

TD 5 : Grammaires algébriques

1 Applications des lemmes d'itération

Exercice 1 (Lemme d'OGDEN).

1. Montrer que le langage $L_{\text{cross}} \stackrel{\text{def}}{=} \{a^n b^m c^n d^m \mid n, m \geq 0\}$ n'est pas algébrique.
2. Montrer que le langage $L_{\text{copy}} \stackrel{\text{def}}{=} \{ww \mid w \in \{a, b\}^*\}$ n'est pas algébrique.
3. Montrer que le langage $\{a^n b^n c^m \mid n, m \geq 0\} \cup \{a^n b^m c^m \mid n, m \geq 0\}$ est linéaire et inhéremment ambigu. On pourra appliquer le lemme aux mots $a^{n+n!} b^n c^n$ puis $a^n b^n c^{n+n!}$ et exhiber deux arbres de dérivations différents pour le mot $a^{n+n!} b^{n+n!} c^{n+n!}$.

Exercice 2 (Langage de programmation [Floyd, *Comm. ACM* 5(9), 1962]). La première utilisation des grammaires algébriques pour décrire la syntaxe d'un langage de programmation a été faite pour le langage ALGOL 60. Néanmoins, la modélisation par une grammaire algébrique seule n'est pas possible : le programme suivant

```
begin
  real x;
  y := 10
end
```

n'est correct que si les identifiants `x` et `y` coïncident. Cependant un identifiant ALGOL 60 est une chaîne arbitrairement longue du langage rationnel $\ell(\ell+c)^*$ où ℓ désigne n'importe quelle lettre majuscule ou minuscule de l'alphabet latin, et c n'importe quel chiffre de 0 à 9.

Montrer que ce niveau de correction ne peut pas être assuré par une grammaire algébrique seule.

Exercice 3 (Langage naturel [Shieber, *Lingu. Phil.* 8(3), 1985]). Un exemple de phénomène syntaxique non exprimable par une grammaire algébrique est issu du Suisse-Allemand :

Jan säit das mer d'chind em Hans es huus haend wele laa hülfe aastrüiche
Jan dit que nous les enfants-ACC Hans-DAT la maison-ACC avons voulu laisser aider peindre
 « *Jan dit que nous avons voulu laisser les enfants aider Hans à peindre la maison* »

Dans cette phrase, on peut distinguer

- des syntagmes nominaux à l'accusatif, comme *d'chind*,
- des syntagmes nominaux au datif, comme *em Hans*,
- des verbes recevant exactement un argument accusatif, comme *laa*, et
- des verbes recevant exactement un argument datif, comme *hülfe*.

De plus, cet exemple est productif : on peut continuer à ajouter des subordinées comme *em Hans... hülfe* et *d'chind... laa* de manière arbitraire.

Montrer qu'il n'existe pas de grammaire algébrique pour le Suisse-Allemand.

2 Problème du mot

Exercice 4 (Intersection avec un langage reconnaissable). Soit G une grammaire algébrique et A un automate fini. Construire G' algébrique telle que $L(G') = L(G) \cap L(A)$.

Exercice 5 (Algorithme de COCKE, KASAMI et YOUNGER).

1. Quelle est la complexité de la construction d'une grammaire algébrique pour le langage $L(G) \cap \{w\}$?
2. Modifier la construction précédente pour calculer *au vol* si $L(G') = \emptyset$.