*, q \star ua = (q \star u)(\delta^*(q, u) \star a)$$

Le mot produit par le calcul de $\mathcal{M}$ sur w est défini comme $q_0 \star w$. Notons $L \subset \Sigma^*$ le langage reconnu par l'automate déterministe contenu dans la définition de $\mathcal{M}$. La fonction rationelle réalisée par $\mathcal{M}$ est alors définie par

$$f : \begin{array}{ll} L & \longrightarrow \Gamma^* \\ w & \longmapsto q_0 \star w \end{array}$$

On peut généraliser la fonction réalisée par $\mathcal{M}$ en la définissant non pas sur L mais sur tout Σ^*. Cela revient à rendre tout les états acceptants

Figure 6 – Un exemple de machine de Moore
Ouais jsp j'ai envie de mettre une autre machine de Moore un peu golri

3 Machines de Mealy

3.1 Définition

Une machine de Mealy est également un cas particulier de transducteur fini, moins restrictive à première vue que les machines de Moore. Il s'agit d'un transducteur déterministe sans ε-transition.

Du point de vue du formalisme, elles sont définies de la même manière qu'un automate de Moore, à l'exception de la fonction de sortie λ qui est cette fois $\lambda : Q \times \Sigma \to \Gamma$. La fonction rationelle réalisée par une machine de Mealy est définie de manière très similaire à celle pour les machines de Moore (il suffit juste de tenir comptre de la lettre de Σ lue à chaque étape du calcul)

3.2 Equivalence avec Moore

Le théorème suivant montre que malgré une plus grande expressivité à première vue, les machines de Mealy sont en fait aussi expressives que les machines de Moore.

Théorème 3. *Pour toute machine de Mealy il existe une machine de Moore équivalente (implémentant la même fonction) et réciproquement*

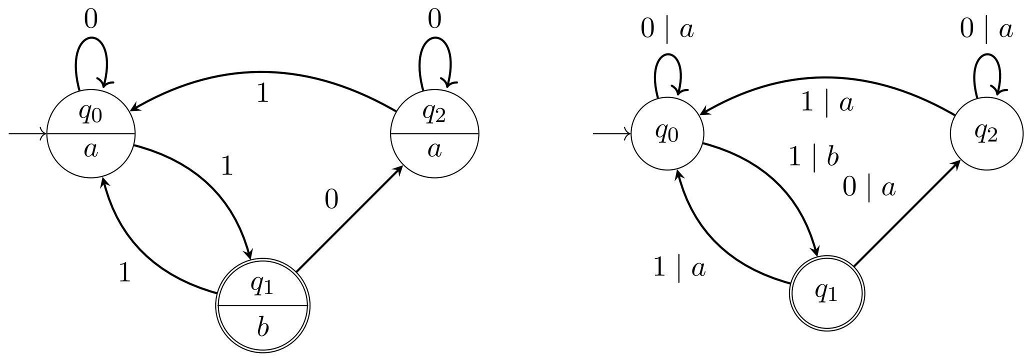
Démonstration. Considérons d'abord le sens réciproque. Soit $\mathcal{M}_1 = (\Sigma, \Gamma, Q, q_0, F, \delta, \lambda)$ une machine de Moore et notons f la fonction rationelle implémentée par $\mathcal{M}_1$. La machine de Mealy $\mathcal{M}_2 = (\Sigma, \Gamma, Q, q_0, F, \delta, \lambda')$ où λ' est définie de la manière suivante

$$\forall q \in Q, \forall a \in \Sigma, \lambda'(q, a) = \lambda(\delta(q, a))$$

implémente la fonction f. Notons f' la fonction implémentée par $\mathcal{M}_2$. On note d'abord que le langage rationnel L reconnu par l'automate sous jacent est le même pour les 2 machines. En effet les deux machines effectuent les mêmes calculs (les mêmes suites d'états sont produites en lisant les même mots). Puis pour $w \in L$, considérons le calcul $q_0, ..., q_n$ des machines sur w, avec $w = w_1, ..., w_n$. On a que

$$f'(w) = \lambda'(q_0, w_1)...\lambda'(q_{n-1}, w_n) = \lambda(\delta(q_0, w_1))....\lambda(\delta(q_{n-1}, w_n)) = q_0 \star w = f(w)$$

FIGURE 7 – Exemple de construction d'une machine de Mealy équivalente à une machine de Moore
Une machine de Moore à gauche et à droite la machine de Mealy correspondante



On montre maintenant le sens direct. Soit $\mathcal{M}_1 = (\Sigma, \Gamma, Q, q_0, F, \delta, \lambda)$ une machine de Mealy. On suppose quitte à rajouter un état que l'état initial est inacessible depuis n'importe quel autre état. On pose la machine de Moore $\mathcal{M} = (\Sigma, \Gamma, (Q \backslash q_0) \times \Gamma \cup \{q_0\}, F \times \Gamma, \delta', \lambda')$ où λ' et δ' sont définis comme suit

$$\begin{aligned}
&\forall a \in \Sigma, \delta'(q_0, a) = \delta(q_0, a) \\
&\forall q \in Q \backslash q_0, \forall b \in \Gamma, \forall a \in \Sigma, \delta'(q, b, a) = (\delta(q, a), \lambda(q, a)) \\
&\forall q \in Q \backslash q_0, \forall b \in \Gamma, \lambda'(q, b) = b
\end{aligned}$$

Notons d'abord que $w \in \Sigma^*$ est reconnu par l'automate d'entrée de $\mathcal{M}_1$, si et seulement si il l'est par l'automate d'entrée de $\mathcal{M}$: en effet, un calcul $q_0, ..., q_n$ dans le premier automate correspond à un calcul $q_0, (q_1, b_1), ..., (q_n, b_n)$ dans le second automate avec $b_1, ..., b_n \in \Gamma$ et réciproquement. Soit L le langage reconnu par les deux automates d'entrées, f la fonction implémentée par $\mathcal{M}_1$ et F celle implémentée par $\mathcal{M}$. Pour $w \in L$, $w = w_1...w_n$ et produisant le calcul $q_0, (q_1, b_1), ..., (q_n, b_n)$ dans $\mathcal{M}$

$$\begin{aligned}
F(w) &= q_0 \star w \\
&= \lambda'(q_1, b_1)...\lambda'(q_n, b_n) \\
&= b_1...b_n \\
&= \lambda(q_0, w_1)...\lambda(q_{n-1}, w_n) \\
&= f(w)
\end{aligned}$$

Donc les deux transducteurs sont bien équivalents

□

3.3 Propriétés algébriques

Définition 8 (Demi-groupe). *Soit G un ensemble muni d'une loi de composition interne $\cdot$. Si $\cdot$ est associative, on appelle G un demi-groupe.*

Définition 9 (Taux de croissance). *Si $(G, \cdot)$ est un demi-groupe et $A \subset G$ un ensemble fini de générateurs, on peut définir son taux de croissance $\gamma : \mathbb{N}\backslash\{0,1\} \to \mathbb{N}$ de la façon suivante : pour $n \geq 2$, $\gamma(n)$ est le nombre d'élements de G étant produits de n générateurs mais n'étant pas soi même un générateur. En d'autres termes*

$$\gamma(n) = |\{g_1...g_n \mid g_1, ..., g_n \in A\}\backslash A|$$

Si Σ est un alphabet fini, Σ^* est un monoïde donc un demi-groupe. Son taux de croissance est exponentiel : en effet on a ici $\gamma(n) = |\Sigma|^n$, $n \geq 2$ car tout mot de longueur au moins 2 n'est pas une lettre. Si l'on ajoute à Σ les lettres a^{-1} pour tout $a \in \Sigma$ et que l'on quotiente Σ^* par la relation d'équivalence rendant équivalent les produits aa^{-1}, $a^{-1}a$ et ε pour tout $a \in \Sigma$, on obtient alors un groupe non commutatif. Son taux de croissance est également exponentiel : on compte tout les mots possibles de 2 à n lettres tels que chaque lettre ne peut être suivie de son inverse, donc

$$\begin{aligned}\gamma(n) &= \sum_{k=2}^{n}(2|\Sigma|)(2|\Sigma|-1)^{k-1} \\ &\sim 2|\Sigma|(2|\Sigma|-1)^n\end{aligned}$$

Le demi-groupe libre $\mathbb{N}^k\backslash 0$ ayant pour générateurs la base canonique a un taux de croissance polynômial : on a pour $n \geq 2$

$$\begin{aligned}\gamma(n) &= |\{g_1 + ... + g_n \mid g_1, ..., g_n \in \{e_1, ..., e_k\}\}| \\ &= \left|\left\{(m_1, ..., m_k) \in \mathbb{N} \mid \sum_{i=1}^{k} m_i = n\right\}\right| \\ &\leq n^k\end{aligned}$$

On se demande alors si il existe des demi-groupes tels que leur taux de croissance soit compris strictement entre polynômial et exponentiel, et si les groupes, plus stricts que les demi-groupes, peuvent également réaliser ces conditions. Les machines de Mealy permettent de répondre à ces deux questions

On se fixe une machine de Mealy $\mathcal{M} = (\{0,1\}, \{0,1\}, Q, \delta, \lambda)$ (on ignore ici les états finaux et initiaux) pour la suite de cette section. Pour tout $q \in Q$, on définit

$$\rho_q : \begin{array}{ll} \{0,1\}^* & \longrightarrow \{0,1\}^* \\ w & \longmapsto q \star w \end{array}$$

On notera $G_{\mathcal{M}}$ le demi-groupe engendré par les $(\rho_q)_{q\in Q}$ avec l'opération de composition. La machine de Mealy figure 8, engendre un groupe $G_{\mathcal{M}}$ ayant un taux de croissance intermédiaire, entre polynômial et exponentiel, qui est le suivant [4]

$$\gamma(n) \sim 2^{5/2}3^{3/4}\pi^{-2}n^{1/4}\exp(\pi\sqrt{n/6})$$

Un autre exemple est l'automate de la figure 9 [2], dont le demi-groupe engendré est en fait un groupe (de Grigorchuk), et ayant lui aussi un taux de croissance intermédiaire [3].

Figure 8 – Une machine de Mealy produisant un demi-groupe à croissance intermédiaire

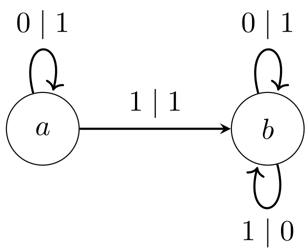
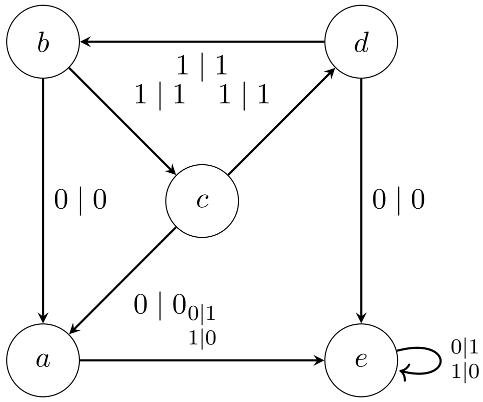


Figure 9 – L'automate de Mealy engendrant le groupe de Grigorchuk



Références

[1] T. V. Griffiths. The unsolvability of the equivalence problem for a-free nondeterministic generalized machines. *Journal of the Association for Computing Machinery*, 15 :409–413, 1968.

[2] Rostislav I. Grigorchuk. On burnside's problem on periodic groups. *Funktsional. Anal. i Prilozheniya*, 14(1), 1980.

[3] Rostislav I. Grigorchuk. Degrees of growth of finitely generated groups and the theory of invariant means. *Izv. Akad. Nauk SSSR Ser. Mat.*, 48(5), 1984.

[4] Ilya I. Reznykov et Vitalii I. Sushchansky Laurent Bartholdi. The smallest mealy automaton of intermediate growth. *Journal of Algebra*, 295(2), 2006.

[5] Jacques Sakarovitch. *Elements de théorie des automates*. 2003.