Langages Formels

Rappel : la famille des langages algébriques est close par concaténation, itération, union, intersection avec un rationnel, morphisme, et morphisme inverse (cf. exercice 6). On pourra utiliser ces propriétés dans les exercices qui suivent.

Exercice 1 (Lemme d'Ogden)
Le but de cet exercice est de prouver le lemme d'Ogden, qui est l'analogue pour les langages algébriques du lemme de l'étoile.

Pour toute grammaire algébrique $G = \langle N, \Sigma, P, S \rangle$ et $A \in N$, on note $\widehat{L}_G(A) \stackrel{\text{def}}{=} \{\alpha \in (N \cup \Sigma)^* \mid A \Rightarrow_G^* \alpha\}$.

Lemme 1 (Lemme d'Ogden)**.** *Soit $G = \langle N, \Sigma, P, S \rangle$ une grammaire algébrique. Il existe un entier K tel que pour tout $A \in N$ et $w \in \widehat{L}_G(A)$ ayant au moins K lettres distinguées, il existe $B \in N$ et $\alpha, u, \beta, v, \gamma \in (N \cup \Sigma)^*$ tels que :*

(a) $w = \alpha u \beta v \gamma$;

(b) $A \Rightarrow^* \alpha B \gamma$, $B \Rightarrow^* uBv$, et $B \Rightarrow^* \beta$;

(c) (α et u et β) ou (β et v et γ) contiennent des lettres distinguées ;

(d) $u\beta v$ contient moins de K lettres distinguées.

En particulier, pour tout $n \geq 0$, $\alpha u^n \beta v^n \gamma \in \widehat{L}_G(A)$.

1. Soit un arbre t où certaines feuilles sont distinguées. On dit qu'un noeud est *distingué* lorsque le sous-arbre dont il est racine contient des feuilles distinguées, et *spécial* lorsqu'il a au moins deux fils distingués.

   Montrer que si t est de degré au plus m, a k feuilles distinguées, et que chaque branche contient au plus r noeuds spéciaux, alors $k \leq m^r$.
2. Montrer le lemme d'Ogden.

Exercice 2 (Intersection de langages algébriques)

1. Montrer que le langage $\{a^n b^n c^n \mid n \geq 0\}$ n'est pas algébrique.
2. Montrer que la classe des langages algébriques n'est close ni par intersection, ni par complémentation.

Exercice 3 (Applications du 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 algébrique et inhéremment ambigu. On pourra appliquer le lemme aux mots $a^{K+K!}b^K c^K$ puis $a^K b^K c^{K+K!}$ et exhiber deux arbres de dérivations différents pour le mot $a^{K+K!}b^{K+K!}c^{K+K!}$.

Exercice 4 (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 5 (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*	*hälfe*	*aastriiche*
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 *hälfe*.

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

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

Exercice 6 (Clotûre par morphisme inverse)
Étant donnés un alphabet fini Σ et $\Delta \subseteq \Sigma$, on appelle *projection de* Σ *sur* Δ le morphisme $\pi \colon \Sigma^* \to \Delta^*$ défini par $\pi(a) = a$ si $a \in \Delta$ et $\pi(a) = \varepsilon$ sinon.

1. Montrer que la famille des langages algébriques est fermée par projection inverse.
2. Montrer que la famille des langages algébriques est fermée par morphisme inverse.