Informatique théorique III
Examen hiver solution

Prof. Nestmann, 2005

Nom :

Durée : 225 min. **Rendu à :**

Points :

Veuillez s'il vous plaît prendre connaissance des points suivants :

– Résolvez chaque problème sur une feuille *séparée*.
– Notez votre nom et prénom sur chaque feuille rendue et rendez la donnée de l'examen.
– Les questions d'un problème sont indépendantes. Si vous ne trouvez pas la réponse à une question vous pouvez admettre le résultat et passer à la question suivante.
– On attachera la plus grande importance à la clarté, à la précision et à la concision de la rédaction. Relisez donc vos preuves d'un oeil critique et incrédule.

L'obtention d'au moins 50% des points garantit la réussite de l'examen.

---

1. Ambiguïté

(30 min., 20%)

Soit $\Sigma = \{\mathcal{A}, \mathcal{B}, \mathcal{C}, \neg, \wedge, \rightsquigarrow,), (\}$ et $G = (\{S\}, \Sigma, P, S)$ avec P tel que :

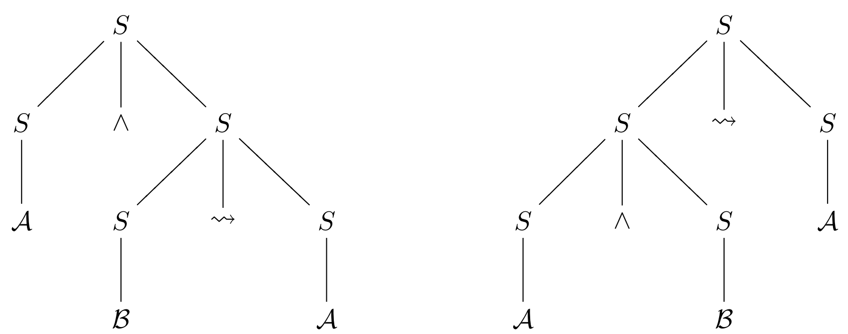
$$S \to \mathcal{A} \mid \mathcal{B} \mid \mathcal{C} \mid \neg S \mid S \wedge S \mid S \rightsquigarrow S \mid (S)$$

Les symboles $\neg, \wedge, \rightsquigarrow$ représentent réspectivement, la négation, la conjonction et l'implication logique. La grammaire G génère l'ensemble des formules de la logique propositionelle dont les variables sont $\mathcal{A}, \mathcal{B}, \mathcal{C}$.

1. Démontrer l'ambiguïté de la grammaire.
2. Donner (sans preuve) une grammaire G' telle que :
   (a) $\mathrm{L}(G') = \mathrm{L}(G)$
   (b) G' soit non ambiguë.
   (c) L'ordre de priorité des opérateurs soit, du plus faible au plus fort, $\rightsquigarrow, \wedge, \neg$.
   (d) L'opérateur $\rightsquigarrow$ soit associatif à droite, et $\wedge$ associatif à gauche. Par exemple $\mathcal{A} \rightsquigarrow \mathcal{A} \rightsquigarrow \mathcal{A}$ se lit $\mathcal{A} \rightsquigarrow (\mathcal{A} \rightsquigarrow \mathcal{A})$ et $\mathcal{A} \wedge \mathcal{A} \wedge \mathcal{A}$ se lit $(\mathcal{A} \wedge \mathcal{A}) \wedge \mathcal{A}$.
3. Définir une machine M (AFD, AFN, AAP ou MT) telle que $\mathrm{L}(M) = \mathrm{L}(G)$.

Preuve.

1. Le mot $\mathcal{A} \wedge \mathcal{B} \rightsquigarrow \mathcal{A}$ admet les deux arbres d'analyse suivants.



Par définition G est donc ambiguë.

Autre solution, le mot $\neg\mathcal{A} \wedge \mathcal{B}$ admet deux dérivations disctinctes la plus à gauche :

$$S \Rightarrow \neg S \Rightarrow \neg S \wedge S \Rightarrow \neg\mathcal{A} \wedge S \Rightarrow \neg\mathcal{A} \wedge \mathcal{B}$$
$$S \Rightarrow S \wedge S \Rightarrow \neg S \wedge S \Rightarrow \neg\mathcal{A} \wedge S \Rightarrow \neg\mathcal{A} \wedge \mathcal{B}$$

2. La grammaire $G' = (\{E, T, F\}, \Sigma, P', E)$ avec P' tel que :

$$\begin{aligned} E &\to T \rightsquigarrow E \mid T \\ T &\to T \wedge F \mid F \\ F &\to \neg F \mid (E) \mid \mathcal{A} \mid \mathcal{B} \mid \mathcal{C} \end{aligned}$$

a les propriétés désirées.

3. On définit $M = (\{q\}, \Sigma, \Sigma \cup \{S\}, \Delta, q, E)$ qui accepte par le critère de la pile vide avec Δ tel que :

$$\begin{aligned} \Delta(q, \mathbf{e}, S) &= \left\{(q, \mathcal{A}), (q, \mathcal{B}), (q, \mathcal{C}), (q, \neg S), (q, S \wedge S), (q, S \rightsquigarrow S), (q, (S))\right\} \\ \Delta(q, \mathcal{A}, \mathcal{A}) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, \mathcal{B}, \mathcal{B}) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, \mathcal{C}, \mathcal{C}) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, \neg, \neg) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, \wedge, \wedge) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, \rightsquigarrow, \rightsquigarrow) &= \left\{(q, \epsilon)\right\} \\ \Delta(q, (, () &= \left\{(q, \epsilon)\right\} \\ \Delta(q,),)) &= \left\{(q, \epsilon)\right\} \end{aligned}$$

■

2. Racine n^{e} d'un langage (60 min., 25%)

Soit Σ un alphabet et $M = (Q, \Sigma, \delta, q_1, F)$ un AFD. On rappelle d'abord que la fonction de transition étendue $\widehat{\delta}$ a la propriété suivante :

$$\forall q \in Q \,.\, \forall u, v \in \Sigma^* \,.\, \widehat{\delta}(q, u \cdot v) = \widehat{\delta}(\widehat{\delta}(q, u), v)$$

On dira que M est *régulier à gauche* si

$$\forall u, v \in \Sigma^* \,.\, \Big(\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, v) \Rightarrow \big(\forall w \in \Sigma^* \,.\, \widehat{\delta}(q_1, w \cdot u) = \widehat{\delta}(q_1, w \cdot v)\big)\Big)$$

1. Soit $M_1 = (\{q_1, q_2, q_3\}, \{a, b\}, \delta, q_1, \{q_3\})$ l'AFD défini dans la figure 1.

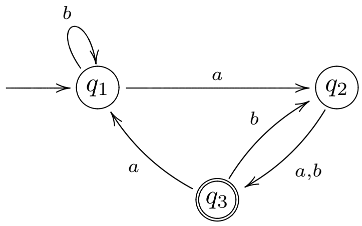


FIG. 1 – L'automate M_1

Montrer que M_1 n'est pas régulier à gauche en donnant deux mots u et v tels que $\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, v)$ et un mot w tel que $\widehat{\delta}(q_1, w \cdot u) \neq \widehat{\delta}(q_1, w \cdot v)$.

Dans la suite, on admet le théorème suivant :
Théorème Pour tout AFD M, il existe un AFD M^g régulier à gauche tel que $\mathrm{L}(M^g) = \mathrm{L}(M)$.

2. Soit $M = (Q, \Sigma, \delta, q_1, F)$ un AFD régulier à gauche.
Montrer (par induction sur n) que

$$\forall u, v \in \Sigma^* \,.\, \Big(\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, v) \Rightarrow \big(\forall n \in \mathbb{N}^* \,.\, \widehat{\delta}(q_1, u^n) = \widehat{\delta}(q_1, v^n)\big)\Big)$$

3. **Définition**($\sqrt[n]{L}$) Soit L un langage sur Σ. On définit, pour $n \in \mathbb{N}^*$, la racine n^{e} de L par $\sqrt[n]{L} = \{u \in \Sigma^* \mid u^n \in L\}$.

Définition($\sqrt[n]{M}$) Soit $M = (Q, \Sigma, \delta, q_1, F)$ un AFD. On définit, pour $n \in \mathbb{N}^*$,
- $F_n = \Big\{q \in Q \mid \exists u \in \Sigma^* \,.\, \big(\widehat{\delta}(q_1, u) = q \ \wedge\ \widehat{\delta}(q_1, u^n) \in F\big)\Big\}$
- $\sqrt[n]{M} = (Q, \Sigma, \delta, q_1, F_n)$

Soit $M = (Q, \Sigma, \delta, q_1, F)$ un AFD et $n \in \mathbb{N}^*$.

(a) Montrer que $\mathrm{L}(\sqrt[n]{M}) = \Big\{w \in \Sigma^* \mid \exists u \in \Sigma^* \,.\, \big(\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, w) \ \wedge\ u^n \in \mathrm{L}(M)\big)\Big\}$.

(b) On suppose maintenant que M est régulier à gauche
   i. Est-ce que $\sqrt[n]{M}$ est régulier à gauche ? Justifier rapidement.
   ii. Montrer que $\mathrm{L}(\sqrt[n]{M}) \subseteq \sqrt[n]{\mathrm{L}(M)}$ (en utilisant 2).
   iii. Montrer que $\sqrt[n]{\mathrm{L}(M)} \subseteq \mathrm{L}(\sqrt[n]{M})$.
   iv. En déduire que $\mathrm{L}(\sqrt[n]{M}) = \sqrt[n]{\mathrm{L}(M)}$.

4. Conclure en montrant que si $L \subseteq \Sigma^*$ est régulier, alors pour tout $n \in \mathbb{N}^*$ le langage $\sqrt[n]{L}$ est régulier. □

Solution.

1. Considérer (par exemple) $u = aa$, $v = ab$ et $w = a$.
2. On montre ce résultat par induction sur $n \in \mathbb{N}^*$.

   Plus précisément, soit $u, v \in \Sigma^*$ tel que (*) $\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, v)$. On montre que $\forall n \in \mathbb{N}^* . P(n)$ où $P(n)$ est le prédicat défini par

   $$P(n) : \widehat{\delta}(q_1, u^n) = \widehat{\delta}(q_1, v^n)$$

   – Cas de base : $n = 1$.
     L'hypothèse (*) est exactement $P(1)$ donc $P(1)$ est vrai.
   – Cas inductif.
     Soit $n \in \mathbb{N}^*$ tel que $P(n)$ est vrai. Montrons que $P(n+1)$ est vrai.
     On a

   $$\begin{aligned}
   \widehat{\delta}(q_1, u^{n+1}) &= \widehat{\delta}(q_1, u^n \cdot u) && \text{par définition de } u^{n+1} \\
   &= \widehat{\delta}(q_1, u^n \cdot v) && \text{car } M \text{ est régulier à gauche et (*)} \\
   &= \widehat{\delta}(\widehat{\delta}(q_1, u^n), v) && \text{propriété de la fonction de transition étendue} \\
   &= \widehat{\delta}(\widehat{\delta}(q_1, v^n), v) && \text{par l'hypothèse d'induction } P(n) \\
   &= \widehat{\delta}(q_1, v^n \cdot v) && \text{propriété de la fonction de transition étendue} \\
   &= \widehat{\delta}(q_1, v^{n+1}) && \text{par définition de } v^{n+1}
   \end{aligned}$$

     D'où $P(n+1)$.

   On conclut par principe d'induction que $\forall n \in \mathbb{N}^* : P(n)$. D'où le résultat demandé.
3. (a) Soit $w \in \Sigma^*$.

   $$\begin{aligned}
   w \in \mathrm{L}(\sqrt[n]{M}) &\Leftrightarrow \widehat{\delta}(q_1, w) \in F_n && \text{par définition de } \mathrm{L}(\sqrt[n]{M}) \\
   &\Leftrightarrow \exists u \in \Sigma^* . \big(\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, w) \ \wedge\ \widehat{\delta}(q_1, u^n) \in F\big) \\
   &\Leftrightarrow \exists u \in \Sigma^* . \big(\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, w) \ \wedge\ u^n \in \mathrm{L}(M)\big) && \text{par définition de } \mathrm{L}(M)
   \end{aligned}$$

   D'où l'égalité.

   (b) i. Oui $\sqrt[n]{M}$ est régulier à gauche car la fonction de transition et l'état initial de $\sqrt[n]{M}$ sont les mêmes que ceux de M qui est régulier à gauche et la propriété de régularité à gauche ne dépend que de ces deux paramètres.

   ii. Soit $u \in \mathrm{L}(\sqrt[n]{M})$.
   Par la question 3a, il existe $v \in \Sigma^*$ tel que $\widehat{\delta}(q_1, u) = \widehat{\delta}(q_1, v)$ et $v^n \in \mathrm{L}(M)$. On a donc $\widehat{\delta}(q_1, v^n) \in F$.
   D'après la question 2, on a, puisque $\sqrt[n]{M}$ est régulier à gauche et que $\widehat{\delta}(q_1, v) = \widehat{\delta}(q_1, u)$, que $\widehat{\delta}(q_1, v^n) = \widehat{\delta}(q_1, u^n)$.
   Comme $\widehat{\delta}(q_1, v^n) \in F$, on a donc $\widehat{\delta}(q_1, u^n) \in F$, c'est-à-dire que $u^n \in \mathrm{L}(M)$.
   D'où $u \in \sqrt[n]{\mathrm{L}(M)}$ et l'inclusion.

   iii. Soit $u \in \sqrt[n]{\mathrm{L}(M)}$.
   Par définition, on a $u^n \in \mathrm{L}(M)$ et donc $\widehat{\delta}(q_1, u^n) \in F$.
   Par définition de F_n, on a donc $\widehat{\delta}(q_1, u) \in F_n$ et donc $u \in \mathrm{L}(\sqrt[n]{M})$. D'où l'inclusion.

iv. Les points ii et iii permettent de conclure que $\mathrm{L}(\sqrt[n]{M}) = \sqrt[n]{\mathrm{L}(M)}$.

4. Soit $L \subseteq \Sigma^*$ un langage régulier et $n \in \mathbb{N}^*$. Par définition, il existe un AFD M tel que $\mathrm{L}(M) = L$. Il existe donc un automate M^g régulier à gauche tel que $\mathrm{L}(M_g) = \mathrm{L}(M) = L$. Par la question 3, l'automate $\sqrt[n]{M^g}$ est un AFD tel que $\mathrm{L}(\sqrt[n]{M^g}) = \sqrt[n]{\mathrm{L}(M^g)} = \sqrt[n]{L}$. Donc $\sqrt[n]{L}$ est reconnu par un AFD, ce qui montre que $\sqrt[n]{L}$ est régulier.

■

3. Régularité (60 min.,25%)

Rappellons que le *mot miroir* $\overline{w}$ d'un mot w est l'unique mot tel que :

$$|\overline{w}| = |w| \text{ et } \forall i \in [1, |w|] \,.\, (\overline{w})_i = (w)_{|w|+1-i}$$

Soit $\Sigma = \{a, b\}$, on considère les langages suivants avec $m \in \mathbb{N}$:

$$L_m = \left\{ uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^m \right\}$$

1. En utilisant la propriété suivante :

$$w \in L_m \Rightarrow \left(l = |w| - m \text{ est pair } \wedge \ \forall i \in [1, l/2] \,.\, (w)_i = (w)_{|w|+1-i} \right) \qquad (\star)$$

   Montrer que pour tout $n, m \in \mathbb{N}^*$ et $n > i > 0$, on a $b^{n-i}a^m b^n \notin L_m$ (l'induction est inutile).
2. Le langage L_0 n'est pas régulier. En effet, c'est l'ensemble des palindromes de longueur paire sur Σ.

   Montrer que pour tout $m > 0$, L_m n'est pas non plus régulier. Pour cela, prendre $m > 0$, attention c'est tout ce que l'on sait de m, et montrer que L_m n'est pas régulier.
3. Considérons maintenant le langage L_* suivant :

$$L_* = \left\{ uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^* \right\}$$

   (a) Donner une expression régulière α telle que $L_* = \mathrm{L}(\alpha)$.
   (b) Démontrer que $L_* = \mathrm{L}(\alpha)$.
4. Le langage $L = \bigcup_{m \in \mathbb{N}} L_m$ est-il régulier ? Justifier. □

Preuve.

1. Soit $n, m \in \mathbb{N}^*$ et $i > 0$. Posons $w' = b^{n-i}a^m b^n$. Nous avons $|w'| = 2n + m - i$. Posons $l = |w'| - m = 2n + m - i - m = 2n - i$. Il y deux cas à considérer :
   (a) l impair. Dans ce cas la condition (**??**) n'est pas satisfaite et donc $w' \notin L_m$.
   (b) l pair. Dans ce cas prenons $j = n - i + 1$. Sachant $m > 0$, on a $(w')_j = (w')_{n-i+1} = a$. Par ailleurs étant donné que $i > 0$ et $2n - i$ est pair, on a :

$$j = n - i + 1 \leq n - i/2 = l/2$$

   Dès lors, par (**??**), pour que w' appartienne à L_m, il est nécessaire que $(w')_j = (w')_{|w'|+1-j}$. Or nous avons

$$\begin{aligned} (w)_{|w'|+1-j} &= (w)_{2n+m-i+1-n+i-1} \\ &= (w)_{n+m} \\ &= b \neq a \end{aligned}$$

   La condition (**??**) n'est pas satisfaite et donc $w' \notin L_m$.

   Dans les deux cas nous avons $w' \notin L_m$.
2. Prenons $m \in \mathbb{N}^*$ et montrons que L_m n'est pas régulier en utilisant la contraposée du lemme de gonflement.

   Soit $n \in \mathbb{N}$, construisons le mot $w = b^{(n+1)}a^m b^{(n+1)} \in L_m$.

   Soit $w = xyz$ une décomposition de w telle que $y \neq \epsilon$ et $|xy| \leq n$. Sachant que $w = b^{n+1}a^m b^{n+1}$ nous avons $y = b^i$ avec $i > 0$.

   Prenons $k = 0$ et formons $w' = xy^k z = xy^0 z = b^{n+1-i}a^m b^{n+1}$. Par le point précédent on sait que $w' \notin L_m$ et donc L_m n'est pas régulier.

3. (a) $\alpha = a(a+b)^*a + b(a+b)^*b$.

   (b) $L_* = \mathrm{L}(\alpha)$. En effet,

      i. $w \in \mathrm{L}(\alpha) \Rightarrow w \in L_*$

$$\begin{aligned} w \in \mathrm{L}(\alpha) &\Leftrightarrow w \in \mathrm{L}(a(a+b)^*a) \cup \mathrm{L}(b(a+b)^*b) \\ &\Leftrightarrow w \in \big(\{a\} \cdot \{a,b\}^* \cdot \{a\}\big) \cup \big(\{b\} \cdot \{a,b\}^* \cdot \{b\}\big) \\ &\Leftrightarrow w \in \{ava \mid v \in \Sigma^*\} \cup \{bvb \mid v \in \Sigma^*\} \end{aligned}$$

      sachant que $a = \overline{a}$, $b = \overline{b}$ et $a, b \in \Sigma^+$ nous avons,

$$\Rightarrow w \in \big\{uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^*\big\}$$

      ii. $w \in \mathrm{L}(\alpha) \Leftarrow w \in L_*$

$$\begin{aligned} w \in L_* &\Leftrightarrow w \in \big\{uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^*\big\} \\ &\Leftrightarrow w \in \{auv\overline{u}a \mid u, v \in \Sigma^*\} \cup \{buv\overline{u}b \mid u, v \in \Sigma^*\} \end{aligned}$$

      sachant que $\{uv\overline{v} \mid u, v \in \Sigma^*\} \subseteq \Sigma^*$ nous avons,

$$\begin{aligned} &\Rightarrow w \in \{ava \mid v \in \Sigma^*\} \cup \{bvb \mid v \in \Sigma^*\} \\ &\Leftrightarrow w \in \big(\{a\} \cdot \{a,b\}^* \cdot \{a\}\big) \cup \big(\{b\} \cdot \{a,b\}^* \cdot \{b\}\big) \\ &\Leftrightarrow w \in \mathrm{L}(a(a+b)^*a) \cup \mathrm{L}(b(a+b)^*b) \\ &\Leftrightarrow w \in \mathrm{L}(\alpha) \end{aligned}$$

   De (i) et (ii) nous déduisons l'égalité $L_* = L(\alpha)$.

4. Étant donné que,

$$\begin{aligned} L &= \bigcup_{m \in \mathbb{N}} L_m \\ &= \bigcup_{m \in \mathbb{N}} \big\{uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^m\big\} \\ &= \big\{uv\overline{u} \mid u \in \Sigma^+ \wedge v \in \Sigma^*\big\} \\ &= L_* \end{aligned}$$

   et que le point précédent a prouvé la régularité de L_*, L est régulier.

∎

4. Indécidabilité (75 min.,30%)

Dans ce problème, on s'intéresse, pour $i \in \{0, 2, 3\}$ à la décidabilité du problème INCLUS$_i$:

INCLUS$_i$: « Étant donnés deux langages $L, L' \in \mathcal{L}_i$, est-ce que $L \subseteq L'$? »

On définit aussi, pour $i \in \{0, 2, 3\}$ le problème VIDE$_i$:

VIDE$_i$: « Étant donné un langage $L \in \mathcal{L}_i$, est-ce que $L = \emptyset$? »

On rappelle que
- $\mathcal{L}_0$ est l'ensemble des langages semi-décidables (acceptés par les machines de Turing).
- $\mathcal{L}_2$ est l'ensemble des langages non-contextuels (générés par les grammaires non-contextuelles).
- $\mathcal{L}_3$ est l'ensemble des langages réguliers (acceptés par les automates finis déterministes).

0. Soit A et B deux ensembles. Montrer que $A \subseteq B \Leftrightarrow A \cap \overline{B} = \emptyset$.
1. Dans cette question, on traite le cas $i = 3$, le cas des langages réguliers.
   (a) Soit $M = (Q, \Sigma, \delta, q_0, F)$ un AFD.
   Donner un AFD $\overline{M}$ tel que $\mathrm{L}(\overline{M}) = \overline{\mathrm{L}(M)}$. Justifier.
   (b) On admet l'existence d'un algorithme qui, étant donné un AFD M détermine si, oui ou non, on a $\mathrm{L}(M) = \emptyset$ (c'est-à-dire un algorithme qui résout le problème VIDE$_3$). Décrire alors un algorithme qui résout le problème INCLUS$_3$, c'est-à-dire un algorithme qui prend en entrée deux AFD M et M' et détermine si, oui ou non, on a $\mathrm{L}(M) \subseteq \mathrm{L}(M')$.
2. Dans cette question, on traite le cas $i = 2$.
   (a) Soit $\Sigma = \{a, b\}$. Donner une grammaire G non-contextuelle telle que le langage de G soit l'ensemble de tous les mots sur Σ qui ne sont pas des palindromes, c'est-à-dire telle que $\mathrm{L}(G) = \{w \in \Sigma^* \mid w \neq \overline{w}\}$.

   Dans la suite, on admet que :
   Si Σ est un alphabet quelconque, alors $\{w \in \Sigma^* \mid w \neq \overline{w}\}$ est un langage non-contextuel (c'est le résultat précédent généralisé au cas d'un alphabet quelconque).
   (b) Montrer que s'il existe un algorithme qui, étant donné deux grammaires non-contextuelles G et G', détermine si, oui ou non, on a $\mathrm{L}(G) \subseteq \mathrm{L}(G')$, alors il existe un algorithme qui détermine si, oui ou non, une instance du PCP admet une solution.
   On pourra s'intéresser en particulier au cas où G' génère un langage de la forme $\{w \in \Sigma^* \mid w \neq \overline{w}\}$.
   Comme PCP est indécidable – et donc qu'il n'existe pas de tel algorithme pour PCP – , cela montre qu'il n'existe pas de tel algorithme pour INCLUS$_2$.
   (c) Il existe un algorithme qui résout le problème VIDE$_2$, c'est-à-dire un algorithme qui étant donné une grammaire non-contextuelle G détermine si, oui ou non, on a $\mathrm{L}(G) = \emptyset$.
   Expliquer brièvement pourquoi une construction similaire à celle faite en 1b ne peut pas s'appliquer et donc ne contredit pas le point précédent.
3. On s'intéresse maintenant au cas $i = 0$.

(a) En utilisant le théorème de Rice, montrer que le problème $\textsc{vide}_0$ est indécidable.

(b) En déduire, par réduction de $\textsc{vide}_0$ à $\textsc{inclus}_0$ que $\textsc{inclus}_0$ est indécidable. On représentera le problème $\textsc{vide}_0$ par le langage $\{[M] \mid \mathrm{L}(M) = \emptyset\}$ et le problème $\textsc{inclus}_0$ par le langage $\{[(M, M')] \mid \mathrm{L}(M) \subseteq \mathrm{L}(M')\}$ (où $[(M, M')]$ est un codage (injectif) du couple (M, M')).

□

Solution.

0. Soit A et B deux ensembles. On montre en fait $A \not\subseteq B \Leftrightarrow A \cap \overline{B} \neq \emptyset$.

$$\begin{aligned} A \not\subseteq B &\Leftrightarrow \exists x \,.\, x \in A \wedge x \notin B \\ &\Leftrightarrow \exists x \,.\, x \in A \wedge x \in \overline{B} \\ &\Leftrightarrow \exists x \,.\, x \in A \cap \overline{B} \\ &\Leftrightarrow A \cap \overline{B} \neq \emptyset \end{aligned}$$

Ou alors,

$$\begin{aligned} A \subseteq B &\Leftrightarrow \forall x \,.\, x \in A \Rightarrow x \in B \\ &\Leftrightarrow \forall x \,.\, \neg(x \in A) \vee x \in B \\ &\Leftrightarrow \forall x \,.\, \neg(x \in A \wedge x \notin B) \\ &\Leftrightarrow \forall x \,.\, \neg(x \in A \cap \overline{B}) \\ &\Leftrightarrow \neg \exists x \,.\, x \in A \cap \overline{B} \\ &\Leftrightarrow A \cap \overline{B} = \emptyset \end{aligned}$$

1. (a) Soit $M = (Q, \Sigma, \delta, q_0, F)$ un AFD.
On définit $\overline{M} = (Q, \Sigma, \delta, q_0, \overline{F})$.
Soit $w \in \Sigma^*$

$$\begin{aligned} w \in \mathrm{L}(\overline{M}) &\Leftrightarrow \widehat{\delta}(q_0, w) \in \overline{F} \\ &\Leftrightarrow \widehat{\delta}(q_0, w) \notin F \\ &\Leftrightarrow \neg(\widehat{\delta}(q_0, w) \in F) \\ &\Leftrightarrow \neg(w \in \mathrm{L}(M)) \\ &\Leftrightarrow w \notin \mathrm{L}(M) \\ &\Leftrightarrow w \in \overline{\mathrm{L}(M)} \end{aligned}$$

(b) Étant donnés deux AFD M et M', on construit par la question précédente l'AFD $\overline{M'}$. D'après le cours on peut alors construire l'AFD $M \otimes \overline{M'}$. On peut alors utiliser l'algorithme résolvant $\textsc{vide}_3$ pour déterminer si, oui ou non, on a $\mathrm{L}(M \otimes \overline{M'}) = \emptyset$. C'est équivalent, par la question 0, à déterminer si, oui ou non, on a $\mathrm{L}(M) \subseteq \mathrm{L}(M')$ car $\mathrm{L}(M \otimes \overline{M'}) = \mathrm{L}(M) \cap \overline{\mathrm{L}(M')}$.

2. (a) La grammaire $G = (\{S, S'\}, \Sigma, P, S)$ avec les productions P définies par

$$\begin{aligned} S &\to aSa \mid bSb \mid aS'b \mid bS'a \\ S' &\to \epsilon \mid aS' \mid bS' \end{aligned}$$

convient.

(b) Supposons qu'il existe un algorithme qui, étant donné deux grammaires non-contextuelles G et G', détermine si, oui ou non, on $\mathrm{L}(G) \subseteq \mathrm{L}(G')$.

Montrons alors qu'on peut trouver un algorithme permettant de résoudre le PCP.

Soit (A, B) un instance du PCP. Comme dans la série 14, on construit une grammaire non-contextuelle G telle que G génère au moins un palindrome si et seulement si (A, B) a une solution. Soit Σ l'alphabet de G. On considère alors une grammaire non-contextuelle G' telle que $\mathrm{L}(G') = \{w \in \Sigma^* \mid \overline{w} \neq w\}$. En utilisant l'algorithme supposé, on peut déterminer si, oui ou non, on a $\mathrm{L}(G) \subseteq \mathrm{L}(G')$. C'est-à-dire, on peut déterminer si, oui ou non, $\mathrm{L}(G)$ contient au moins un palindrome. C'est-à-dire, on peut déterminer si, oui ou non, (A, B) a une solution.

(c) On ne peut pas appliquer la même méthode car les langages non-contextuels ne sont pas stables par complémentation et intersection.

3. (a) Ce problème est équivalent à décider la propriété $\phi : \mathcal{L}_0 \to \{0, 1\}$

$$L \mapsto \begin{cases} 1 & \text{si } L = \emptyset \\ 0 & \text{sinon} \end{cases}$$

ϕ est une propriété non-triviale car $\phi(\emptyset) = 1$ et $\phi(\{0\}) = 0$. D'après le théorème de Rice, ϕ — et donc VIDE_0 — est indécidable.

(b) Rappelons d'abord que M^{triv} est telle que $\mathrm{L}(M^{\text{triv}}) = \emptyset$. Soit M_0 la machine de Turing qui entre dans un état accepteur quel que soit le mot d'entrée. On a $\mathrm{L}(M_0) = \{0, 1\}^*$. On définit la réduction σ de VIDE_0 à INCLUS_0 comme suit :

$$\sigma : \{0,1\}^* \to \{0,1\}^*$$
$$x \mapsto \begin{cases} [(M, M^{\text{triv}})] & \text{si } x = [M] \\ [(M_0, M^{\text{triv}})] & \text{sinon} \end{cases}$$

Montrons que σ est effectivement une réduction de VIDE_0 à INCLUS_0.
Soit $x \in \{0, 1\}^*$.

– Supposons $x \in \text{VIDE}_0$
  Alors $x = [M]$ pour une certaine MT M telle que $\mathrm{L}(M) = \emptyset$.
  Par construction, $\sigma(x) = [(M, M^{\text{triv}})]$. Comme $\mathrm{L}(M) = \emptyset$, on a $\mathrm{L}(M) \subseteq \emptyset = \mathrm{L}(M^{\text{triv}})$. Donc $\sigma(x) \in \text{INCLUS}_0$.
– Supposons $\sigma(x) \in \text{INCLUS}_0$.
  Par construction, on a forcément $x = [M]$ pour un certain M car sinon $\sigma(x) = [(M_0, M^{\text{triv}})]$ et $\mathrm{L}(M_0) = \{0, 1\}^* \not\subseteq \emptyset = \mathrm{L}(M^{\text{triv}})$ (et donc $\sigma(x) \notin \text{INCLUS}_0$).
  On a donc $\sigma(x) = [(M, M^{\text{triv}})]$. Comme $\sigma(x) \in \text{INCLUS}_0$, on a $\mathrm{L}(M) \subseteq \mathrm{L}(M^{\text{triv}}) = \emptyset$, donc $\mathrm{L}(M) = \emptyset$. Et $x \in \text{VIDE}_0$.

σ étant de plus calculable, on a donc montré que $\text{VIDE}_0 \leq_{\text{red}} \text{INCLUS}_0$. Le problème VIDE_0 étant indécidable, on en déduit qu'il en est de même pour INCLUS_0.

■