Un automate pour reconnaître les mots finis et infinis qui ont au moins deux factorisations sur un langage fini

Xavier Augros

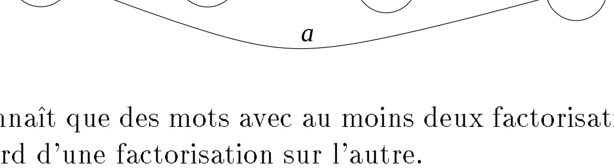
Laboratoire I3S bât. ESSI
930, Route des Colles, B.P. 145
06903 Sophia Antipolis Cedex
e-mail: augros@i3s.unice.fr

L'objectif est de reconnaître les mots (finis ou infinis) qui ont au moins deux factorisations sur un ensemble fini L. On s'intéresse plus particulièrement aux mots qui sont couvert de manière ambiguë [Kar85] par ce langage fini, c'est à dire les mots qui ont au moins deux factorisations de premier facteur diffèrent. Pour reconnaître ces mots, nous utilisons un graphe, appelé graphe des retards, qui est inspiré par J.Karhumäki dans [Kar85]. Il utilise ce graphe pour représenter les deux factorisations d'un mot infini dans le cas des codes à trois éléments. Étant donné un langage fini, nous définissons le graphe des retards comme un graphe orienté où l'ensembles des sommets est l'ensemble des retards entre deux factorisations, *i.e* les étiquettes des sommets sont les avances d'une factorisation sur l'autre pour les préfixes d'un mot ambigu donné. Intuitivement, l'étiquette d'une arrête correspond au plus petit préfixe d'une des deux factorisations à partir du préfixe déjà lu. Le retard ε signifie soit que l'on est au début d'un mot, soit que l'ont vient de lire un mot fini qui a deux factorisations.
Nous avons alors les résultats suivant: Le langage L est un code si et seulement si il n'y a pas de cycle dans le graphe qui passe par l'état initial, et L est un ω-code[Sta86] si et seulement si il n'y a pas de cycle.
Ce graphe est un automate non-deterministe qui reconnaît l'ensemble des mots finis et infinis qui sont couvert de manière ambiguë par L. L'ensemble des états est l'ensemble des retard (*i.e.* l'ensemble des sommets du graphe). L'état initial est le sommet étiqueté par ε. L'ensemble des états finaux est $\{\varepsilon\}$ pour la reconnaissance des mots finis. Nous utilisons la définition des automates de Büchi (voir par exemple [PP]) pour la reconnaissance des mots infinis, dans ce cas l'ensemble des états finaux est l'ensemble de tous les états. Les transitions de l'automates sont les arrêtes du graphe.

Prenons un exemple:
soit le langage $L = \{a, abac, ba, ca\}$ et considérons les langages L^* and L^ω.

- L'ensemble des mots finis couvert de manière ambiguë sur L est $a(baca)^+$
- Celui des infinis est $(a(baca)^+)^*((abac)^\omega \cup (abaca)^\omega$

Le graphe suivant représente tous ces mots:



Cet automate ne reconnaît que des mots avec au moins deux factorisations et l'étiquette d'un état représente le retard d'une factorisation sur l'autre.

Quand nous lisons les factorisations d'un tel mot, nous commençons avec un retard nul. Si dans une factorisation nous lisons un a le retard devient a:

a . . .

Alors, si nous lisons le mot ba, la factorisation du bas a une avance de aba:

a b a . . .

Maintenant, en lisant c, la factorisation du haut commence avec le mot $abac$ et le retard de celle du bas est c:

a b a c . . .

La factorisation du bas continue avec le mot ca et le retard de celle d'en haut est a:

a b a c a . . .

Si dans la factorisation du haut, nous lisons un a, nous avons un mot de L^+ qui a deux factorisations de premier pas diffèrent. Cependant, on peut continuer le mot en lisant ba dans la factorisation du bas.
Nous pouvons alors lire tous les mots finis qui ont deux factorisations. Ces mots de L^+ sont représentés par un chemin fini, dans le graphe, qui commence et se termine par le sommet ε. Un mot infini, qui a deux factorisations, est lui représenté par un chemin infini dans le graphe qui commence par le sommet etiqueté par ε et passe une infinité de fois par n'importe quel sommet du graphe.

Références

[Kar85] J. Karhumäki. A property of three-element codes. *TCS*, 41:215–222, 1985.

[PP] D. Perrin and J.E. Pin. Mots infinis. (to be published) http://www.liafa.jussieu.fr/~jep/Resumes/MotsInfinis.html.

[Sta86] L. Staiger. On infinitary finite length codes. *Theoretical Informatics and Applications*, 24(4):483–494, 1986.