Automates sur les mots transfinis

Guillaume Claret

19/12/2008

Résumé

L'objectif de ce travail de rédaction est de s'intéresser à la notion de mot transfini et plus précisément à la généralisation de la notion d'automate et d'expression rationnelle à de tels mots : l'idée générale est alors d'indexer les lettres non plus par des entiers mais par des ordinaux quelconques. Cette généralisation peut sembler de prime abord artificielle mais son introduction a été historiquement motivée par de nombreuses applications pratiques, comme la décidabilité en logique (Büchi, 1960) ou la modélisation de machines séquentielles (Muller, 1963). Nous introduirons ici les définitions et outils de base nécessaires à leur étude, et enfin nous démontrerons un analogue du théorème de Kleene, en nous basant sur la démarche utilisée dans [3].

1 Définitions et notations

1.1 Ordinaux

On supposera connu dans la suite les notions de base sur les ordinaux, comme la notion d'ordinal limite, l'arithmétique des ordinaux, ...

1.2 Mot transfini

Définition 1. Soit Σ un alphabet fini ; un *mot transfini* est une application $u : \alpha \longrightarrow \Sigma$ pour un $\alpha \in Ord$. La *longueur* de u, notée $|u|$, est alors α.

Définition 2. Pour $\beta \leq \gamma \leq \alpha$, le *sous-mot* $u_{\beta\gamma}$ est la fonction $\delta \longmapsto (\beta + \delta)$ définie sur $\gamma - \delta$.

Définition 3. La *concaténation* de u et v mots transfinis est le mot :

$$\begin{array}{rccl} uv : & |u| + |v| & \longrightarrow & \Sigma \\ & \alpha & \longmapsto & \begin{cases} u(\alpha) & \text{si } \alpha < |u| \\ v(\alpha - |v|) & \text{sinon} \end{cases} \end{array}$$

Définition 4. Étant donné, dans le cas le plus général, des classes A et B d'ordinaux (et non simplement des ensembles) on pose naturellement :

$$AB = \{uv \mid u \in A, v \in B\}$$

et pour l'*exponentiation* :

$$A^\alpha = \{u \mid \exists\ (\beta_\gamma)_{\gamma<\alpha} \text{ suite propre dans } |u|+1 \text{ avec } \beta_0 = 0, \beta_\alpha = |u|, \forall \gamma < \alpha, v_{\beta_\gamma \beta_{\gamma+1}} \in A\}$$

une suite propre d'ordinaux étant une suite croissante telle que pour γ ordinal limite, $\beta_\gamma = \sup\limits_{\gamma'<\gamma} \beta_{\gamma'}$.

Définition 5. On note enfin $A^* = \bigcup\limits_{n\in\omega} A^n$ et $A^\# = \bigcup\limits_{\alpha\in Ord} A^\alpha$.

Exemple 6. Pour $u = b^3 a^\omega (ac)^{\omega^2} b$, $|u| = \omega^2 + 1$, $u_{\omega,\omega^2} = (ac)^\omega$ et :

$$L = \Sigma^*(\Sigma - \{a\})^\#$$

est l'ensemble des mots ayant un nombre fini de a.

1.3 Expression rationnelle transfinie

Définition 7. On définit inductivement une *expression rationnelle transfinie* par :
- $\varnothing$, ϵ, $a \in \Sigma$
- $e_1 + e_2$, $e_1 \cdot e_2$, e^*, e^ω, $e^\#$ avec e_1, e_2 et e expressions rationnelles

Le langage reconnu associé est alors notée $L(e)$ et suit la définition usuelle.

Exemple 8. Pour le langage L précedent avec $\Sigma = \{a, b\}$, on a $L = L((a+b)^* b^\#)$.

1.4 Automate transfini

Définition 9. On note, pour un ensemble E, $\mathcal{P}_s(E) = \mathcal{P}(E) \cup E$. Alors un *automate transfini* A est un quintuplet $(\Sigma, Q, I, F, \Delta)$ avec :

- Σ l'alphabet, Q l'ensemble des états
- $I \in \mathcal{P}_s(Q)$, $F \subset \mathcal{P}_s(Q)$
- $\Delta \subset \mathcal{P}_s(Q) \times \Sigma \times Q$

L'objectif va alors être d'étendre la notion de calcul à des mots de longueur infinie ; pour cela, l'idée va être de faire la distinction au niveau des longueurs des sous-mots entre les ordinaux successeurs et limites : pour les premiers, la définition va être la même (si on peut calculer u en arrivant sur $q \in Q$, alors pour $(q, a, q') \in \Delta$ on peut calculer ua en arrivant sur q'). Cependant, pour les ordinaux limites, on va devoir faire appel au concept de cofinalité, appliqué à tous les états précédemment atteints, et utiliser de nouvelles transitions qui partent d'un ensemble d'états. Plus formellement :

Définition 10. Un *calcul* sur $u \in \Sigma^{\#}$ est une fonction $H : |u| + 1 \longrightarrow \mathcal{P}_s(Q)$ continue, ie telle que, si α successeur, $H(\alpha) \in Q$, si α limite :

$$H(\alpha) = \{q \in Q \mid \{\beta < \alpha \mid H(\beta) = q\} \text{ cofinal à } \beta\}$$

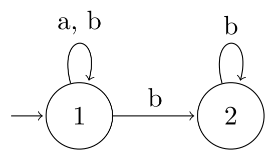
et vérifiant :

$$\forall \alpha < |u|, (H(\alpha), u(\alpha), H(\alpha + 1)) \in \Delta$$

Un mot est alors dit reconnu par A si il existe un *calcul acceptant* H avec $H(0) = I$ et $H(|u|) \in F$.

On notera $L(A)$ le langage des mots reconnaissables.

Exemple 11. En reprenant toujours le langage L précédent, un automate A associé peut par exemple être :



avec la transition supplémentaire $\{2\} \xrightarrow{b} 2$ et :

$$I = 1$$

$$F = \{1, 2, \{2\}\}$$

On a alors bien $L = L(A)$:

- Soit $u \in L$; alors $\exists v \in \Sigma^*, w \in b^{\#}, u = vw$. Soit le calcul suivant :

$$\begin{array}{rccl} H: & |u| + 1 & \longrightarrow & Q \\ & \alpha & \longmapsto & \begin{cases} 1 & \text{si } \alpha \leq |v| \\ 2 & \text{sinon} \end{cases} \end{array}$$

  On vérifie que H est bien un calcul valide sur u ; c'est de plus un calcul acceptant car $H(0) = I$ et $H(|u|) \in F$ selon que $w = \epsilon$, $|w|$ successeur ou $|w|$ limite.
- Réciproquement, tout calcul H acceptant un mot u est croissant sur les états (sans compter les images d'ordinaux limites), et l'état 1 ne peut être atteint une infinité de fois, car sinon $H(\omega) = \{1\}$ qui n'est état de départ d'aucune transition et non final. En notant n le dernier entier tel que $H(n) = 1$ (valant $|u|$ si H constante à 1), on en déduit $u_{0,n} \in \Sigma^*$ et $u_{n,|u|} \in b^{\#}$ car on peut lire une lettre quelconque en restant dans 1, puis seuls des b peuvent être lus en allant dans 2.

2 Équivalence avec les expressions rationnelles

Comme dans le cas fini, ces deux types de représentation d'un langage sont en fait équivalentes, et il existe de plus un moyen constructif de passer d'une forme à l'autre, ce qui permet d'obtenir par là même certain résultats de décidabilité, comme décider si un langage transfini est vide.

2.1 Des expressions vers les automates

Théorème 12. *Pour toute expression rationnelle e transfinie on peut construire un automate A tel que :*

$$L(e) = L(A)$$

Nous ne montrerons pas cette partie, qui est traitée par exemple dans [2].

2.2 Des automates vers les expressions rationnelles

Théorème 13. *Réciproquement, étant donné un automate A, on peut construire une expression rationnelle e vérifiant :*

$$L(A) = L(e)$$

C'est la partie la plus intéressante. Nous allons raisonner par récurrence en appliquant une variante de la démonstration standard du théorème de Kleene, où l'on associe une expression rationnelle aux langages des mots permettant d'accéder à j à partir de i en ne passant que par des états d'indice inférieur à k. Pour cela, étant donné $A = (\Sigma, Q, I, F, \Delta)$ avec $Q = \{q_1, \cdots q_n\}$, on définit, pour $Z_1, Z_2 \in \mathcal{P}_s(Q)$ et $Z_2 \in \mathcal{P}(Q)$, $C(Z_1, Z_2, Z_3)$ le langage des mots qui possèdent un calcul allant de l'état Z_1 à Z_3 en ne passant que par des états de $\mathcal{P}_s(Z_2)$, et $CC(Z_1, Z_2, Z_3)$ de même mais en s'interdisant de passer par Z_2. Nous noterons $E(Z_1, Z_2, Z_3)$ et $EE(Z_1, Z_2, Z_3)$ les expressions rationnelles associées.
Par récurrence sur k le cardinal de Z_2 :

- Si k = 0 :

  $$E(Z_1, \varnothing, Z_3) = EE(Z_1, \varnothing, Z_3) = \begin{cases} \{a \in \Sigma \mid (Z_1, a, Z_3) \in \Delta\} \cup \{\epsilon\} & \text{si } Z_1 = Z_3 \\ \{a \in \Sigma \mid (Z_1, a, Z_3) \in \Delta\} & \text{sinon} \end{cases}$$

  convient.
- Pour $k \geq 1$ ($Z_2 = \{q_1, \ldots, q_k\}$) : on va traiter trois cas différents :
  - $EE(Z_1, Z_2, Z_3)$ avec $Z_2 \neq Z_3$:
    Soient :

    $$E_{boucle} = E(q_1, Z_2 - \{q_2\}, q_2) \cdots E(q_{k-1}, Z_2 - \{q_k\}, q_k) \cdot E(q_k, Z_2 - \{q_1\}, q_1)$$

    $$E_i = (\prod_{j=1}^{i-1} E(q_j, Z_2 - \{j+1\}, q_{j+1})) \cdot E(q_i, Z_2 - \{q_{i+1 \text{ ou } 1 \text{ si } i=k}\}, Z_3)$$

Alors on va avoir :

$$EE(Z_1, Z_2, Z_3) = E(Z_1, Z_2 - \{q_1\}, Z_3) + E(Z_1, Z_2 - \{q_1\}, q_1) \cdot E^*_{boucle} \cdot \bigcup_{i=1}^{k} E_i$$

(qui est bien exprimé en fonction d'expressions construites par récurrence) L'idée est que lorsque l'on considère un chemin acceptant pour un mot u, si on ne passe pas par q_1 on se retrouve dans le premier cas de la somme, sinon on considère la première fois que l'on atteint q_1 (expression $E(Z_1, Z_2 - \{q_1\}, q_1)$). Puis chaque boucle va représenter une suite d'étapes $q_1 \to q_2 \to \cdots \to q_n \to q_1$ en considérant à chaque fois la première fois que l'état q_i est atteint. Cette suite d'étapes ne peut se répéter qu'un nombre fini de fois car on ne sera jamais dans l'état Z_2 : en effet, sinon on aurait une partie du calcul acceptant qui correspondrait à $(E_{boucle})^\omega$, dont l'état final serait nécessairement Z_2. Or on n'a pas le droit de passer par Z_2 lors d'une étape intermédiaire et $Z_3 \neq Z_2$. Enfin on réalise une boucle partielle qui se termine sur Z_3 (dernière union).

– $EE(Z_1, Z_2, Z_2)$:
  Dans ce cas on pose :

$$EE(Z_1, Z_2, Z_3) = E(Z_1, Z_2 - \{q_1\}, q_1) \cdot (E_{boucle})^\omega$$

  Ceci va convenir car il est nécessaire que les $\{\beta < |u| \mid H(\beta) = q_i\}$ soient confinaux dans $|u|$, et donc qu'il y ait une infinité de boucles. De plus il doit y en avoir exactement ω car sinon, si il y en avait α avec $\alpha > \omega$, en notant H le calcul acceptant dans A, H restreint à un certain ordinal β tel que H deviennent un calcul d'un mot de $E(Z_1, Z_2 - \{q_1\}, q_1) \cdot (E_{boucle})^\omega$ vérifierait $H(\beta) = Z_2$ par confinalité des $\{\beta < |u| \mid H(\beta) = q_i\}$ dans β et donc impliquerait $\beta = |u|$ car $Z_2 = Z_3$. Ainsi l'expression rationnelle proposée convient.

– $E(Z_1, Z_2, Z_3)$:
  Posons maintenant :

$$E(Z_1, Z_2, Z_3) = EE(Z_1, Z_2, Z_3) +$$
$$EE(Z_1, Z_2, Z_2) \cdot (EE(Z_2, Z_2, Z_2))^\# \cdot EE(Z_2, Z_2, Z_3)$$

  On applique ici exactement la démarche utilisée pour montrer le théorème de Kleene ; le danger dans les cas précédents était l'introduction d'états non désirés avec les ordinaux limites, mais ici il n'y a pas de problème car Z_2 est stable par union.
  Enfin on conclut par $L(A) = L(E)$ avec $E = \bigcup_{Z \in F} E(I, Q, Z)$.

On remarque cependant que cette construction est très peu efficace algorithmiquement, bien moins que dans le cas fini, notamment à cause de ensembles d'états. En effet, on a ici dans le pire cas à calculer de l'ordre de $2^n n 2^n$ expressions rationnelles de la forme $E(Z_1, Z_2, Z_3)$, là où il n'en fallait que n^3 avec Kleene.

Références

[1] N. Bedon. Langages reconnaissables de mots indexés par des ordinaux. 1998.

[2] J. Wojciechowski. Classes of transfinite sequences accepted by finite automata. 1984.

[3] J. Wojciechowski. Finite automata on tranfinite sequences and regular expressions. 1984.