

THÉORÈME DE CHOMSKY-SCHÜTZENBERGER

Si $G = (\Sigma, V, P, S)$ est une grammaire et $X \in V$, on note $\mathcal{L}_G(X)$ le langage engendré par la grammaire (Σ, V, P, X) . On a donc en particulier $\mathcal{L}_G(S) = \mathcal{L}(G)$.

I Langage de Dyck

On définit la grammaire $G_n = (\Sigma_n, V, P_n, S)$ par $\Sigma_n = \{a_1, \dots, a_n, \bar{a}_1, \dots, \bar{a}_n\}$ et P_n constituée des règles suivantes :

$$\begin{aligned} S &\rightarrow ST \mid \varepsilon \\ T &\rightarrow a_1 S \bar{a}_1 \mid \dots \mid a_n S \bar{a}_n \end{aligned}$$

On appelle *langage de Dyck primitif sur n parenthèses* le langage $D_n = \mathcal{L}_{G_n}(T)$ et *langage de Dyck sur n parenthèses* le langage $\mathcal{L}_{G_n}(S)$.

Remarque

Un alphabet de la forme Σ_n , où les symboles viennent par paire a, \bar{a} , est dit *apparié*.

- **Question 1** Montrer que $\mathcal{L}_{G_n}(S) = D_n^*$.
- **Question 2** Donner un arbre de dérivation pour $a_1 a_2 \bar{a}_2 a_3 \bar{a}_3 \bar{a}_1 a_3 \bar{a}_3$ pour G_3 .
- **Question 3** Cette grammaire est-elle ambiguë? Justifier sommairement.
- **Question 4** On définit le type OCaml suivant pour représenter les éléments de Σ_n (où **0** i code a_i et **F** i code \bar{a}_i) :

```
type parenthese = 0 of int | F of int
```

Écrire une fonction permettant de décider si un mot $w \in \Sigma_n^*$ (donné sous forme de liste) appartient au langage D_n^* .

```
val est_parenthese : parenthese list -> bool
```

II Morphismes

Définition 1.1

Soient Σ_1, Σ_2 deux alphabets. Un *morphisme* de Σ_1^* vers Σ_2^* est une application $\varphi : \Sigma_1^* \rightarrow \Sigma_2^*$ telle que, pour tous mots $u, v \in \Sigma_1^*$, on ait :

$$\varphi(uv) = \varphi(u)\varphi(v)$$

Un morphisme est dit *alphabétique* si $|\varphi(a)| \leq 1$ pour toute lettre $a \in \Sigma$.

- **Question 5** Exprimer le nombre de morphismes alphabétiques de Σ_1^* vers Σ_2^* en fonction des cardinaux n_1 et n_2 des alphabets.
- **Question 6** Soit $\varphi : \Sigma_1^* \rightarrow \Sigma_2^*$ un morphisme et $L \subseteq \Sigma_1^*$ un langage algébrique. Montrer que $\varphi(L)$ est algébrique.

► **Question 7** Soit $\varphi : \Sigma_1^* \rightarrow \Sigma_2^*$ un morphisme alphabétique et $L \subseteq \Sigma_2^*$ un langage algébrique. Montrer que $\varphi^{-1}(L)$ est algébrique.
 On pourra poser $A = \{a \in \Sigma_1 \mid \varphi(a) \in \Sigma_2\}$ et $B = \{a \in \Sigma_1 \mid \varphi(a) = \varepsilon\}$, puis construire à partir d'une grammaire G engendrant L une grammaire G' engendrant $\varphi^{-1}(L)$ en ajoutant à G une variable et en remplaçant chacune de ses règles par un ensemble de règles.

Remarque

C'est en fait vrai pour un morphisme quelconque (et vous devriez voir comment votre preuve peut s'adapter).

III Théorème de représentation de Chomsky-Schützenberger

Le but de cette partie est de prouver le théorème suivant :

Théorème I.2 – Chomsky-Schützenberger

Un langage L est algébrique si et seulement il existe un langage rationnel K , un langage de Dyck D_n et un morphisme alphabétique φ tels que $L = \varphi(D_n \cap K)$

► **Question 8** On suppose L algébrique et K rationnel sur un même alphabet Σ . En considérant $A = (\Sigma, Q, \delta, q_0, F)$ un automate fini déterministe reconnaissant K et $G = (\Sigma, V, P, S)$ une grammaire en forme normale de Chomsky engendrant L , montrer que $L \cap K$ est algébrique. On construira une grammaire G' ayant un symbole initial S' et une variable X_{pq} pour toute variable $X \in V$ et tous états $p, q \in Q$.

► **Question 9** En déduire un sens du théorème.

On considère à présent un langage algébrique L ne contenant pas le mot vide, engendré par une grammaire $G = (\Sigma, V, P, S)$ en forme normale de Chomsky. On définit la grammaire $G' = (\Sigma_n, V, P', S)$ (où Σ_n est un alphabet apparié) en fixant une numérotation $P = \{R_1, \dots, R_k\}$ des règles de G et en prenant les règles suivantes pour G' :

- pour chaque règle R_i de la forme $X \rightarrow YZ$ de P , P' contient la règle $X \rightarrow a_i b_i Y \overline{b_i} c_i Z \overline{c_i} a_i$;
- pour chaque règle R_i de la forme $X \rightarrow a$ de P , P' contient la règle $X \rightarrow a_i \overline{a_i}$.

On note $L' = \mathcal{L}(G')$.

► **Question 10** Exprimer n en fonction du nombre de règles (de chaque forme) de G .

► **Question 11** Construire la grammaire G' correspondant à la grammaire G suivante :

$$\begin{array}{lll} S \rightarrow XY & X \rightarrow YX & Y \rightarrow ZZ \\ Z \rightarrow AB & A \rightarrow a & B \rightarrow b \end{array}$$

► **Question 12** Montrer que $L' \subseteq D_n^*$.

On définit un morphisme alphabétique $\varphi : \Sigma_n^* \rightarrow \Sigma^*$ en posant :

- $\varphi(a_i) = a$ et $\varphi(\overline{a_i}) = \varepsilon$ si la règle R_i est de la forme $X \rightarrow a$;
- $\varphi(a_i) = \varphi(\overline{a_i}) = \varphi(b_i) = \varphi(\overline{b_i}) = \varphi(c_i) = \varphi(\overline{c_i}) = \varepsilon$ si la règle R_i est de la forme $X \rightarrow YZ$.

► **Question 13** Montrer que $L = \varphi(L')$.

► **Question 14** On définit P comme l'ensemble des premières lettres des mots de L' . Exprimer P en fonction des règles de G .

► **Question 15** On définit F comme l'ensemble des facteurs de longueur 2 des mots de L' . Exprimer F en fonction des règles de G .

► **Question 16** On pose $N = \Sigma_n^2 \setminus F$ et $K = (P\Sigma_n^* \setminus (\Sigma_n^* N \Sigma_n^*))$. Montrer que $L' = D_n^* \cap K$.

► **Question 17** Conclure.