

Exercices

1.1 Langages rationnels

Exercice 1.1. Un mot w sur un alphabet A est dit *double* si pour toute lettre $a \in A$, on a $|w|_a \neq 1$.

1. Montrer que tout mot sur un alphabet à n lettres et de longueur au moins égale à 2^n contient un facteur qui est double.
2. Montrer que la borne 2^n est optimale.

Solution.

1. On montre le résultat par récurrence sur n . Le résultat est évident si $n = 1$ car tout mot de longueur au moins 2 sur un alphabet unaire est double. Soit w un mot sur un alphabet à n lettres de longueur au moins égale à 2^n . Il se factorise $w = w_1 w_2$ où $|w_1| = 2^{n-1}$ et $|w_2| \geq 2^{n-1}$. Si w n'est pas double, il existe une lettre a qui n'apparaît pas dans w_1 ou w_2 . Si la lettre a n'apparaît pas par exemple dans w_1 , l'hypothèse de récurrence s'applique à w_1 qui est écrit sur l'alphabet $A \setminus \{a\}$ à $n - 1$ lettres. Le mot w_1 et donc aussi w contient un facteur qui est double.
2. Soit A l'alphabet $\{x_1, x_2, x_3, \dots\}$. On définit la suite $(z_n)_{n \geq 0}$ de mots par $z_0 = \varepsilon$ et $z_n = z_{n-1} x_n z_{n-1}$ pour tout $n \geq 1$. Chaque mot z_n est sur l'alphabet $\{x_1, \dots, x_n\}$ et il est de longueur $2^n - 1$. En revanche, il ne contient aucun facteur double. Ces mots sont appelés mots de Zimin (cf. p. 24).

Exercice 1.2. Montrer que l'ensemble des écritures en base 2 des nombres premiers n'est pas un langage rationnel.

Solution. Soit A l'alphabet $\{0, 1\}$. Pour un mot $w = b_n \dots b_0$ sur A , on note $[w]$ l'entier $\sum_{i=0}^n b_i 2^i$. Pour un entier $n \geq 1$, on note $(n)_2$ l'écriture en binaire de n , c'est-à-dire l'unique mot $w = b_n \dots b_0$ tel que $b_n = 1$ et $n = [w]$. Soit L le langage $\{(p)_2 \mid p \text{ premier}\}$ des écritures en binaire des nombres premiers.

On raisonne par l'absurde et on suppose que le langage L est rationnel. D'après le lemme de l'étoile, il existe un entier n tel que tout mot f de L vérifiant $|f| > n$ se factorise $f = uvw$ avec $0 < |v| < n$ et $uv^*w \subseteq L$. On choisit un nombre premier p tel que $p > 2^n$ de telle sorte que le mot $f = (p)_2$ vérifie $|f| > n$. Le mot f se factorise $f = uvw$ avec $0 < |v| < n$ et $uv^*w \subseteq L$. Pour tout entier $k \geq 1$, on a la formule suivante

$$[uv^k w] = [u] 2^{k|v|+|w|} + [v] \frac{2^{k|v|} - 1}{2^{|v|} - 1} 2^{|w|} + [w]$$

qui donne la valeur en binaire du mot $uv^k w$. On choisit $k = p$ et on a $2^p \equiv 2 \pmod p$ d'après le petit théorème de Fermat. Puisque $|v| < n$, on a $2^{|v|} < p$ et le nombre $2^{|v|} - 1$ est non nul modulo p . On en déduit que $[uv^p w] \equiv [uvw] \pmod p$ et que l'entier $[uv^p w]$ est divisible par p , ce qui contredit l'hypothèse que $uv^p w$ appartient à L .

Exercice 1.3. On définit la relation \sim sur A^* par $w \sim w'$ s'il existe deux mots u et v tels que $w = uv$ et $w' = vu$. On dit que les mots u et v sont *conjugués*.

1. Montrer que la relation \sim est une relation d'équivalence.

On note $[w]$ la classe d'un mot w et pour un langage L , on note $[L]$ le langage $\bigcup_{w \in L} [w]$.

2. Montrer que si L est rationnel, alors le langage $[L]$ est encore rationnel.

Solution. Soit $\mathcal{A} = (Q, A, E, I, F)$ un automate acceptant le langage L . Pour des états p et q , on note $L_{p,q}$ l'ensemble des mots qui étiquettent un chemin de p à q . On a alors la formule suivante

$$L = \bigcup_{i \in I, f \in F} L_{i,f} \quad \text{et} \quad [L] = \bigcup_{i \in I, q \in Q, f \in F} L_{q,f} L_{i,q}$$

qui montre que $[L]$ est rationnel puisque chacun des langages $L_{p,q}$ est bien sûr rationnel.

Exercice 1.4. Soit A l'alphabet $\{a, b\}$. Montrer que tout automate déterministe acceptant le langage $L_n = A^* a A^n$ de l'exemple 1.72 possède au moins 2^{n+1} états.

Solution. On montre que le nombre de quotients à gauche du langage L_n est au moins 2^{n+1} . D'après la lemme 1.81 (p. 45), le nombre d'états de tout automate déterministe acceptant L_n possède au moins 2^{n+1} états.

On montre que pour deux mots distincts w et w' de longueur $n + 1$, les quotients à gauche $w^{-1} L_n$ et $w'^{-1} L_n$ sont différents. Puisque w et w' sont différents, il existe deux mots u et v tels que $w = uav$ et $w' = ubv$ (ou l'inverse). On en déduit que $wu = uavu$ appartient à L_n car $|wu| = n$ alors que $w'u$ n'appartient pas à L_n . Ceci montre que $u \in w^{-1} L_n \setminus w'^{-1} L_n$.

Exercice 1.5. Montrer que l'automate déterministe obtenu par construction par sous-ensembles sur l'automate de l'exemple 1.73 possède 2^n états et que cet automate est minimal.

Solution. Soient $A = \{a, b\}$, n un entier et Q_n l'ensemble $\{1, \dots, n\}$. On considère l'automate $\mathcal{A}_n = (Q_n, A, E_n, \{1\}, \{1\})$ où l'ensemble E_n des transitions est donné par

$$E_n = \{i \xrightarrow{a} i + 1 \mid 1 \leq i \leq n - 1\} \cup \{n \xrightarrow{a} 1\} \\ \cup \{i \xrightarrow{b} i \mid 2 \leq i \leq n\} \cup \{i \xrightarrow{b} 1 \mid 2 \leq i \leq n\}.$$

Soit $\hat{\mathcal{A}}_n$ l'automate déterministe obtenu par la construction par sous-ensembles à partir de l'automate \mathcal{A}_n . Chaque état de $\hat{\mathcal{A}}_n$ est un sous-ensemble de l'ensemble Q_n . L'état initial de $\hat{\mathcal{A}}_n$ est l'ensemble $I = \{1\}$ des états initiaux de \mathcal{A}_n .

Soit J un sous-ensemble de $Q_n = \{1, \dots, n\}$. Si J est vide, soit w_J le mot b et si J est égal à $\{i_1, \dots, i_k\}$ où $i_1 < \dots < i_k$, soit w_J le mot

$$a^{i_k - i_{k-1}} b a^{i_{k-1} - i_{k-2}} b \dots b a^{i_2 - i_1} b a^{i_1 - 1}.$$

On vérifie que $I \cdot w_J = J$ où $I = \{1\}$ est l'ensemble des états initiaux de l'automate \mathcal{A}_n . Ceci montre que que tous les sous-ensembles de $\{1, \dots, n\}$ sont accessibles à partir de I dans l'automate $\hat{\mathcal{A}}_n$.

Soit J et J' deux sous-ensembles distincts de $\{1, \dots, n\}$. On montre que ces deux états ne sont pas équivalents pour la congruence de Nerode. Puisque $J \neq J'$, il existe un état i tel que $i \in J$ et $i \notin J'$ ou l'inverse. L'ensemble $J \cdot a^{n-i}$ contient l'unique état final 1 de \mathcal{A}_n alors que l'ensemble $J' \cdot a^{n-i}$ ne le contient pas. Ces états ne sont donc pas équivalents et l'automate $\hat{\mathcal{A}}_n$ est minimal

Exercice 1.6. Soient E et F deux ensembles munis de quasi-ordres (notés tous les deux \preceq). Une application $f : E \rightarrow F$ est *croissante* si pour tous $x, y \in E$, la relation $x \preceq y$ implique la relation $f(x) \preceq f(y)$.

1. Montrer que si f est croissante et si \preceq est un bon quasi-ordre sur E alors \preceq est un bon quasi-ordre sur $f(E)$.

Soit A un alphabet. Un quasi-ordre \preceq sur A^* est dit *régulier* si pour tous mots $u, u', v, v' \in A^*$, les deux relations $u \preceq u'$ et $v \preceq v'$ entraîne la relation $uv \preceq u'v'$.

2. Montrer qu'un quasi-ordre \preceq est régulier si pour tous mots u, u', v , la relation $u \preceq u'$ implique les deux relations $uv \preceq u'v$ et $vu \preceq vu'$.

Dans les deux questions suivantes, on fixe un quasi-ordre régulier \preceq sur A^* . Soient K et L deux langages.

3. Montrer que si les restrictions de \preceq à K et L sont des bons quasi-ordres, alors sa restriction à KL est aussi un bon quasi-ordre.
4. On suppose dans cette question que $\varepsilon \preceq w$ pour tout mot $w \in A^*$. Montrer que si la restriction de \preceq à K est un bon quasi-ordre, alors sa restriction à K^* est aussi un bon quasi-ordre. On pourra considérer l'ensemble $[K]^*$ formé des suites (w_1, \dots, w_n) finies de mots de K muni de l'ordre \preceq^* .

Solution.

1. D'après le théorème 1.35 (p. 27), il suffit de montrer que toute suite infinie d'éléments de $f(E)$ contient une sous-suite croissante de longueur 2. Soient $(y_n)_{n \geq 0}$ une d'éléments de $f(E)$. Il existe donc une suite d'éléments $(x_n)_{n \geq 0}$ telle que $y_n = f(x_n)$ pour tout $n \geq 0$. Puisque \preceq est un bon quasi-ordre sur E , il existe $i < j$ tels que $x_i \preceq x_j$ et donc $y_i \preceq y_j$ car f est croissante.
2. Il suffit d'écrire $uv \preceq u'v \preceq u'v'$ en utilisant successivement les deux hypothèses.
3. On considère l'ensemble $E = K \times L$ muni du quasi-ordre produit $\preceq \times \preceq$. D'après le lemme 1.37, c'est un bon quasi-ordre sur E . On considère l'application $(u, v) \mapsto uv$ de E dans $F = KL$. Comme le quasi-ordre \preceq est régulier, cette application est croissante. D'après la question 1, la restriction de \preceq à KL est un bon quasi-ordre.
4. On note $[K]^*$ l'ensemble des suites (w_1, \dots, w_n) où chaque w_i est un mot de K . Cet ensemble est à distinguer de K^* qui est l'ensemble des mots $w_1 \dots w_n$ pour

(w_1, \dots, w_n) dans $[K]^*$. D'après le théorème de Higman, l'ordre \preceq^* induit sur $[K]^*$ par l'ordre \preceq sur K est un bon quasi-ordre. Puisque le quasi-ordre \preceq est régulier, l'application de $[K]^*$ dans K^* qui associe à chaque suite (w_1, \dots, w_n) son produit $w_1 \cdots w_n$ est croissante. D'après la question 1, la restriction de \preceq à K^* est un bon quasi-ordre.

Exercice 1.7. On rappelle qu'un quasi-ordre \preceq sur A^* est dit *régulier* si pour tous mots $u, u', v, v' \in A^*$, les deux relations $u \preceq u'$ et $v \preceq v'$ entraîne la relation $uv \preceq u'v'$ (cf. exercice précédent). Le but de cette exercice est de montrer qu'un langage est rationnel si et seulement si il est un idéal d'un bon quasi-ordre régulier.

1. Donner une condition nécessaire et suffisante pour qu'une relation d'équivalence soit un bon quasi-ordre.
2. Donner une condition nécessaire et suffisante pour qu'une relation d'équivalence soit un bon quasi-ordre régulier.
3. En déduire que tout langage rationnel est bien un idéal d'un bon quasi-ordre régulier.

On suppose maintenant que le langage L est un idéal d'un bon quasi-ordre régulier \preceq .

4. Montrer que pour tout $u \in A^*$, $u^{-1}L$ est encore un idéal d'ordre.
5. Montrer que si $u \preceq u'$, alors $u^{-1}L \subseteq u'^{-1}L$.
6. En déduire que L a un nombre fini de quotients à gauche et conclure.

Solution.

1. Une relation d'équivalence est toujours un quasi-ordre car elle est transitive. Deux éléments non équivalents sont incomparables. Pour qu'il n'existe pas d'antichaîne, il est nécessaire que la relation d'équivalence soit d'indice finie (nombre fini de classes). Cette condition est aussi suffisante.
2. La relation d'équivalence est un quasi-ordre régulier si et seulement si elle est compatible avec la concaténation, c'est-à-dire si c'est une congruence. Une relation d'équivalence est un bon quasi-ordre régulier si et seulement si c'est une congruence d'indice fini.
3. Il suffit de remarquer que toute union de classes d'équivalence d'une congruence d'indice fini est un idéal d'ordre pour cette congruence vue comme un quasi-ordre.
4. Soient u, v et v' tels que $v \in u^{-1}L$ et que $v \preceq v'$. Le mot uv appartient donc à L . Comme le quasi-ordre \preceq est régulier, la relation $v \preceq v'$ entraîne la relation $uv \preceq uv'$. Comme L est un idéal d'ordre, uv' appartient aussi à L et v' appartient finalement à $u^{-1}L$. On a donc montré que $u^{-1}L$ est un idéal d'ordre.
5. Soient u, u' et v tels que $u \preceq u'$ et que $v \in u^{-1}L$. Le mot uv appartient donc à L . Comme le quasi-ordre \preceq est régulier, la relation $u \preceq u'$ entraîne la relation $uv \preceq u'v$. Le mot v appartient donc à $u'^{-1}L$. On a donc montré l'inclusion $u^{-1}L \subseteq u'^{-1}L$.
6. Supposons par l'absurde que L ne soit pas rationnel. Il a donc un nombre infini de quotients à gauche. Il existe donc une suite $(u_n)_{n \geq 0}$ de mots tels que les quotients $u_n^{-1}L$ soient distincts. Il est possible d'extraire une suite croissante $(u'_n)_{n \geq 0}$ de la suite $(u_n)_{n \geq 0}$. D'après les deux questions précédentes, les ensembles $u_n'^{-1}L$ forment

une suite croissante d'idéaux. Cette suite doit donc être stationnaire à partir d'un certain rang. Ceci contredit le fait qu'ils soient distincts.

Exercice 1.8. On dit qu'un morphisme $\mu : A^* \rightarrow B^*$ a la *propriété de sélection* si pour tout langage rationnel L , il existe un langage rationnel K inclus dans L tel que μ est injectif sur K et $\mu(K) = \mu(L)$. Le but de cet exercice est de montrer que tout morphisme a la propriété de sélection.

1. Montrer que tout morphisme injectif a la propriété de sélection.
2. Montrer que si les morphismes μ et ν ont la propriété de sélection, alors le morphisme $\mu \circ \nu$ a encore la propriété de sélection.

On appelle *projection* un morphisme $\pi : A^* \rightarrow B^*$ tel que pour toute lettre a de A , on a $\pi(a) = a$ ou $\pi(a) = \varepsilon$.

3. Montrer que pour tout morphisme $\mu : A^* \rightarrow B^*$, il existe un alphabet C , un morphisme injectif $\iota : A^* \rightarrow C^*$ et une projection $\pi : C^* \rightarrow B^*$ tels que $\mu = \pi \circ \iota$.

On appelle *projection élémentaire* une projection $\pi : A^* \rightarrow B^*$ telle qu'il existe une seule lettre a de A vérifiant $\pi(a) = \varepsilon$.

4. Montrer que toute projection est la composition de projections élémentaires.
5. Montrer que toute projection élémentaire a la propriété de sélection.
6. Conclure.

Solution.

1. Si $\mu : A^* \rightarrow B^*$ est injectif, on peut choisir $K = L$ pour tout L . Le morphisme μ a donc la propriété de sélection.
2. D'après les propriétés de clôture des langages rationnels, le langage $\nu(L)$ est rationnel. Puisque μ a la propriété de sélection, il existe K inclus dans $\nu(L)$ tel que μ est injectif sur K et $\mu(K) = \mu(\nu(L))$. On applique alors la propriété de sélection du morphisme ν avec le langage $L' = \mu^{-1}(K) \cap L$ qui est aussi rationnel. Il existe K' inclus dans L' tel que ν est injectif sur K' et $\nu(K') = \nu(L') = K$. Le morphisme $\mu \circ \nu$ est injectif sur K' et $\mu(\nu(K')) = \mu(K) = \mu(\nu(L))$.
3. On peut supposer que les alphabets A et B sont disjoints et on pose $C = A \cup B$. On définit alors le morphisme $\iota : A^* \rightarrow C^*$ par $\iota(a) = a\mu(a)$ et le morphisme $\pi : C^* \rightarrow B^*$ par $\pi(a) = \varepsilon$ si $a \in A$ et $\pi(a) = a$ si $a \in B$. On vérifie facilement que ι est injectif et que $\mu = \pi \circ \iota$.
4. Soit $A = \{a_1, \dots, a_n\}$ et soit $\pi : A^* \rightarrow B^*$ une projection. Pour toute lettre a de A , on définit la projection élémentaire π_a par $\pi_a(a) = \varepsilon$ et $\pi_a(b) = b$ si $b \neq a$. Soit $\{a'_1, \dots, a'_k\}$ l'ensemble $\{a \mid \pi(a) = \varepsilon\}$. On a alors l'égalité $\pi = \pi_{a'_1} \circ \dots \circ \pi_{a'_k}$.
5. Soit $A = \{a_1, \dots, a_n\}$ et soit $\pi : A^* \rightarrow B^*$ une projection élémentaire. Pour fixer les notations, on suppose que $\pi(a_n) = \varepsilon$ et que $\pi(a_i) = a_i$ pour $1 \leq i \leq n-1$. On suppose aussi que $B = \{a_1, \dots, a_{n-1}\}$. Soit L un langage rationnel sur A . On considère l'ordre lexicographique $<_{\text{lex}}$ sur A^* induit par l'ordre $a_1 < \dots < a_n$ des lettres. Soit $u = b_1 \dots b_m$ un mot sur B . Un mot w vérifie $\pi(w) = u$ si w est égal à $a_n^{k_0} b_1 a_n^{k_1} b_2 \dots b_m a_n^{k_m}$ où k_0, \dots, k_m sont des entiers. On remarque que la restriction de $<_{\text{lex}}$ à $\pi^{-1}(u)$ est bien fondée. Deux mots w et w' vérifient $\pi(w) = \pi(w') = u$ et $w <_{\text{lex}} w'$ si et seulement si les deux $m+1$ -uplets d'entiers (k_0, \dots, k_m) et (k'_0, \dots, k'_m) vérifient $(k_0, \dots, k_m) < (k'_0, \dots, k'_m)$ pour l'ordre lexicographique

sur les $m + 1$ -uplets d'entiers. Il s'ensuit que l'ordre lexicographique $<_{\text{lex}}$ retreint à $\pi^{-1}(u)$ est bien fondé.

Pour tout mot w de L , on note $f(w)$ le plus petit mot pour l'ordre $<_{\text{lex}}$ de $\pi^{-1}(\pi(w)) \cap L = \{w' \in L \mid \pi(w') = \pi(w)\}$. Le mot $f(w)$ appartient à L et vérifie $\pi(f(w)) = \pi(w)$. De plus si w et w' vérifient $\pi(w) = \pi(w')$, alors $f(w) = f(w')$. On définit finalement le langage K par $K = f(L)$. Par définition, la projection π est injective sur K et $\pi(K) = \pi(L)$. Il reste à montrer que le langage K est rationnel.

Soit $\mathcal{A} = (Q, A, E, \{i\}, F)$ un automate déterministe et complet acceptant le langage L . On construit un automate \mathcal{A}' qui accepte les mots de K . L'idée générale de cet automate est d'accepter un mot w s'il est accepté par \mathcal{A} si aucun mot w' tel que $\pi(w') = \pi(w)$ et $w' <_{\text{lex}} w$ n'est accepté par \mathcal{A} .

6. D'après les questions 2 et 3, il suffit de montrer que les projections ont la propriété de sélection. D'après les questions 2 et 4, il suffit encore de montrer que les projections élémentaires ont la propriété de sélection. Le résultat de la question 5 permet de conclure que tout morphisme a la propriété de sélection.

Exercice 1.9. 1. Montrer qu'un sous-monoïde M de A^* est libre (cf. définition 1.28) si et seulement si il vérifie $M^{-1}M \cap MM^{-1} = M$.

2. Soit M un sous-monoïde de A^* . On définit la suite de $(M_n)_{n \geq 0}$ de sous-monoïdes de A^* par $M_0 = M$ et $M_{n+1} = (M_n^{-1}M_n \cap M_n M_n^{-1})^*$. Montrer que l'enveloppe libre de M est $\hat{M} = \bigcup_{n \geq 0} M_n$.
3. Soit $M = X^*$ un sous-monoïde de A^* . On définit les deux suites de langages $(U_{n \geq 0})$ et $(V_{n \geq 0})$ par $U_0 = V_0 = \{\varepsilon\}$ et $U_{n+1} = U_n^{-1}X + X^{-1}U_n$ et $V_{n+1} = XV_n^{-1} + V_n X^{-1}$. Montrer que $(M^{-1}M \cap MM^{-1})^*$ est égal à $(U \cap V)^*$ où les langages U et V sont définis par $U = \bigcup_{n \geq 0} U_n$ et $V = \bigcup_{n \geq 0} V_n$.
4. Montrer que l'enveloppe libre d'un langage rationnel est encore un langage rationnel.

Solution.

1. Pour tout sous-monoïde M de A^* , l'inclusion $M \subseteq M^{-1}M \cap MM^{-1}$ est vérifiée. Supposons que M soit libre et soit u un mot appartenant à $M^{-1}M \cap MM^{-1}$. Il existe donc des mots v et w de M tels que wu et uv appartiennent à M . Les mots vwu et uvw appartiennent à M . Puisque M est libre, u appartient aussi à M . La réciproque est évidente.
2. On vérifie par récurrence que $M_n \subseteq M_{n+1}$ pour tout $n \geq 0$. Il s'ensuit que l'union \hat{M} est bien un sous-monoïde de A^* . Il faut ensuite montrer que c'est le plus petit sous-monoïde libre contenant M .

Soient u et v des mots tels que uv , vu et v appartiennent à \hat{M} . Puisque les sous-monoïdes M_n forment une suite croissante, il existe un entier n tel que uv , vu et v appartiennent à M_n . Le mot u appartient alors à M_{n+1} et donc à \hat{M} . Ceci prouve que le sous-monoïde \hat{M} est libre. Soit M' un sous-monoïde libre de A^* contenant M . On montre par récurrence sur n que chacun des sous-monoïdes M_n est inclus dans M' . On en déduit que \hat{M} est finalement inclus dans M' .

3. On vérifie que $U_1 = V_1 = X$. On montre d'abord par récurrence sur $n \geq 1$ qu'un mot u appartient à U_n s'il est suffixe d'un mot $x \in X$ et s'il existe des mots $v \in X^k$ et $w \in X^\ell$ avec $k + \ell = n - 1$ tels que $vu = wx$. Le résultat est immédiat pour $n = 1$ puisque $U_1 = X$. Soit u un mot de U_{n+1} . Si u appartient à $U_n^{-1}X$ il existe un mot $x \in X$ et un mot $u' \in U_n$ tel que $u'u = x$. Par hypothèse de récurrence, il existe des mots $x' \in X$, $v' \in X^k$ et $w' \in X^\ell$ tels que $v'u' = w'x'$. On a alors $v'u'u = v'x = w'x'u$ et on pose $v = w'x' \in X^{\ell+1}$ et $w = v' \in X^k$ pour avoir $vu = wx$. Si u appartient à $X^{-1}U_n$, il existe un mot $x \in X$ et un mot $u' \in U_n$ tel que $xu = u'$. Par hypothèse de récurrence, il existe des mots $x' \in X$, $v' \in X^k$ et $w' \in X^\ell$ tels que $v'u' = w'x'$. On a alors $v'u'u' = v'xu = w'x'$. On pose $v = v'x \in X^{k+1}$ et $w = w' \in X^\ell$ pour avoir $vu = wx'$.

La propriété ci-dessus montre que $U \subseteq M^{-1}M$. Par symétrie, on a $V \subseteq MM^{-1}$ et donc $(U \cap V)^* \subseteq (M^{-1}M \cap MM^{-1})^*$. Pour l'inclusion inverse, il suffit de montrer que $M^{-1}M \cap MM^{-1} \subseteq (U \cap V)^*$. On montre par récurrence sur $|u|$ que si $u \in M^{-1}M \cap MM^{-1}$, alors $u \in (U \cap V)^*$. Si u est vide, le résultat est évident. Soit u tel que $|u| \geq 1$ et $u \in M^{-1}M \cap MM^{-1}$. Puisque $u \in M^{-1}M$, il existe des mots $v \in X^k$ et $w = x_1 \cdots x_\ell \in X^\ell$ tels que $vu = w$. Soit j le plus petit entier tel que $|x_1 \cdots x_j| \geq |v|$ et soit u' le mot $v^{-1}x_1 \cdots x_j$. Le mot u' est un suffixe de x_j et il appartient donc à U . De plus, le mot u est égal à $u'x_{j+1} \cdots x_\ell$. Puisque u appartient à MM^{-1} , le mot u' appartient à $U \cap MM^{-1}$. Par symétrie, il est possible de trouver un mot $v' \in V$ et des mots x'_1, \dots, x'_m tels que $u = x'_1 \cdots x'_m v'$. Puisque $u \in M^{-1}M$, le mot v' appartient à $M^{-1}M \cap V$. Si $|u'| < |u|$ ou $|v'| < |u|$, on applique l'hypothèse pour conclure. Sinon, on a $u = u' = v'$ et le mot u appartient à $U \cap V$.

4. Soit le sous-monoïde $M = X^*$ où X est un langage rationnel. On suppose que X est reconnu par le morphisme $\mu : A^* \rightarrow N$ où N est un monoïde fini. On montre par récurrence que chaque sous-monoïde M_n est égal à X_n^* où X_n est encore reconnu par le morphisme μ . Le résultat est vrai pour $M_0 = M$ avec $X_0 = X$. D'après la question précédente, si M_n est égal à X_n^* , le sous-monoïde M_{n+1} est égal $(U \cap V)^*$ où les langages U et V sont définis en partant de X_n . Chacun des ensembles U_k et V_k est encore reconnu par le morphisme μ et dont U et V le sont également. On en déduit que $X_{n+1} = U \cap V$ est encore reconnu par μ . Comme le nombre de langages reconnus par un morphisme μ est fini, il existe un entier n tel que $M_{n+1} = M_n$. Le sous-monoïde \hat{M} est alors égal à $M_n = X_n^*$ et il est rationnel.

1.2 Langages algébriques

Exercice 1.10. On considère le langage $L = L_G(S)$ où la grammaire G contient les règles $S \rightarrow aSSb + c$. Montrer que tout langage rationnel inclus dans L est fini.

Solution. D'après le lemme de l'étoile, tout langage rationnel infini contient un langage de la forme uv^*w où u , v et w sont des mots et v est non vide. Il suffit donc de montrer que L ne contient aucun langage de cette forme.

On montre facilement par récurrence sur la longueur de la dérivation que tout mot w de L vérifie l'égalité $|w|_a = |w|_b$. On montre ensuite que tout mot w de L vérifiant $|w| \geq 2^n - 1$ pour un entier $n \geq 0$ possède un préfixe u tel que $|u|_a - |u|_b \geq n - 1$. Le seul mot obtenu par une dérivation de longueur 1 est le mot $w = c$, qui vérifie la

propriété. L'entier n est au plus égal à 1 et le préfixe $u = \varepsilon$ de w convient. Si w est obtenu par une dérivation de longueur au moins 2, il se factorise $w = aw_1w_2b$ où les mots w_1 et w_2 appartiennent à L . Un des deux mots w_1 et w_2 est de longueur au moins $2^{n-1} - 1$. Sinon le mot w est de longueur au plus $2 + 2(2^{n-1} - 2) = 2^n - 2$ contrairement à l'hypothèse $|w| \geq 2^n - 1$. Si w_1 vérifie $|w_1| \geq 2^{n-1} - 1$, il possède par hypothèse de récurrence un préfixe u_1 tel que $|u_1|_a - |u_1|_b \geq n - 2$. Le préfixe $u = au_1$ de w vérifie alors $|u|_a - |u|_b \geq n - 1$. Si w_2 vérifie $|w_2| \geq 2^{n-1} - 1$, il possède par hypothèse de récurrence un préfixe u_2 tel que $|u_2|_a - |u_2|_b \geq n - 2$. Le préfixe $u = aw_1u_2$ de w vérifie encore $|u|_a - |u|_b \geq n - 1$ puisque $|w_1|_a = |w_1|_b$.

Soit u, v et w des mots tels que v soit non vide et que uv^*w soit inclus dans L . Puisque $|uv^nw|_a = |uv^nw|_b$ pour tout entier n , on a $|v|_a = |v|_b$. Il s'ensuit que pour tout entier n et pour tout préfixe z de uv^nw , on a l'inégalité $|z|_a - |z|_b \leq |uvw|_a$. Comme cette majoration est indépendante de n et donc de la longueur de uv^nw , il y a une contradiction avec la propriété énoncée ci-dessus pour les mots de L .

Exercice 1.11. Soit A un alphabet et $\#$ un symbole n'appartenant pas à A . Soit $K \subseteq A^*$ un langage. Montrer que le langage $L = \{u\#v \mid u \in K \text{ et } |u| = |v|\}$ est algébrique si et seulement si L est rationnel.

Solution. On remarque que $L = K\#A^* \cap L'$ où L' est le langage $\{u\#v \mid |u| = |v|\}$. Le langage $K\#A^*$ est bien sûr rationnel. Le langage L' est algébrique puisqu'il est engendré par la grammaire $S \rightarrow \sum_{a,b \in A} aSb + \#$. Le langage L est algébrique d'après la proposition 2.53 p. 98. On peut aussi construire directement une grammaire ou un automate à pile pour L à partir d'un automate acceptant K .

On suppose maintenant que le langage L est algébrique. Soit $G = (A + \{\#\}, V, P)$ une grammaire réduite telle $L = L_G(S_0)$ pour une variable S_0 de G . On va montrer que le langage L est linéaire (cf. exercice 2.71).

Soit S une variable de G . Puisque tout mot de L a exactement une occurrence du symbole $\#$, tout mot engendré par S , c'est-à-dire de $L_G(S)$ a un plus une occurrence de $\#$. On va montrer que $L_G(S) \subseteq A^*$ ou $L_G(S) \subseteq A^*\#A^*$. Puisque G est réduite, il existe des mots u et v sur $A + \{\#\}$ tels que $S_0 \xrightarrow{*} uSv$. Si $uv \in A^*$, tout mot w de $L_G(S)$ doit avoir exactement une occurrence de $\#$ car uvw appartient à L . Si au contraire uv a déjà une occurrence, alors tout mot w de $L_G(S)$ n'a aucune occurrence de $\#$.

On montre maintenant que si $L_G(S) \subseteq A^*$, alors le langage $L_G(S)$ est fini. Soient u et v des mots tels que $S_0 \xrightarrow{*} uSv$. Par symétrie on peut supposer que $u = u_0\#u_1$ où $u_0, u_1 \in A^*$. Pour tout mot w de $L_G(S)$, on doit avoir $|u_0| = |u_1| + |w| + |v|$. Ceci impose que tous les mots de $L_G(S)$ ont même longueur et que $L_G(S)$ est fini.

On peut remplacer dans la grammaire G toute occurrence d'une variable S telle que $L_G(S) \subseteq A^*$ par les différents mots de $L_G(S)$. On obtient alors une grammaire G' telle que toute variable S de G' vérifie $L_G(S) \subseteq A^*\#A^*$. La grammaire G' est alors linéaire. En effet pour toute règle $S \rightarrow w$ de G' , le mot w peut au plus contenir une seule occurrence d'une variable. Sinon, il existe un mot de $L_G(S)$ ayant au moins deux occurrences du symbole $\#$.

On peut supposer que toute règle de la grammaire linéaire G' est de la forme $S \rightarrow uTv$ où les mots u et v vérifient $|u| \leq 1$ et $|v| \leq 1$. Si $|u| \geq 2$ ou $|v| \geq 2$, il suffit d'introduire

des variables intermédiaires pour décomposer la règle. On peut aussi supposer qu'il existe une unique règle $S_1 \rightarrow \#$ qui produit le symbole $\#$.

On définit alors l'automate $\mathcal{A} = (V, A, E, \{S_0\}, \{S_1\})$ dont l'ensemble E des transitions est donné par

$$E = \{S \xrightarrow{a} T \mid S \rightarrow aTb \text{ pour } a, b \in A + \{\varepsilon\}\}.$$

Il est facile de vérifier que l'automate \mathcal{A} accepte le langage K .

Exercice 1.12. Montrer que pour un langage rationnel K et un langage algébrique L ,

1. l'inclusion $K \subseteq L$ est indécidable.
2. l'inclusion $L \subseteq K$ est décidable.

Solution. Dans le cas où le langage K est fixé à l'ensemble A^* , l'inclusion $K \subseteq L$ est équivalente à l'égalité $L = A^*$ et elle est donc indécidable. Elle est *a fortiori* indécidable pour des langages K et L quelconques.

Puisque le langage $A^* \setminus K$ est rationnel, le langage $L \cap (A^* \setminus K)$ est algébrique. Il est de plus possible de construire une grammaire qui engendre ce langage à partir d'une grammaire pour L et d'un automate acceptant le langage $A^* \setminus K$. Il suffit alors de tester si cette grammaire n'engendre aucun mot pour savoir si $L \subseteq K$.

1.3 Complexité

Exercice 1.13. Soit φ une formule en forme normale conjonctive. Une affectation des variables est appelée une *affectation symétrique* si chaque clause de φ contient deux littéraux avec des valeurs différentes. Le problème SYM est de savoir si une formule en forme normale conjonctive avec au plus trois littéraux par clause possède une affectation symétrique.

1. Montrer que SYM est inclus dans SAT. On entend par là l'inclusion des ensembles d'instances positives.
2. Montrer que cette inclusion est stricte.
3. Montrer que pour la formule $\varphi = (x \vee y \vee z)$ est satisfiable si et seulement si la formule $\varphi' = (x \vee y \vee l) \wedge (\bar{l} \vee z \vee b)$ appartient à SYM.
4. Dédire de la question précédente que le problème SYM est NP-complet.
5. Le problème SET-SPLITTING consiste à savoir, étant donné un ensemble fini E et k sous-ensembles F_1, \dots, F_k , s'il est possible d'associer à chaque élément de E la couleur rouge ou bleu de telle sorte qu'aucun des sous-ensembles F_i n'a tous ses éléments de la même couleur. Montrer en utilisant les questions précédentes que ce problème est NP-complet.

Exercice 1.14. Le but de ce problème est de montrer qu'un langage accepté par une machine de Turing en espace $o(\log \log n)$ est nécessairement rationnel. Comme pour l'espace logarithmique, on considère une machine $M = (Q, \Sigma, \Gamma, E, q_0, F, \#)$ ayant une

bande d'entrée sur laquelle elle n'écrit pas et une bande de sortie dont la tête de lecture ne recule jamais. Sans perte de généralité, on suppose que M a une seule bande de travail. On suppose en outre que la machine M n'a pas de calcul infini. On note $s(n)$ l'espace maximal utilisé par M sur les entrées de taille n .

1. Montrer que si $s(n)$ est borné par une constante, alors M accepte nécessairement un langage rationnel.

On appelle *configuration locale* de la machine un triplet formé de l'état interne, du contenu de la bande de travail et de la position de la tête de lecture sur cette bande de travail. Les positions des têtes de lecture sur les bandes d'entrée et de sortie ne sont pas prises en compte.

2. Donner une borne $N(n)$ du nombre de configurations locales de M en fonction de $s(n)$.

On suppose maintenant que M n'accepte pas un langage rationnel. On dit qu'une étape de calcul $C \rightarrow C'$ de M franchit la frontière entre les cases k et $k+1$ si la tête de lecture de la bande d'entrée passe de la case k à la case $k+1$ ou de la case $k+1$ à la case k . Soit $\gamma = C_0 \rightarrow \dots \rightarrow C_n$ un calcul de M . Soit $i_0 < \dots < i_m$ la suite des indices où les étapes $C_i \rightarrow C_{i+1}$ franchissent la frontière k . On appelle *suite des franchissements* de γ en k , la suite de configurations locales C'_0, \dots, C'_m où $C'_j = C_{i_j+1}$. C'est la configuration locale de la machine juste après le franchissement qui est prise en considération.

3. Montrer que pour chaque entier k , il existe un mot x_k le plus court possible tel que l'espace utilisé par x_k est au moins k .

On note n_k la longueur de x_k et γ_k le calcul de M sur x_k .

4. Montrer que parmi les n_k suites de franchissements de γ_k , il y a en a au moins $n_k/2$ distinctes.
5. On note m_k la longueur maximale d'une suite de franchissements de γ_k . Montrer que

$$n_k \leq 4N(n_k)^{m_k}$$

où $N(n)$ est le nombre de configurations de M trouvé à la question 2.

6. Montrer que $m_k \leq 2N(n_k)$.
7. Montrer finalement que $s(n) = \Omega(\log \log n)$. Ceci signifie que $s(n) \geq K \log \log n$ pour une constante K et pour n assez grand.

Pour tout entier n , on note $(n)_2$ l'écriture en binaire de n . On a par exemple $(1)_2 = 1$, $(2)_2 = 10$, $(3)_2 = 11$ et $(4)_2 = 100$. Soit L le langage défini par

$$L = \{(1)_2 \# (2)_2 \# (3)_2 \# \dots \# (2^k - 1)_2 \mid k \geq 1\}.$$

8. Montrer que L n'est pas rationnel.
9. Montrer que L peut être accepté par une machine de Turing déterministe en espace $O(\log \log n)$.

Solution.

1. Si l'espace utilisé par la machine est borné, le nombre de configurations globales de la machine est fini. La machine est en fait un automate fini et le langage accepté est rationnel.
2. Soit k le cardinal de l'alphabet de bande de la machine. Le nombre de contenus de bande est donc au plus $k^{s(n)}$. Le nombre de positions de la tête lecture sur la bande de travail est au plus $s(n)$. Le nombre maximal de configurations locales est donc $|Q|s(n)k^{s(n)}$ où Q est l'ensemble d'états de la machine.

3. Comme on a supposé que le langage accepté n'est pas rationnel, l'espace utilisé par la machine n'est pas borné. Pour tout entier k , il existe un mot d'entrée x_k tel que l'espace utilisé par la machine sur x_k est au moins k . On prend pour x_k un tel mot le plus court possible.
4. Supposons d'abord que la configuration de γ_k avec un espace maximal a sa tête de lecture de la bande d'entrée dans la première moitié de l'entrée. On montre alors que les $n_k/2$ dernières suites de franchissements sont nécessairement distinctes. Supposons par l'absurde que les suites de franchissements en j et j' avec $n_k/2 \leq j < j'$ soient égales. Soit x'_k le mot obtenu en supprimant de x_k les lettres entre les positions j et j' . En recolant les parties de γ_k avant la position j et après la position j' (cf. figure 4.18 p. 227), on obtient un calcul γ'_k sur l'entrée x'_k . L'espace maximal de γ'_k est identique à celui de γ_k . Ceci contredit la définition de x_k puisque x'_k est plus court que x_k .
Si la configuration de γ_k avec un espace maximal a sa tête de lecture de la bande d'entrée dans la seconde moitié de l'entrée, on montre de la même façon que les $n_k/2$ premières suites de franchissements de γ_k sont distinctes.
5. D'après la question précédente, $n_k/2$ est inférieur au nombre de suites de franchissements distinctes qui est lui même inférieur au nombre total de suites de franchissements. On obtient donc l'inégalité

$$n_k/2 \leq \frac{N(n_k)^{m_k+1} - 1}{N(n_k) - 1} \leq 2N(n_k)^{m_k}$$

6. On a supposé que la machine s'arrête sur toute entrée. Elle ne peut donc pas repasser dans la même configuration locale avec la tête de lecture à la même position. On obtient donc la majoration $m_k \leq 2N(n_k)$. Le facteur 2 provient du sens de déplacement de la tête.
8. Supposons par l'absurde que L est rationnel et soit N l'entier fourni par le lemme de l'étoile. Pour k supérieur à N , le mot $(1)_2\#\cdots\#(2^k - 1)_2$ se factorise uvw où v est un facteur de $(2^k - 1)_2 = 1 \cdots 1$ et où uv^nw appartient à X pour chaque $n \geq 0$. Pour $n = 0$, on obtient une contradiction car le mot uw n'appartient pas à X .
9. Pour chaque $k \geq 1$, le mot $(1)_2\#\cdots\#(2^k - 1)_2$ est de longueur $k2^k - 1$. On montre qu'il existe une machine utilisant un espace $\log \log n$ si un mot d'entrée w de longueur n appartient à L . La machine commence par vérifier que le mot commence par $1\#$ et que le bloc sur $\{0, 1\}$ après le dernier $\#$ ne contient que des 1. Ensuite, elle vérifie pour chaque paire des blocs consécutifs qu'ils sont bien les écritures en binaire de deux entiers consécutifs. Elle les compare chiffre par chiffre. Pour cela, la machine mémorise sur sa bande le numéro de la position qu'elle est en train de comparer dans les deux blocs. Comme les blocs sont de longueur au plus k , l'écriture en binaire de cette position utilise un espace au plus $\log k$.