Leçon 1

Le modèle « transducteur fini »

Dans cette leçon, on introduit le modèle des automates finis « avec sortie », qu'on appelle « transducteurs », et qui sont des automates (finis) sur des produits directs de monoïdes libres. Ce modèle de calcul est équivalent aux machines de Turing unidirectionnelles à 2 bandes (ou plus). On montre la fermeture par composition des relations de mots réalisées par les transducteurs.

Contents

Le fait de considérer des automates « avec sortie » est une extension très naturelle, voire nécessaire, des automates qui lisent des suites de symboles. Dès le début de la théorie des automates finis (*i.e.* la fin des années cinquante), de telles machines ont été définies et étudiées : *machines de Moore*, où on observe la suite (final, non final, ...) des états atteints au cours de la lecture d'un mot, *machines de Mealy*, qui associe une sortie à chaque transition. Les deux modèles sont en fait équivalents, moyennant quelques aménagements. On va partir d'un modèle plus général, qui permet de mettre en place une théorie plus riche.

1.1 Premières définitions

En un mot (en cinq, en fait), un transducteur est un automate dont les transitions sont étiquetées par des couples de mots. On commence par rappeler la définition des automates « classiques » et les notions qui y sont directement rattachées.

Dans la suite, A est un *alphabet*, ensemble fini d'éléments appelés lettres. Le *monoïde libre* engendré par A, noté A^*, est l'ensemble des suites finies de lettres, appelées *mots*, muni de la concaténation comme produit. La suite vide, ou *mot vide*, notée 1_{A^*}, ou même 1, est l'élément neutre de A^*.

1.1.1 Automates sur un monoïde libre

Définition Un *automate* $\mathcal{A}$ *sur* A, ou *sur* A^*, est un graphe orienté dont les arêtes, appelées *transitions*, sont étiquetées par des lettres et dans lequel sont distingués deux sous-ensembles de sommets (appelés *états*).

Plus précisément, l'automate $\mathcal{A}$ sur A est spécifié par la donnée d'un ensemble Q d'états, de deux sous-ensembles I et T de Q, respectivement ensembles des *états initiaux* et des *états finals* et de l'ensemble $E \subseteq Q \times A \times Q$ des transitions. On note $\mathcal{A} = \langle A, Q, I, E, T \rangle$ ou $\mathcal{A} = \langle Q, I, E, T \rangle$ si l'alphabet A est implicite. Graphes, les automates ont une représentation graphique naturelle. Comme sur la Figure 1, les états initiaux sont repérés par une flèche entrante, les états finals par une flèche sortante. Dans le même esprit, une transition (p, a, q) de $\mathcal{A}$ est notée $p \xrightarrow{a} q$, ou $p \xrightarrow[\mathcal{A}]{a} q$ si on doit préciser l'automate.

Un automate $\mathcal{A}$ est *fini* si E est fini et donc, avec l'hypothèse que A est fini et la définition que nous avons prise, si Q est fini.

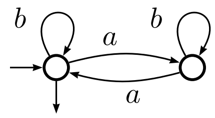


FIG. 1 – Automate $\mathcal{E}_1$

Calculs Un calcul c de $\mathcal{A}$ est une suite de transitions telles que l'origine de chacune coïncide avec la destination de la précédente, ce que l'on note ainsi :

$$c = p_0 \xrightarrow{a_1} p_1 \xrightarrow{a_2} p_2 \cdots p_{n-1} \xrightarrow{a_n} p_n \quad .$$

L'état p_0 est l'*origine* du calcul c, p_n sa *destination.* L'*étiquette* d'un calcul est le produit des étiquettes des transitions qui le composent et on écrit :

$$c = p_0 \xrightarrow{a_1 a_2 \cdots a_n} p_n \quad .$$

Un calcul est *réussi* si son origine est un état initial et sa destination un état final. Un mot de A^* est *accepté*, ou *reconnu*, par $\mathcal{A}$ s'il est l'*étiquette d'un calcul réussi.*

Langage accepté Le *langage accepté*, ou *reconnu*, par $\mathcal{A}$, noté $L(\mathcal{A})$, est l'ensemble des mots acceptés par $\mathcal{A}$:

$$L(\mathcal{A}) = \left\{ w \in A^* \;\middle|\; \exists i \in I\,,\ \exists t \in T \quad i \xrightarrow[\mathcal{A}]{w} t \right\} \quad .$$

Par exemple le langage de l'automate de la Figure 1 est l'ensemble des mots qui contiennent un nombre pair de 'a'.

Deux automates sont *équivalents* s'ils acceptent le même langage.

La famille des langages acceptés par les *automates finis* est celle des *langages rationnels* qui jouit de nombreuses propriétés, supposées pour l'essentiel connues des lecteurs.

Modèle généralisé Avant de généraliser avec les transducteurs ce modèle d'automates finis, il faut rappeler une généralisation qui ne change pas la puissance du modèle. A condition de n'appeler *finis* que les automates dont l'ensemble des *transitions* (et pas seulement des *états*) est fini, on peut considérer des automates dont les transitions sont étiquetée par des *mots* de A^*.

On se ramène au modèle précédent des automates dont les transitions sont étiquetées par des lettres en deux étapes. On remplace d'abord chacune des transitions étiquetées par un mot de longueur ℓ strictement supérieure à 1 en une suite de ℓ transitions étiquetées par des lettres et dont les extrémités sont $\ell - 1$ nouveaux états en plus des extrémités de la transition en cause. On obtient ainsi un automate dont les transitions sont étiquetées soit par une lettre, soit par le mot vide, et qu'on appelle alors *transitions spontanées*.

Qu'on puisse éliminer ces dernières sans changer le langage accepté (ni d'ailleurs l'ensemble des états) est un résultat classique qu'on réétablit ci-dessous dans un cadre plus général (Théorème 9).

1.1.2 Transducteurs

Dans la suite, B est un autre alphabet. L'ensemble $A^* \times B^*$ des couples (u, v) avec u dans A^* et v dans B^*, muni de l'opération :

$$(u, v)\,(u', v') = (u\,u', v\,v')$$

est un monoïde, dont l'élément neutre est $(1_{A^*}, 1_{B^*})$, le plus souvent noté $(1, 1)$. La *longueur* d'un élément de $A^* \times B^*$ est la somme des longueurs de ses composantes : $|(u, v)| = |u| + |v|$.

Définition 1. *Un* transducteur *est un automate sur $A^* \times B^*$ c'est-à-dire un automate dont les transitions sont étiquetées par des couples de mots.*

On note donc un transducteur $\mathcal{T} = \langle\, A^* \times B^*, Q, I, E, T \,\rangle$ où, comme précédemment, Q est l'ensemble des états, I et T les ensembles des états initiaux et finals respectivement, et où $E \subseteq Q \times (A^* \times B^*) \times Q$ est l'ensemble des transitions. La Figure 2 montre quatre transducteurs.

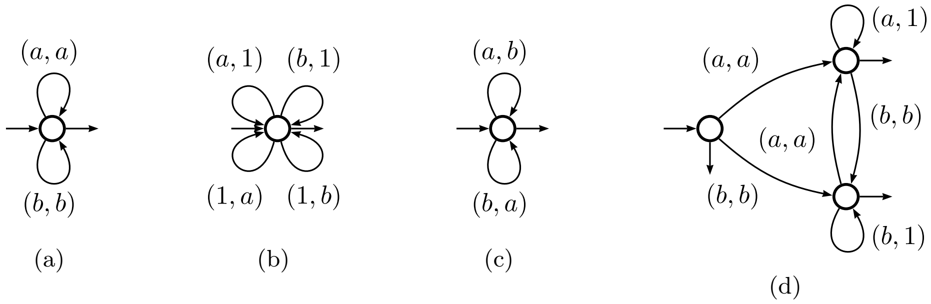


FIG. 2 – Quatre transducteurs

On note $p \xrightarrow{(u,v)} q$ une transition et

$$c = p_0 \xrightarrow{(u_1,v_1)} p_1 \xrightarrow{(u_2,v_2)} p_2 \cdots p_{n-1} \xrightarrow{(u_n,v_n)} p_n$$

un calcul de $\mathcal{T}$. L'étiquette d'un calcul est le produit des étiquettes des transitions qui le composent et on écrit :

$$c = p_0 \xrightarrow{(u_1 u_2 \cdots u_n, v_1 v_2 \cdots v_n)} p_n \quad .$$

Un calcul est *réussi* si son origine est un état initial et sa destination un état final. Un couple de mots (u,v) de $A^* \times B^*$ est accepté par $\mathcal{T}$ s'il est l'étiquette d'un calcul réussi de $\mathcal{T}$.

Définition 2. *Le* comportement *d'un transducteur* $\mathcal{T} = \langle A^* \times B^*, Q, I, E, T \rangle$, *noté* $|\mathcal{T}|$, *est l'ensemble des couples de mots acceptés par* $\mathcal{T}$ *:*

$$|\mathcal{T}| = \left\{ (u,v) \in A^* \times B^* \;\middle|\; \exists i \in I\,,\ \exists t \in T \quad i \xrightarrow[\mathcal{T}]{(u,v)} t \right\} \quad .$$

Exemples 3. Le transducteur de la Fig. 2 (a) accepte l'ensemble des couples (u,u) où u est un mot quelconque de $\{a,b\}^*$; en (b) il accepte l'ensemble des couples (u,v) où u et v sont des mots quelconques de $\{a,b\}^*$; en (c) l'ensemble des couples (u,v) où u est un mot quelconque de $\{a,b\}^*$ et où v est obtenu à partir de u en remplaçant les 'a' par des 'b' et les 'b' par des 'a' ; en (d) l'ensemble des couples (u,v) où u est un mot quelconque de $\{a,b\}^*$ et où v est obtenu à partir de u en remplaçant les blocs de 'a' par un seul 'a', les blocs de 'b' par un seul 'b'.

Le comportement d'un transducteur est donc un sous-ensemble de $A^* \times B^*$, ce que l'on va appeler relation de mots.

1.1.3 Relations de mots

Relations Une relation θ de A^* dans B^* qu'on note (abusivement) $\theta \colon A^* \to B^*$ est *définie par son graphe* $\widehat{\theta} \subseteq A^* \times B^*$. Par définition, une relation de A^* dans B^*

associe à chaque mot de A^* un sous-ensemble de B^* :

$$\forall u \in A^* \quad \theta(u) = \left\{ v \in B^* \,\middle|\, (u,v) \in \widehat{\theta} \right\} \quad .$$

L'idée sous-jacente est que la première composante d'un couple (u,v) est une « donnée »— *input* en anglais, traduit en général par *entrée* — et que la deuxième composante est le « résultat »— *output* en anglais retraduit en général par *sortie*. Ce point de vue donne deux rôles distincts aux deux composantes du couple, mais ne rompt pas la symétrie.

Inverse En effet, A^* et B^* jouent des rôles symétriques par l'intermédiaire du graphe de θ et on définit la *relation inverse* de θ :

$$\theta^{-1} \colon B^* \to A^* \quad ,$$

comme la relation de B^* dans A^* qui a le même graphe que θ, modulo l'identification canonique entre $A^* \times B^*$ et $B^* \times A^*$:

$$\forall v \in B^* \qquad \theta^{-1}(v) = \left\{ u \in A^* \,\middle|\, (u,v) \in \widehat{\theta} \right\} \quad .$$

Domaine Si $\theta \colon A^* \to B^*$ est une relation, le *domaine* et l'*image* de θ sont les projections de $\widehat{\theta}$ sur A^* et B^* respectivement :

$$\mathsf{Dom}\,\theta = \left\{ u \in A^* \,\middle|\, \exists v \in B^* \quad (u,v) \in \widehat{\theta} \right\} \qquad \text{et}$$
$$\mathsf{Im}\,\theta = \left\{ v \in B^* \,\middle|\, \exists u \in A^* \quad (u,v) \in \widehat{\theta} \right\} \quad .$$

Bien entendu, $\mathsf{Dom}\,\theta^{-1} = \mathsf{Im}\,\theta$ et $\mathsf{Im}\,\theta^{-1} = \mathsf{Dom}\,\theta$. Il vient également que $u \notin \mathsf{Dom}\,\theta$ si, et seulement si, $\theta(u) = \emptyset$.

Additivité *Par définition*, une relation s'étend par *additivité* :

$$\forall L \subseteq A^* \quad \theta(L) = \bigcup_{u \in L} \theta(u)$$

et devient ainsi une *application de* $\mathfrak{P}(A^*)$ *dans* $\mathfrak{P}(B^*)$. La notion de relation porte donc en elle de façon implicite la propriété d'additivité. *A contrario*, la complémentation par exemple, qui à un sous-ensemble de A^* associe aussi un sous-ensemble de A^* *n'est pas une relation* de A^* dans lui-même.

Complément Le *complément* d'une relation $\theta \colon A^* \to B^*$ est la relation $\complement\theta \colon A^* \to B^*$ dont le graphe est le complément dans $A^* \times B^*$ du graphe de θ :

$$\widehat{\complement\theta} = \complement\widehat{\theta} \quad ,$$

et pour tout u dans A^*, $[\complement\theta](u) = \complement_{B^*}\theta(u)$.

1.1.4 Relations réalisées par transducteurs

Si un couple (u,v) appartient au comportement d'un transducteur $\mathcal{T}$, on considère intuitivement que la première composante u, l'*entrée*, est « lue » par $\mathcal{T}$ et que la deuxième composante v, la *sortie*, est le « résultat » d'un calcul de $\mathcal{T}$.

Comme le passage du graphe à la relation, ce point de vue différencie les rôles des deux composantes. Il sera accentué dans des leçons ultérieures, où la symétrie même sera brisée.

Puisque qu'un transducteur sur $A^* \times B^*$ définit par son comportement un sous-ensemble de $A^* \times B^*$, on peut dire qu'il « réalise » une relation de A^* dans B^*. Inversement, une relation de A^* dans B^* peut être, on ne pas être, réalisée par un transducteur de telle ou telle sorte.

En particulier, de même qu'un langage de A^* est accepté par un automate *fini* sur A^* si, et seulement si, il est *rationel*, on verra, à la leçon suivante, qu'une relation de A^* dans B^* est réalisée par un transducteur fini sur $A^* \times B^*$ si, et seulement si, son graphe est une partie rationnelle de $A^* \times B^*$. Pour cette leçon, nous nous en tiendrons à la définition :

Définition 4. *Une relation* $\theta \colon A^* \to B^*$ *est dite* réalisée par un transducteur fini *(*RTF*, pour faire court) s'il existe un transducteur fini* $\mathcal{T}$ *sur* $A^* \times B^*$ *dont le comportement est égal au graphe de* θ *:* $|\mathcal{T}| = \widehat{\theta}$ *.*

Et on dira désormais « relation réalisée » par $\mathcal{T}$ plutôt que « comportement » de $\mathcal{T}$.

Des transformations simples sur les transducteurs donnent deux premiers résultats. Si on échange la première et la deuxième composante de l'étiquette de chaque transition d'un transducteur $\mathcal{T}$, on obtient un transducteur qui réalise la relation inverse de la relation réalisée par $\mathcal{T}$, d'où l'on conclut :

Proposition 5. *L'inverse d'une relation* RTF *est une relation* RTF. ■

Si on projette l'étiquette de chaque transition d'un transducteur $\mathcal{T}$ sur sa première (resp. sa deuxième) composante, on obtient un automate dont les transitions sont étiquetées par des mots et qui accepte le domaine (resp. l'image) de la relation réalisée par $\mathcal{T}$, d'où l'on conclut :

Proposition 6.
Le domaine (resp. l'image) d'une relation RTF *est un langage rationnel.* ■

1.2 Travail sur le modèle

Pour travailler efficacement avec les transducteurs, il est utile d'avoir une définition plus restreinte sans diminuer la puissance du modèle, et aussi l'enrichir sans augmenter la puissance.

1.2.1 Normalisation

L'alphabet A « engendre » *librement* A^* puisque chaque mot de A^* est le produit d'une suite *unique* de lettres de A. L'ensemble $(A{\times}1_{B^*})\cup(1_{A^*}{\times}B)$ engendre $A^*\times B^*$ puisque chaque couple de $A^*\times B^*$ est le produit de suites d'éléments de $(A{\times}1_{B^*})\cup(1_{A^*}\times B)$, mais ces suites ne sont pas (en général) uniques :

$$(ab,bab) = (a,1)\,(1,b)\,(1,a)\,(b,1)\,(1,b) = (1,b)\,(a,1)\,(b,1)\,(1,a)\,(1,b) \quad .$$

On peut également prendre $(A{\times}1_{B^*})\cup(1_{A^*}{\times}B)\cup(A{\times}B)$ comme ensemble générateur de $A^*\times B^*$, cela permet d'avoir des suites de décomposition plus courtes :

$$(ab,bab) = (a,b)\,(b,a)\,(1,b) = (1,b)\,(a,a)\,(b,b) \quad .$$

Les automates sur A^* sont définis avec des transitions étiquetées dans A et on montre que le modèle n'est pas plus puissant si on prend les étiquettes dans A^* tout entier. Pour les transducteurs, on suit le processus à rebours : ils sont définis avec des transitions étiquetées dans $A^*\times B^*$ tout entier, et on va montrer que le modèle n'est pas moins puissant si on restreint l'ensemble des étiquettes possibles.

Définition 7. (i) *Un transducteur sur $A^*\times B^*$ dont les étiquettes sont dans $(A\times 1_{B^*})\cup(1_{A^*}\times B)$ est appelé transducteur* normalisé*.*

(ii) *Un transducteur sur $A^*\times B^*$ dont les étiquettes sont dans $(A\times 1_{B^*})\cup(1_{A^*}\times B)\cup(A\times B)$ est appelé transducteur* sous-normalisé*.*

Les transducteurs (a), (b) et (d) de la Figure 2 sont sous-normalisés, le transducteur (b) est normalisé.

Proposition 8.
Tout transducteur est équivalent à un transducteur normalisé (ou sous-normalisé).

On commence par procéder comme pour les automates sur A^* : chaque transition dont l'étiquette (u,v) est de longueur $\ell = |u| + |v|$ supérieure à 1 est remplacée par $\ell - 1$ transitions étiquetées dans $(A\times 1_{B^*})\cup(1_{A^*}\times B)$ ou par k transitions étiquetées dans $(A\times 1_{B^*})\cup(1_{A^*}\times B)\cup(A\times B)$, avec k compris entre $\mathsf{max}(|u|,|v|)$ et $\ell - 1$.

Pour obtenir un transducteur normalisé, ou sous-normalisé, il faut alors éliminer les transitions qui sont étiquetées par $(1,1)$, l'élément neutre de $A^*\times B^*$. C'est le résultat d'un algorithme classique, qui se décrit dans un cadre général et qui vaut d'être explicité car il sera encore utilisé dans une construction ultérieure.

Soit M un monoïde. Un *automate sur M* est un graphe dont les transitions sont étiquetées par des éléments de M. Un tel automate est dit *propre* si aucune de ses transition n'est étiquetée par l'élément neutre de M.

Théorème 9. *Tout automate fini sur M est équivalent à un automate fini* propre*.*

Démonstration. Soit $\mathcal{A} = \langle\, M, Q, I, E, T \,\rangle$ un automate sur M. On écrit : $E = F \cup S$ où S est l'ensemble des transitions *spontanées* de $\mathcal{A}$, *i.e.* étiquetées par 1_M. Sans perte de généralités, on peut supposer que S est un sous-graphe *transitif* de $\mathcal{A}$: en effet, rajouter les transitions correspondant à la fermeture transitive changera les calculs de $\mathcal{A}$, mais pas leurs étiquettes. Un calcul de $\mathcal{A}$ est de la forme :

$$c = p_0 \xrightarrow{m_1} p_1 \xrightarrow{m_2} p_2 \cdots p_{n-1} \xrightarrow{m_n} p_n \quad ,$$

et on peut supposer, grâce à l'hypothèse sur S, que deux m_i consécutifs ne sont jamais égaux à 1_M. On définit l'automate $\mathcal{B} = \langle\, M, Q, J, G, T \,\rangle$ par :

$$G = F \,\cup\, \{(p, m, r) \mid \exists q \in Q \quad (p, m, q) \in F \text{ et } (q, 1_M, r) \in S\} \quad \text{et}$$
$$J = I \,\cup\, \{j \mid \exists i \in I \quad (i, 1_M, j) \in S\}$$

dont on vérifie qu'il est équivalent à $\mathcal{A}$. ∎

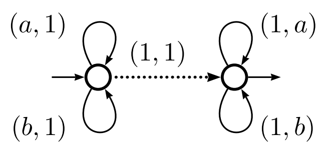
Cette construction achève d'établir la Proposition 8. ∎

L'intérêt de ce résultat est aussi de permettre l'utilisation de transitions spontanées pour une description succincte de transducteurs.

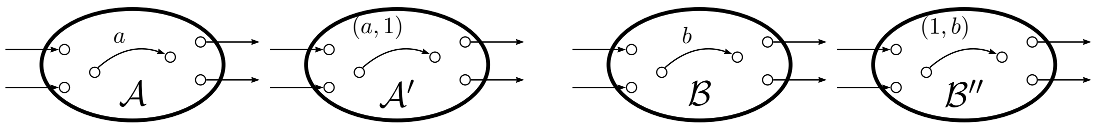
Remarque 10. La construction décrite dans la preuve du Théorème 9 peut être appelée une procédure de « fermeture avant » puisque les nouvelles transitions sont obtenues en considérant les transitions spontanées qui *suivent* des transitions étiquetées par des éléments différents de l'élément neutre. On peut également construire un automate propre équivalent à $\mathcal{A}$ par une procédure de « fermeture arrière ».

Exemples 11. (i) **Relation universelle, produit direct de rationnels**.

La relation universelle, *i.e.* la relation dont le graphe est $A^* \times A^*$ tout entier, est réalisée par le transducteur de la Figure 2 (b). Elle est aussi réalisée par le transducteur ci-dessous, dans lequel chaque élément de $A^* \times A^*$ est l'étiquette d'un seul calcul (et où on voit l'intérêt des transitions spontanées).



Si K est un langage de A^*, accepté par l'automate $\mathcal{A}$, et L un langage de B^*, accepté par l'automate $\mathcal{B}$, on transforme respectivement $\mathcal{A}$ en un transducteur $\mathcal{A}'$ en remplaçant l'étiquette 'a' de chaque transition en '$(a, 1)$' et $\mathcal{B}$ en un transducteur $\mathcal{B}''$ en remplaçant l'étiquette 'b' de chaque transition en '$(1, b)$' comme représenté ci-dessous.



Le transducteur de la Figure 3 réalise la relation dont le graphe est $K \times L$.

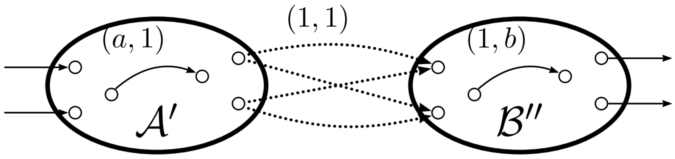
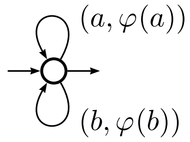


Fig. 3 – Un transducteur pour $K \times L$

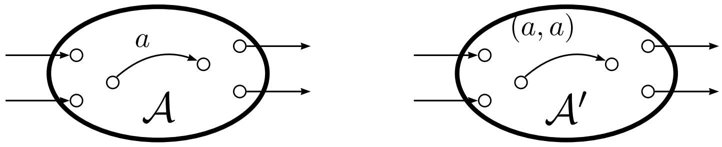
(ii) **Identité, morphismes**. L'identité, c'est-à-dire la relation dont le graphe est $\{(w, w) \mid w \in A^*\}$, est réalisée par le transducteur de la Figure 2 (a). Un *morphisme* $\varphi\colon A^* \to B^*$ est réalisé par le transducteur ci-dessous.



(iii) **Intersection avec un rationnel**. Si K est un langage de A^*, l'*intersection avec* K est une relation de A^* dans lui-même, notée ι_K, et définie par :

$$\forall w \in A^* \qquad \iota_K(w) = \begin{cases} \quad w & \text{si } w \in K\,; \\ \text{indéfini (ou } \emptyset) & \text{sinon}. \end{cases} .$$

Si K est accepté par un automate $\mathcal{A}$, la relation ι_K est réalisée par le transducteur $\mathcal{A}'$ obtenu à partir de $\mathcal{A}$ en remplaçant l'étiquette 'a' de chaque transition en '$(a, 1)$', comme représenté ci-dessous.



(iv) **Facteurs, sous-mots**. La relation de A^* dans lui-même qui à un mot associe ses facteurs est réalisée par le transducteur représenté à la Figure 4 (a) ; celle qui lui associe ses sous-mots par le transducteur représenté à la Figure 4 (b).

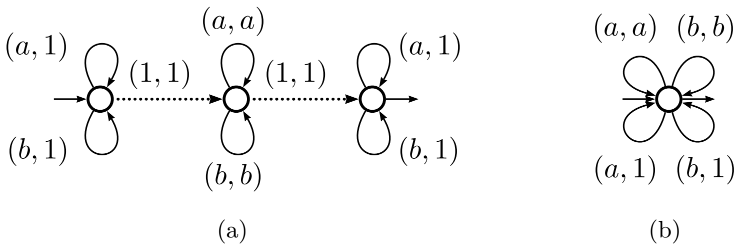


Fig. 4 – Facteurs et sous-mots

(vi) **Opérations sur les nombres représentés en base** p. Si on choisit une base p, les nombres (entiers positifs) sont écrits[1] sur l'alphabet $A_p = \{0, 1, \ldots, p-1\}$ et les opérations sur les nombres sont des fonctions de A_p^*, ou $(A_p^*)^2$, ou $(A_p^*)^3$, *etc.* dans A_p^*. Certaines sont réalisées par des transducteurs. La Figure 5 donne l'exemple de la division (entière) par un entier k fixé, dans le cas où $p = 2$ et $k = 3$.

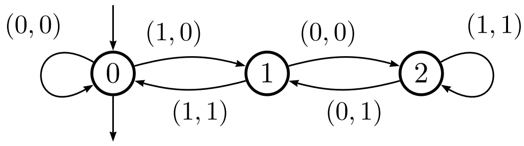


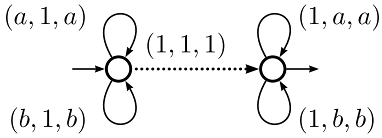
Fig. 5 – Division entière par 3 des nombres écrits en binaire

1.2.2 Extensions

Une fois qu'on a généralisé les automates classiques en prenant les étiquettes dans un produit direct $A^* \times B^*$ plutôt que dans un monoïde libre, il ne coûte pas grand chose de plus de considérer les automates dont les transitions sont étiquetées par les éléments d'un produit direct de k monoïdes libres $A_1^* \times A_2^* \times \cdots \times A_k^*$ — k arbitraire mais fixé — et qu'on appelle également *transducteurs*.

Le *comportement* d'un tel transducteur est une partie de $A_1^* \times A_2^* \times \cdots \times A_k^*$ qui est le graphe d'une *relation sur* $A_1^* \times A_2^* \times \cdots \times A_k^*$, relation que l'on dit *réalisée* par le transducteur. Il y a de multiples façons de « curryfier » la relation. Il est courant de voir les $k-1$ premières composantes d'un k-uplet comme les données et la k-ième commme le résultat, c'est-à-dire de voir la relation comme une application de $A_1^* \times A_2^* \times \cdots \times A_{k-1}^*$ dans $\mathfrak{P}(A_k^*)$. Mais on peut également la voir comme une application de A_1^* dans $\mathfrak{P}(A_2^* \times \cdots \times A_k^*)$ (*cf.* Leçon 5).

Exemple 12. Produit dans A^*. La relation $\pi \colon A^* \times A^* \to A^*$ qui à chaque couple de mots associe leur produit : $\pi(u, v) = uv$ pour tout u, v dans A^* est réalisée par le transducteur ci-dessous.



On définit de façon évidente les notions de transducteur normalisé ou sous-normalisé (comme serait le transducteur ci-dessus si on élimine la transition spontanée) et tout transducteur est équivalent à un transducteur normalisé ou sous-normalisé. Il n'est pas inutile d'avoir à l'esprit qu'une telle extension de 2 à k monoïdes libres est possible, mais dans la suite de ce cours nous ne considérerons presqu'exclusivement que des tranducteurs sur $A^* \times B^*$.

[1]Quand on utilise des alphabets de chiffres, le mot vide est représenté par ϵ.

Une deuxième variation sur le modèle des transducteurs concerne le *sens de lecture*. En effet, quand nous avons écrit que l'*étiquette d'un calcul est le produit des étiquette des transitions* qui constituent ce calcul (qu'il s'agisse d'un automate ou d'un transducteur), nous avons tenu pour évident que ce produit se fasse de gauche à droite. Cela correspond, dans le modèle de machine décrit dans la sous-section suivante, à une lecture *de gauche à droite*. La convention inverse aurait été tout aussi légitime. Il y a même des cas, comme celui de l'Exemple 13 ci-dessous, où elle peut sembler plus naturelle.

On peut utiliser la *transposition* pour résoudre ce problème. Le *transposé*, ou *image miroir*, d'un mot $w = a_1 a_2 \cdots a_n$ est le mot ${}^{\mathsf{t}}w = a_n \cdots a_2 a_1$; le transposé d'un couple (u, v) est le couple $({}^{\mathsf{t}}u\ , {}^{\mathsf{t}}v)$. Le transposé d'un automate, ou transducteur, $\mathcal{A} = \langle\, Q, I, E, T\,\rangle$ est l'automate, ou transducteur, ${}^{\mathsf{t}}\mathcal{A} = \langle\, Q, T, {}^{\mathsf{t}}E\ , I\,\rangle$, avec

$$ {}^{\mathsf{t}}E = \{(p, {}^{\mathsf{t}}x\ , q) \mid (q, x, p) \in E\} \quad . $$

Un mot w est accepté par $\mathcal{A}$ dans une *lecture de droite à gauche* si, et seulement si, ${}^{\mathsf{t}}w$ est accepté par ${}^{\mathsf{t}}\mathcal{A}$ dans une lecture de gauche à droite. Un mot v appartient à l'image d'un mot u dans la relation réalisée par un transducteur $\mathcal{A}$ dans une *lecture de droite à gauche* si, et seulement si, ${}^{\mathsf{t}}v$ appartient à l'image de ${}^{\mathsf{t}}v$ par ${}^{\mathsf{t}}\mathcal{A}$ dans une lecture de gauche à droite.

De cette façon, l'inversion du sens de lecture ne change pas la puissance du modèle ou n'apporte rien de nouveau (pour autant qu'on ait à sa disposition l'opération de transposition). Dans certains cas, il peut être néanmoins plus simple, commode ou naturel de considérer des transducteurs lisent de droite à gauche, par exemple quand cela permet d'avoir un transducteur *déterministe sur l'entrée* (ce que nous appellerons *transducteur séquentiel droit* à la Leçon 7) comme à l'Exemple 13.

Exemple 13. Addition en base 2. Soit $A_2 = \{0, 1\}$. L'application qui à chaque couple (u, v) de mots de $A_2 \times A_2$, qui sont la représentation en binaire des entiers $\overline{u}$ et $\overline{v}$, associe la représentation en binaire de $\overline{u} + \overline{v}$ est réalisée par le transducteur de la Figure 6 quand il lit les couples de la droite vers la gauche (ce qui est la façon normale pour effectuer une addition) avec la convention que les deux mots u et v sont alignés à droite, que le plus court est complété par un nmbre suffisant de '0' sur la gauche pour être de la même longueur que le plus long, et que, le cas échéant, un dernier '0' est rajouté à gauche des deux mots pour permettre une dernière transition vers l'état final.

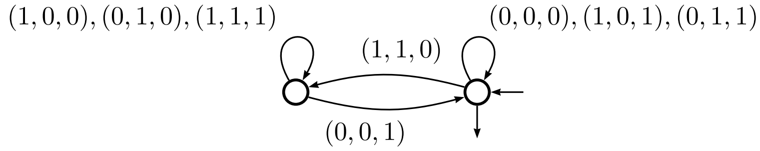


Fig. 6 – Transducteur droit à trois bandes pour l'addition en binaire

1.2.3 Les transducteurs comme machines

La modélisation d'un automate fini booléen sur A^* comme une machine de Turing "unidirectionnelle" conduit naturellement à une généralisation du modèle pour inclure plusieurs bandes.

Le dispositif est constitué d'une unité de commande dotée d'un nombre fini d'états et de *plusieurs* bandes. L'unité de commande est reliée à chacune de ces bandes par une tête de lecture (*cf.* Figure 7).

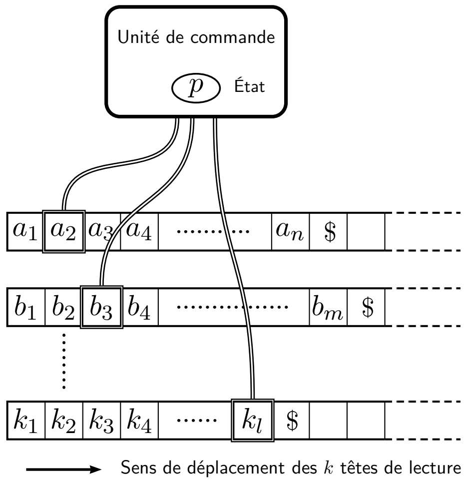


FIG. 7 – Automate à k bandes de Rabin et Scott

À chaque étape du calcul, l'unité de commande, considérant l'état p dans lequel elle se trouve, *choisit une bande* sur laquelle elle va lire, et, suivant le symbole a lu sur cette bande, passe dans un état q et *déplace la tête de lecture de cette bande* d'une case vers la droite[2]. Parce que la tête de lecture se déplace toujours *dans le même sens*, ce type de machine est appelé en anglais *1-way k-tape automaton.*

Au début d'un calcul, un mot est inscrit sur chacune des k bandes. d'entrée, L'unité de commande est dans l'état distingué *initial*, et chaque tête de lecture est placée sur la première case de chacune des k bandes Après une succession d'étapes comme celle décrite précédemment, un calcul *se termine* si toutes les têtes de lecture ont atteint sur leur bande la case dans laquelle est inscrite le *symbole de fin de bande.* Le calcul est dit *réussi* si l'état dans lequel se trouve l'unité de commande à la fin du calcul est un des états *finals.* Un k-uplet de mots est accepté par l'automate s'il peut être lu par un calcul réussi.

Les transducteurs finis sur $A^* \times B^*$ sont *fortement équivalents* aux machines de Turing unidirectionnelles à 2 bandes (1W2TTM) en ce sens que pour chaque transducteur fini on peut contruire une telle machine qui est non seulement équivalente

[2]On pourrait imaginer d'autres définitions du mode de fonctionnement, en particulier que le choix de la bande lue et de l'état d'arrivée dépende non seulement de l'état de départ mais aussi des symboles lus sur toutes les bandes. En fait, toutes ces définitions se révèlent équivalentes.

(*i.e.* accepte les mêmes couples de mots) mais est telle qu'il y a bijection entre les calculs de l'un et ceux de l'autre, et *vice versa*. La généralisation aux transducteurs sur $A_1^* \times A_2^* \times \cdots \times A_k^*$ et aux machines de Turing unidirectionnelles à k bandes est fastidieuse mais sans difficultés aucunes.

A partir de maintenant, transducteur = transducteur fini.

1.3 Composition

La composition des fonctions s'étend très directement en celle des relations.

Définition 14. *Soient* $\theta\colon A^* \to B^*$ *et* $\sigma\colon B^* \to C^*$ *deux relations. La* composée *de* θ *et* σ *est la relation*

$$\sigma \circ \theta\colon A^* \to C^* \qquad \textit{définie par} \qquad \forall u \in A^* \qquad [\sigma \circ \theta](u) = \sigma(\theta(u)) \ .$$

On vérifie que la composition des relations peut se définir (ou s'exprimer) sur leur graphe :

$$\widehat{\sigma \circ \theta} = \left\{ (u,w) \in A^* \times C^* \;\middle|\; \exists v \in B^* \quad (u,v) \in \widehat{\theta} \text{ et } (v,w) \in \widehat{\sigma} \right\} \ .$$

1.3.1 Le théorème principal

La propriété sans doute la plus remarquable des relations réalisées par transducteurs finis est leur fermeture par composition.

Théorème 15 (Elgot & Mezei 1965)**.**
La composée de deux relations RTF *est une relation* RTF*.*

Démonstration. Soient $\theta\colon A^* \to B^*$ et $\sigma\colon B^* \to C^*$ deux relations et

$$\mathcal{T} = \langle\, A^* \times B^*, Q\, I, E, T \,\rangle \quad \text{et} \quad \mathcal{S} = \langle\, B^* \times C^*, R, J, F, U \,\rangle$$

deux transducteurs *sous-normalisés* qui réalisent respectivement θ et σ. On construit le *produit de composition* des deux transducteurs $\mathcal{U} = \mathcal{T} \circ \mathcal{S}$ par :

$$\mathcal{U} = \langle\, A^* \times C^*, Q \times R, I \times J, G, T \times U \,\rangle$$

où G est la *réunion* de trois ensembles de transitions : $G = G_1 \cup G_2 \cup G_3$ avec :

$$G_1 = \Big\{ (p,r) \xrightarrow{(x,y)} (q,s) \;\Big| \; \exists b \in B\,,\ \exists p \xrightarrow{(x,b)} q \in E\,,\ \exists r \xrightarrow{(b,y)} s \in F \quad x \in A \cup 1\,,\ y \in C \cup 1 \Big\} \ ,$$

$$G_2 = \Big\{ (p,r) \xrightarrow{(a,1)} (q,r) \;\Big|\; \exists\, p \xrightarrow{(a,1)} q \in E \qquad \forall r \in R \Big\} \ ,$$

$$G_3 = \Big\{ (p,r) \xrightarrow{(1,c)} (p,s) \;\Big|\; \exists\, r \xrightarrow{(1,c)} s \in F \qquad \forall p \in Q \Big\} \ .$$

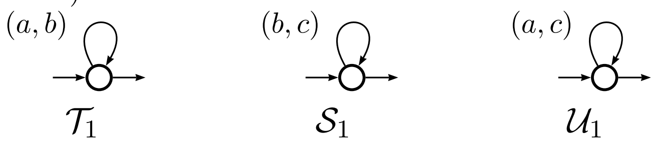
On vérifie, par récurrence sur la longueur des calculs par exemple, que :

$$(p,r) \xrightarrow[\mathcal{U}]{(u,w)} (q,s) \qquad \text{si, et seulement si,} \quad \exists v \quad p \xrightarrow[\mathcal{T}]{(u,v)} q \quad \text{et} \quad r \xrightarrow[\mathcal{S}]{(v,w)} s \ ,$$

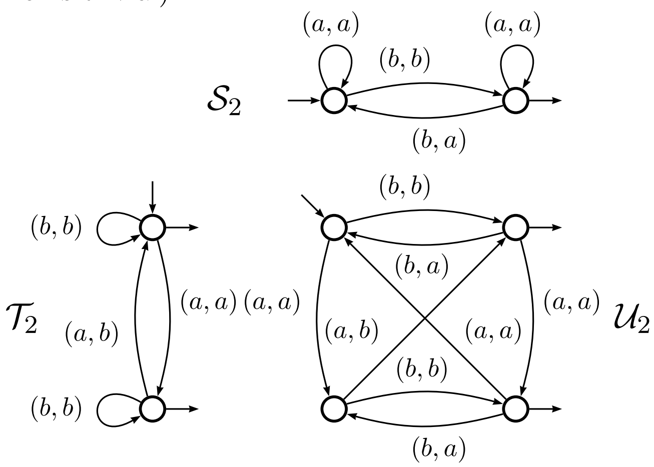
ce qui établit que $|\mathcal{U}| = \widehat{\sigma \circ \theta}$. ■

Il se peut qu'on construise ainsi un transducteur $\mathcal{U}$ dont certaines transitions sont étiquetées par $(1,1)$ (dans le premier groupe, si x *et* y sont simultanément égaux à 1). On supprime ces transitions spontanées (par « clôture arrière », ou « clôture avant », par exemple) pour obtenir un transducteur sous-normalisé. Dans la suite, c'est ce transducteur sous-normalisé qui sera noté $\mathcal{U} = \mathcal{T} \circ \mathcal{S}$.

Exemple 16 (trivial)**.**

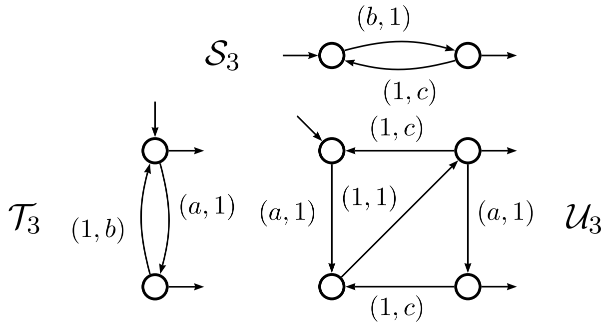


Exemple 17 (moins trivial)**.**



Ces deux exemples montrent des transducteurs « lettre-à-lettre ». On traitera des exemples plus généraux en exercice. On peut donner quand même un exemple simple qui fait apparaître une transition spontanée comme dans la preuve du Théorème 15.

Exemple 18.



Remarque 19. Les transducteurs *normalisés* $\mathcal{T}_3$ et $\mathcal{S}_3$ sont équivalents respectivement à $\mathcal{T}_1$ et $\mathcal{S}_1$. On observe que la construction du produit de composition donne dans ce cas un transducteur dans lequel la « multiplicité » des chemins n'est pas conservée. Une construction plus sophistiquée que nous ne détaillerons pas permet de pallier ce défaut.

1.3.2 Quelques conséquences

On déduit directement du Théorème 15 deux résultats essentiels : un théorème d'évaluation et un théorème de restriction.

Théorème 20.
L'image d'un langage rationnel par une relation RTF *est un langage rationnel.*

C'est-à-dire : $\theta\colon A^* \to B^*$ rel. RTF $\quad K \in \operatorname{Rat} A^* \implies \theta(K) \in \operatorname{Rat} B^*$.

Démonstration. Rappelons (*cf.* Exemple 10 (iii)) que l'on note $\iota_K\colon A^* \to A^*$ la relation « intersection avec K », *i.e.* $\widehat{\iota_K} = \{(w,w) \mid w \in K\}$. Il vient

$$\begin{aligned}\theta(K) &= \bigcup_{u\in K} \theta(u) = \bigcup_{v\in K} \left\{ w \in B^* \,\middle|\, (v,w) \in \widehat{\theta} \right\} \\ &= \left\{ w \in B^* \,\middle|\, \exists v \in K \quad (v,w) \in \widehat{\theta} \right\} \\ &= \left\{ w \in B^* \,\middle|\, \exists u \in A^*, \exists v \in A^* \quad (u,v) \in \widehat{\iota_K} \text{ et } (v,w) \in \widehat{\theta} \right\} \\ &= \mathsf{Im}\,(\theta \circ \iota_K) \ . \end{aligned}$$

■

Nous verrons à la prochaine leçon que l'intersection des graphes de deux relations RTF n'est pas nécessairement le graphe d'une relation RTF. Ce qui donne par contraste son intérêt à l'énoncé suivant.

Théorème 21. *Soient $\theta\colon A^* \to B^*$ une relation* RTF, *K un rationnel de A^* et L un rationnel de B^*. Alors $\widehat{\theta} \cap (K \times L)$ est le graphe d'une relation* RTF.

Démonstration. On vérifie aisément

$$\widehat{\theta} \cap (K \times L) = \widehat{\iota_L \circ \theta \circ \iota_K} \ .$$

■

1.4 Exercices

1..— **Ordres.** Dans cet exercice, l'alphabet A est *totalement ordonné* et cet ordre est noté $\leqslant$.
L'*ordre lexicographique*, noté $\preccurlyeq$, étend cet ordre de A à A^* de la manière suivante. Soient v et w deux mots de A^* et u leur plus long préfixe commun. Alors, $v \preccurlyeq w$ si $v = u$ ou, si $v = u\,a\,s$, et $w = u\,b\,t$ avec a et b dans A, alors $a < b$.

(a) Donner un transducteur fini sur $A^* \times A^*$ qui réalise $\preccurlyeq$, c'est-à-dire qui à chaque mot u de A^* fait correspondre l'ensemble des mots qui lui sont supérieurs (ou strictement supérieurs).

L'ordre *radiciel* (appelé aussi *généalogique*, *militaire*, ou *short-lex*), noté $\sqsubseteq$, est défini de la manière suivante : $v \sqsubseteq w$ si $|v| < |w|$ ou $|v| = |w|$ et $v \preccurlyeq w$.

(b) Donner un transducteur fini sur $A^* \times A^*$ qui réalise $\sqsubseteq$.

Pour tout langage L de A^*, on note $\mathsf{minlg}\,(L)$ (resp. $\mathsf{Maxlg}\,(L)$) l'ensemble des mots de L qui n'ont pas dans L de mots de même longueur plus petits (resp. plus grands) dans l'ordre lexicographique.

(c) Montrer que si L est rationnel, $\mathsf{minlg}\,(L)$ et $\mathsf{Maxlg}\,(L)$ le sont aussi.

2..— **Compléments.**

(a) Montrer que le complément de l'identité est une relation RTF.

(b) Montrer que le complément d'une relation RTF *fonctionnelle* est RTF.

3..— **Représentation des nombres.**

Soient $A_2 = \{0, 1\}$ et $A_3 = \{0, 1, 2\}$ deux alphabets de chiffres. On peut considérer A_3 comme un alphabet non-canonique pour la représentation des entiers en base 2 : $\overline{1\,2} = 4$, $\overline{2\,0\,1} = 9$, *etc.*

Soit $\nu_2 \colon A_3^* \to A_2^*$ la normalisation en base 2, c'est-à-dire la relation qui à un mot de A_3^* associe un mot de A_2^* qui représente le même entier en base 2.

(a) Donner un transducteur qui réalise ν_2.

(b) Soit $\varphi \colon A_2^* \to A_3^*$ la fonction qui envoie la représentation en binaire de chaque entier sur sa représentation en base 3, *e.g.* $\varphi(1\,0\,0\,0) = 2\,2$.

Montrer que φ n'est pas une relation RTF.

4..— **Opérations sur les nombres.**

(a) Donner un transducteur qui réalise la multiplication par 9 sur les entiers écrits en binaire, c'est-à-dire la relation $\tau \colon A_2^* \to A_2^*$ telle que $\overline{\tau(w)} = 9 \cdot \overline{w}$.

(b) Soit $\mu \colon A_2^* \times A_2^* \to A_2^*$ la relation qui réalise la multiplication, c'est-à-dire que $\mu(u, v) = w$ où $\overline{w} = \overline{u} \cdot \overline{v}$.

Montrer que μ n'est pas une relation RTF.

5..— **Equivalence d'application d'un morphisme.**

Soit $\varphi_1 \colon \{a, b, c\}^* \to \{x, y\}^*$ le morphisme défini par :

$$\varphi_1(a) = x\,, \quad \varphi_1(b) = y\,x\,, \quad \varphi_1(c) = x\,y\,.$$

(a) Donner un transducteur sous-normalisé qui réalise φ_1.

(b) Donner un transducteur sous-normalisé qui réalise ${\varphi_1}^{-1}$.

(c) Calculer un transducteur sous-normalisé qui réalise ${\varphi_1}^{-1} \circ \varphi_1$.