MPI: Langages Hors-contexte

Cours

- Grammaire, grammaire hors-contexte, dérivation.
- Grammaires générales, contextuelles (HP)
- Arbre d'analyse, dérivation à gauche, à droite.
- Ambiguité d'une grammaire. Ambiguité du sinon pendant.
- Forme normale de Chompsky (HP)

Question de Cours

- Montrer que les langages réguliers sont hors contexte par induction.
- Montrer que les langages réguliers sont hors contexte par construction de la grammaire à partir de l'automate.
- Montrer que l'intersection d'un langage régulier et d'un langages hors-contexte est hors-contexte.

To Do

- Sujet sur les images de Parikh

Langage hors-contexte

Soit L, L' deux langages hors-contexte. Montrer que les langages suivants sont hors-contexte:

- $\left\{w_1 w_1 w_2 w_2 ... w_{|w|} w_{|w|} : w \in L\right\}$
- $\overline{L} = \{\overline{u} : u \in L\}$ le langages des mirroirs, avec $\overline{u_1 ... u_n} = u_n ... u_1$
- $\bigcup_{w \in L} \text{Pref}(w)$ pour $\text{Pref}(w)$ l'ensemble des préfixes de w
- $\bigcup_{w \in L} S(w)$ pour $S(w)$ l'ensemble des sous-mots de w
- $\{w_1 w_3 w_5 .... w_i : w \in L\}$, une lettre sur deux des mots de L

Paires différentes

On pose $\Sigma = \{a, b\}$. On considère tout d'abord les langages suivants

$$L_a = \{uav : u, v \in \Sigma^* \mid |u| = |v|\}$$
$$L_b = \{ubv : u, v \in \Sigma^* \mid |u| = |v|\}$$
$$L_1 = \{ww' \mid w \neq w' \wedge |w| = |w'|\}$$

1. Montrer que L_a et L_b sont hors-contexte.
2. Montrer que $L_1 = L_a L_b \cup L_b L_a$. En déduire que L_1 est hors-contexte.

On admet que $L' = \{uu : u \in \Sigma^*\}$ n'est pas hors contexte

3. Montrer que les langages hors-contexte ne sont pas stable par complémentaire.

Autant de lettres

Soit $\Sigma = \{a, b\}$, on pose $f : \Sigma^* \to \mathbb{Z}$ tel que $f(w) = |w|_a - |w|_b$. On considère les langages suivants:

$$L = \{w \in \Sigma^* \mid |w|_a = |w|_b\} = f^{-1}(\{0\})$$
$$L_+ \text{ engendré par } S \to SS \mid aSb \mid \varepsilon$$
$$L_- \text{ engendré par } S \to SS \mid bSa \mid \varepsilon$$

On cherche à trouver une grammaire non-ambigue pour L (les mots avec autant de a que de b)

1. Est-ce que la grammaire définissant L_+ est ambigue ? Donner une grammaire pour L (sans preuve)
2. Montrer que L_+ est l'ensemble des mots w tel que $f(w) = 0$ et que pour tout w' préfixe de w on a $f(w') \geq 0$
3. En remarquant qu'un mot de L_+ commence forcément par un a, donner une grammaire non-ambigue pour L_+.
4. Donner une grammaire non-ambigue pour L est hors-contexte

Forme normale de Chompsky

Soit G une grammaire, montrer qu'il existe une grammaire équivalente à G tel que toutes les règles sont soit de la forme $X \to a$ pour a un symbole terminal, soit $X \to AB$ pour A, B non terminaux, soit $S \to \varepsilon$ pour S le symbole initial.

Morphisme de langage hors-contexte

On dit que $\varphi : \Sigma_1^* \longrightarrow \Sigma_2^*$ est un morphisme si pour tout $u, v \in \Sigma_1^*$, $\varphi(uv) = \varphi(u)\varphi(v)$.
Montrer que si L est hors-contexte, alors $\varphi(L)$ l'est aussi.

Lemme d'itération pour les langages hors-contexte

On veut montrer que $L_3 = \{a^n b^n c^n : n \in \mathbb{N}\}$ n'est pas hors contexte. Pour cela on montre un lemme analogue au lemme de l'étoile mais pour les langages hors-contexte.

Soit $G = (\Sigma, \Gamma, R, S)$ une grammaire hors-contexte reconnaissant le langage L. On définit $\|R\|_\infty = \max\{|w| : (X \to w) \in R\}$

1. Montrer que si $w \in L$ est un mot tel que $|w| > |\Gamma| \times \|R\|_\infty$, alors il existe $A \in \Gamma$ et $u, v, x, y, z \in \Sigma^*$ avec $xz \neq \varepsilon$ tel que $S \Rightarrow^* uAv$ et $A \Rightarrow^* xAz$ et $A \Rightarrow^* y$.
2. En déduire le lemme *d'itération pour les langages hors-contexte*: pour tout L un langage hors contexte, il existe $N > 0$ tel que pour tout $w \in L$ tel que $|w| > N$ il existe $u, x, y, z, v \in \Sigma^*$ tel que
   - $uxyzv = w$
   - $uxyzv = w$
   - $\forall n \in \mathbb{N}, ux^n yz^n v \in L$
3. En déduire que $\{a^n b^n c^n : n \in \mathbb{N}\}$ n'est pas hors-contexte.
4. Montrer que l'ensemble des langages hors-contexte n'est pas stable par intersection.

Automates à pile

On cherche à donner un modèle de calcul équivalent aux grammaires hors-contexte: les *automates à pile*. Un automate à pile est un tuple $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ où :

- Σ est l'alphabet d'entrée
- Q est un ensemble fini d'*états* et $q_0 \in Q$ est l'état initial,
- Γ est l'alphabet de pile, et $Z_0 \in \Gamma$ est le symbole initial de pile,
- $F \subseteq Q$ est l'ensemble des états finaux,
- $\delta : Q \times (\Sigma \cup \{\varepsilon\}) \times \Gamma \longrightarrow \mathcal{P}(Q \times \Gamma^*)$ est une relation finie de transition.

Une *configuration* est un couple $(q, p) \in Q \times \Gamma^*$. Pour $\alpha \in \Sigma \cup \{\varepsilon\}$ et $\beta \in Z$, on dit qu'il y a *une transition* de la configuration $(q, \beta p)$ vers $(q', p'p)$ si $(q', p') \in \delta(q, \alpha, \beta)$. On note cela

$$(q, \beta p) \xrightarrow{\alpha} (q', p'p).$$

S'il existe une suite de transition $(q, p) \xrightarrow{a_1} \dots \xrightarrow{a_n} (q', p')$ avec $u = a_1 \dots a_n$, on note cela $(q, p) \xrightarrow{u} (q', p')$. Remarquer que l'on n'a pas forcément $|u| = n$. On dit qu'un mot u est reconnu par $\mathcal{A}$ si $(q_0, Z_0) \xrightarrow{u} (q_f, \varepsilon)$ avec $q_f \in F$. On note par $L(\mathcal{A})$ le langage des mots reconnu par $\mathcal{A}$ un automate à pile. Ces automates sont connu pour être "acceptant par pile vide" dans la littérature.

1. Montrer que $\{a^n b^n : n \in \mathbb{N}\}$ est un langage reconnu par un automate à pile.
2. Montrer que si un langage est régulier alors il est reconnu par un automate à pile.

On va montrer que tout langage hors-contexte est reconnu par un automate à pile. Soit $G = (\Sigma, \Gamma, R, S)$ une grammaire hors-contexte. On construit un automate à pile $\mathcal{A}_G = (Q, \Sigma, \Gamma', \delta, q, Z_0, F)$ de la manière suivante:

- On pose $\Gamma' = \Gamma \cup \Sigma$, $Q = \{q_0\}$, $F = \{q\}$ et $Z_0 = S$,
- Pour chaque règle $(A \to w) \in R$, on ajoute (q, w) à $\delta(q, \varepsilon, A)$,
- Pour chaque lettre terminale $a \in \Sigma$, on ajoute (q, ε) à $\delta(q, a, a)$.

3. Dessiner l'automate à pile associé à la grammaire $S \to aSbS \mid \varepsilon$.
4. Soient $u \in \Sigma^*$ et $p \in (\Gamma \cup \Sigma)^*$. Montrer que $(q, S) \xrightarrow{u} (q', p')$ si et seulement si $S \Rightarrow^* up$
5. Montrer que $L(\mathcal{A}_G) = L(G)$, et en déduire que tout langage hors-contexte est reconnu par un automate à pile.

On va maintenant montrer la réciproque par la méthode des triplets de Ginsburg. Soit $\mathcal{A} = (Q, \Sigma, \Gamma, \delta, q, Z_0, F)$ un automate à pile. Pour tout $q, q' \in Q$ et $\gamma \in \Gamma$, on pose

$$L_{[q,\gamma,q']} = \left\{ u \in \Sigma^* \mid (q, \gamma) \xrightarrow{u} (q', \varepsilon) \right\}$$

6. Montrer que si $(q, \gamma) \xrightarrow{u} (q', \varepsilon)$, alors pour tout $p \in \Gamma^*$ on a $(q, \gamma p) \xrightarrow{u} (q', p)$.
7. Montrer que, pour tout $q, q' \in Q$ et $\gamma \in \Gamma$,

$$L_{[q,\gamma,q']} = \left\{ \alpha \in \Sigma \cup \{\varepsilon\} : (q, \gamma) \xrightarrow{\alpha} (q', \varepsilon) \right\} \cup \bigcup_{\alpha \in \Sigma \cup \{\varepsilon\}} \bigcup_{(q,\gamma) \xrightarrow{\alpha} (q_1, \gamma_1 \dots \gamma_k)} \bigcup_{q_2, \dots, q_{k-1} \in Q} \alpha L_{[q_1,\gamma_1,q_2]} L_{[q_2,\gamma_2,q_3]} \dots L_{[q_{k-1},\gamma_k,q']}$$

8. En déduire qu'il existe une grammaire $G = (\Sigma, \Gamma', R, S)$ avec $\Gamma' = Q \times \Gamma \times Q$ tel que $L(G) = L(\mathcal{A})$

Mélange et grammaires

On fixe $\Sigma = \{a, b, c\}$. Pour deux mots $u, v \in \Sigma^*$, on dit que $w \in \Sigma^*$ est un entrelacement de u et v s'il existe une partition $\{i_1, ..., i_n\} \cup \{j_1, ..., j_m\}$ de $[\![1; |w|]\!]$ avec $i_1 < ... < i_n$ et $j_1 < ... < j_m$ tel que $u = w_{i_1}...w_{i_n}$ et $v = w_{j_1}...w_{j_m}$. On note $u ш v$ l'ensemble des entrelacements de u et v.

Pour L, L' deux langages sur Σ, on définit

$$L ш L' = \bigcup_{\substack{u \in L \\ v \in L'}} u ш v$$

1. Donner ab ш ac.
2. Proposer un algorithme qui prend en entrée $u, v, w \in \Sigma^*$ et qui teste si $w \in u ш v$.

On admet que $L_3 = \{a^n (bc)^n a^n : n \in \mathbb{N}\}$ n'est pas un langage hors-contexte.

3. Est-ce que si L, L' sont hors-contexte, alors $L ш L'$ est forcément hors-contexte?
4. Montrer que si L est hors-contexte et R régulier, alors $L ш R$ est hors-contexte.

Circular shift

Soit L un langage. On définit

$$\mathrm{Circ}(L) = \{uv \mid vu \in L\}.$$

Question 1 Montrer que si L est régulier, alors $\mathrm{Circ}(L)$ l'est aussi.

On cherche maintenant à généraliser le résultat pour les langages hors-contexte. Soit $G = (\Sigma, \Gamma, R, S)$ une grammaire hors-contexte telle que $L = L(G)$. Pour $A \in \Gamma$, on pose

$$C_A = \{vu \in \Sigma^* \mid S \Rightarrow^* uAv\}$$

Question 2 Montrer que pour tout $A \in \Gamma$, C_A est hors-contexte.

Question 3 Montrer que pour tout $u, v \in \Sigma^*$ tel que $uv \in L \setminus \{\varepsilon\}$, il existe $A \in \Gamma$ et $x, y \in \Sigma^*$ et $x', y' \in (\Sigma \cup \Gamma)^*$ tel que $S \Rightarrow^* xAy$ et $A \Rightarrow^1 x'y'$ avec $xx' \Rightarrow^* u$ et $yy' \Rightarrow^* v$

Soit $A \to w_1 ... w_n$ une règle de G, pour tout $0 \leq i \leq n$, on pose $L_A^{i-} = \{u \in \Sigma^* \mid w_1 ... w_i \Rightarrow^* u\}$ et $L_A^{i+} = \{u \in \Sigma^* \mid w_{i+1} ... w_n \Rightarrow^* u\}$. On remarque qu'ils sont presque par définition hors conexte.

Question 4 En considérant le langage suivant, montrer que $\mathrm{Circ}(L)$ est hors-contexte:

$$E \cup \bigcup_{(A \to w_1 ... w_n) \in R} \bigcup_{0 \leq i \leq n} L_A^{i-} C_A L_A^{i+}$$

où $E = \{\varepsilon\}$ si $\varepsilon \in L$, et $E = \emptyset$ sinon.

Théorème de Chomsky-Schützenberger[1]

Soit $n \in \mathbb{N}^*$, on définit $\Sigma_n = \{a_1, \overline{a_1}, ..., a_n, \overline{a_n}\}$. Les lettres a_i représentent des parenthèses ouvrantes et les lettres $\overline{a_i}$ représentent des parenthèses fermanantes.

On considère la grammaire $\mathcal{G}_n$ engendrée par les règles

$$S \to a_1 S \overline{a_1} S \mid ... \mid a_n S \overline{a_n} S \mid \varepsilon$$

On pose $D_n = L(\mathcal{G}_n)$ le langage des mots de dyck engendrée par $\mathcal{G}_n$.

1. Donner un arbre de dérivation pour le mot $a_1 a_2 \overline{a_2 a_1} a_3 \overline{a_3}$

On dit que $\varphi : \Sigma_1^* \to \Sigma_2^*$ et un morphisme de mot si $\forall uv \in \Sigma_1^*, \varphi(uv) = \varphi(u)\varphi(v)$

2. Donner une expression régulière dénotant $\varphi(D_1)$ pour φ telle que $\varphi(a_1) = aa$ et $\varphi(\overline{a_1}) = a$
3. Montrer que si L est régulier alors $\varphi(L)$ aussi. Meme question si L est hors-contexte.

On s'intéresse à montrer le théorème de Chomsky-Schützenberger:

> Un langage L est hors-contexte si et seulement il existe un langage régulier K, un langage de Dyck D_n et un morphisme alphabétique φ tels que $L = \varphi(D_n \cap K)$.

4. Montrer que l'intersection d'un langage régulier et d'un langage hors-contexte est hors-contexte. En déduire un sens du théorème.

On admet que tout grammaire peut être mise en forme normale de Chompsky, c'est à dire que toutes les règles de la grammaire sont soit $X \to YZ$, soit $X \to \alpha$ ou soit $S \to \varepsilon$ avec $Y, Z \in \Gamma$ et $\alpha \in \Sigma$ et S le symbole initial.

Soit $G = (\Sigma, \Gamma, S, R)$ une grammaire hors-contexte sous forme normale de chompsy. On ordonne les $k := |R|$ règles $r_1, ..., r_k$. On pose $G' = (\Sigma', \Gamma', S, R')$ avec:

$$\Sigma' = \Sigma \cup \{\overline{\alpha} : \alpha \in \Sigma\} \cup \bigcup_{i \in [k]} \left\{a_i, \overline{a_i}, b_i, \overline{b_i}, c_i, \overline{c_i}\right\}$$

Et les règles R' sont:
- $X \to a_i b_i Y \overline{b_i} c_i Z \overline{c_i a_i}$ pour chaque $r_i = X \to YZ$
- $X \to \alpha\overline{\alpha}$ pour toute règle de la forme $X \to \alpha$

5. Donner un morphisme de mot φ tel que $L(G) = \varphi(L(G'))$
6. Proposer un langage régulier K tel que $K \cap D_n = L(G)$. Conclure la preuve du théorème.

[1] Exos de la martinière MPI

Image de Parikh de langage[2]

1. Montrer que tous les langages régulier sont hors-contexte.

On cherche à montrer que la réciproque est vraie sur $\Sigma = \{a\}$. Pour cela, on va étudier l'image de parikh d'un langage.

Soit $\Sigma = \{a_1, ..., a_k\}$ un alphabet fini. Pour un mot $w \in \Sigma^*$, on définit son *image de Parikh* par $\Psi(w) = \left(|w|_{a_1}, ..., |w|_{a_k}\right) \in \mathbb{N}^k$. Pour un langage $L \subseteq \Sigma^*$, on pose $\Psi(L) = \{\Psi(w) : w \in L\} \subseteq \mathbb{N}^k$.

Un ensemble $S \subseteq \mathbb{N}^k$ est dit *linéaire* s'il existe $b, p_1, ..., p_m \in \mathbb{N}^k$ tels que

$$S = \{b + \lambda_1 p_1 + ... + \lambda_m p_m : (\lambda_1, ..., \lambda_m) \in \mathbb{N}^m\}.$$

Un ensemble est dit *semi-linéaire* s'il est union finie d'ensembles linéaires.

1. Calculer l'image de Parikh des langages suivants:
   - $L_1 = \{a^n b^n : n \in \mathbb{N}\}$,
   - $L_2 = \{a^n b^m c^{n+m} : n, m \in \mathbb{N}\}$,
   - $L_3 = L(ab^*)$,
2. Montrer que si $A, B \subseteq \mathbb{N}^k$ sont semi-linéaire, alors $A \cup B$ et $A + B$ le sont aussi.
3. Montrer que si L est régulier, alors $\Psi(L)$ est semi-linéaire.

On admet que tout grammaire peut être mise en forme normale de Chompsky, c'est à dire que toutes les règles de la grammaire sont soit $X \to YZ$, soit $X \to \alpha$ ou soit $S \to \varepsilon$ avec $Y, Z \in \Gamma$ et $\alpha \in \Sigma$ et S le symbole initial. Soit $G = (\Sigma, \Gamma, R, S)$ une grammaire hors-contexte sous forme normale de Chompsky.

On dit d'un arbre de dérivation T d'un mot $w \in (\Sigma \cup \Gamma)^*$ qu'il est un *arbre plein* si $w \in \Sigma^*$ et que c'est un *arbre-bloc* un arbre de dérivation si w est de la forme uAv pour $A \in \Gamma$ et $u, v \in \Sigma^*$. Pour un arbre plein T on pose $\Psi(T) = \Psi(w)$ et pour un arbre-bloc T_w de forme $A \Rightarrow^* uAv$, on pose $\Psi(T) = \Psi(uv)$.

4. Donner un exemple d'arbre-bloc pour la grammaire $S \to aSb \mid \varepsilon$. Calculer son image de Parikh.

Soit T un arbre de dérivation. On dit que T est *minimal* s'il ne contient pas strictement un arbre-bloc comme sous-arbre.

5. Montrer qu'un arbre minimal ne peut pas contenir deux occurrences d'un même non-terminal sur une même branche.

TODO: FINISH.

[2]Tiré d'une preuve simplifié https://arxiv.org/pdf/2301.00047