Corrigé de l'épreuve d'informatique
Concours MPI, ENS Ulm, Lyon et Cachan, session 2000

Julien CASSAIGNE[*], Jean MAIRESSE[†]

Thème du sujet: Algorithmes sur les graphes : détection de circuits de poids positif; transducteurs : propriétés de fonctionnalité et de séquentialité.

1 Algorithmique des graphes

1.1.

L'algorithme proposé consiste à parcourir le graphe, d'une part à partir du premier nœud distingué i et dans le sens des flèches, d'autre part à partir du second nœud distingué j et dans le sens inverse. Chaque nœud rencontré est marqué comme étant accessible à partir de i (resp. co-accessible à partir de j). Une fois ces deux parcours terminés, il suffit de sélectionner les nœuds qui portent les deux marques. On utilise un algorithme récursif.

```
ÉMONDER (N, E, i, j)
    pour tout u ∈ N faire
        accessible[u] ← faux
        co-accessible[u] ← faux
    ACCESSIBLE(i)
    CO-ACCESSIBLE(j)
    N' ← ∅
    pour tout u ∈ N faire
        si accessible[u] et co-accessible[u]
            alors N' ← N' ∪ {u}
    E' ← ∅
    pour tout e ∈ E faire
        si i(e) ∈ N' et f(e) ∈ N'
            alors E' ← E' ∪ {e}
    retourner (N', E')
```

[*]IML, CNRS, Campus de Luminy, Case 907, 13288 Marseille Cedex 9. E-mail: cassaigne@iml.univ-mrs.fr

[†]LIAFA, CNRS - Université Paris 7, Case 7014, 2 place Jussieu, 75251 Paris Cedex 5. E-mail: mairesse@liafa.jussieu.fr

Accessible (u)
(1) **si** accessible[u]
  alors sortir
(2) accessible[u] $\leftarrow$ vrai
  pour tout $e \in E$ **tel que** $i(e) = u$ **faire**
(3) Accessible($f(e)$)

Co-Accessible (u)
  si co-accessible[u]
  alors sortir
  co-accessible[u] $\leftarrow$ vrai
  pour tout $e \in E$ **tel que** $f(e) = u$ **faire**
   Co-Accessible($i(e)$)

On suppose que la structure de données du graphe permet d'accéder à la liste des arcs ayant pour nœud initial ou final un nœud donné u. Si ce n'est pas le cas, un prétraitement en $O(\mathrm{Card}(E))$ est nécessaire. Ensuite, chaque parcours demande $O(\mathrm{Card}(E))$ opérations élémentaires : en effet, la boucle de la fonction Accessible est exécutée au plus une fois pour chaque nœud accessible, et cette boucle comprend un appel récursif de la fonction Accessible pour chaque arc partant de u. Ainsi, la ligne (3) est exécutée au plus une fois pour chaque arc, la ligne (1) est exécutée au plus $1 + \mathrm{Card}(E)$ fois, et la ligne (2) au plus $\mathrm{Card}(N')$ fois. En tenant compte des initialisations, des deux parcours, et de la sélection des nœuds et des arcs à conserver, on trouve que la complexité est $O(\mathrm{Card}(E) + \mathrm{Card}(N))$. Si le graphe ne contient pas de nœuds isolés, on a donc une complexité $O(\mathrm{Card}(E))$. Pour avoir une complexité $O(\mathrm{Card}(E))$ dans tous les cas, on peut présélectionner j et les nœuds dont partent au moins un arc.

1.2.a.

Supposons qu'il existe un circuit de poids strictement positif. On peut alors choisir un circuit de poids strictement positif et de longueur minimale, $c = (e_1, \ldots, e_k)$. Supposons que $k > n$. Alors, parmi les k nœuds $i(e_1), \ldots, i(e_k)$, au moins deux sont égaux, disons $i(e_a) = i(e_b)$ avec $1 \leq a < b \leq k$. Mais alors $c_1 = (e_a, \ldots, e_{b-1})$ et $c_2 = (e_b, \ldots, e_k, e_1, \ldots, e_{a-1})$ sont deux circuits, de longueurs respectives $b - a < k$ et $k + a - b < k$, tels que $w(c_1) + w(c_2) = w(c) > 0$; l'un au moins de ces deux circuits a donc un poids strictement positif, ce qui contredit la minimalité de k.

On a montré que s'il existe un circuit de poids strictement positif, alors il existe un circuit de poids strictement positif et de longueur au plus n. La réciproque est évidente.

b.

On montre aisément, par récurrence sur k, que $U_{ij}^{(k)}$ est le poids maximal d'un chemin de longueur au plus k commençant en i et se terminant en j, ou $-\infty$ s'il n'existe pas de tel chemin. Donc $U_{ii}^{(k)}$ est le poids maximal d'un circuit de longueur au plus k passant par i. D'après a), il existe un circuit de poids strictement positif si et seulement s'il existe un circuit c de poids strictement positif et de longueur au plus n. Si i est un nœud de c, alors $U_{ii}^{(n)} > 0$; réciproquement, si $U_{ii}^{(k)} > 0$, alors il existe un circuit de poids strictement positif.

c.

Il suffit de calculer les matrices $U^{(k)}$, pour $k = 1, \ldots, n$, et de tester leurs coefficients diagonaux. Calculer un coefficient $U_{ij}^{(k)}$, connaissant la matrice $U^{(k-1)}$, demande $O(n)$ étapes élémentaires (calcul du maximum), donc calculer une matrice $U^{(k)}$ demande $O(n^3)$ étapes élémentaires et calculer les n matrices demande $O(n^4)$ étapes élémentaires.

1.3.

La vérification de l'associativité de l'opération $\otimes$ est immédiate. On traite la deuxième partie de la question. L'idée est d'utiliser une approche de type 'diviser pour régner'. Remarquons d'abord que pour déterminer s'il existe un circuit de poids strictement positif, il suffit de calculer $U^{(m)}$, pour n'importe quel $m \geq n$, et qu'on peut donc se dispenser de calculer toutes les matrices intermédiaires. L'associativité de l'opération $\otimes$ entraîne la relation $U^{(k+l)} = U^{(k)} \otimes U^{(l)}$ (que l'on peut montrer par récurrence sur l : $U^{(k+l+1)} = (U^{(k)} \otimes U^{(l)}) \otimes W = U^{(k)} \otimes (U^{(l)} \otimes W) = U^{(k)} \otimes U^{(l+1)}$). Soit k l'entier tel que $2^{k-1} < n \leq 2^k$. On calcule alors $U^{(2^k)}$ à l'aide de la récurrence $U^{(1)} = W$, $U^{(2^{i+1})} = U^{(2^i)} \otimes U^{(2^i)}$ pour $i = 0, \ldots, k-1$. Le nombre d'opérations élémentaires à chaque étape est $O(n^3)$, et comme i varie jusqu'à $k = \lceil \log_2(n) \rceil$, la complexité totale est $O(n^3 \log n)$.

Une autre façon de procéder est la suivante (la complexité se justifie de manière analogue). Soit $\langle n \rangle = w_k w_{k-1} \cdots w_0$ la représentation binaire de n (cf. question 2.3), où $k = \lfloor \log_2(n) \rfloor$. Pour $0 \leq i \leq k$, posons $n_i = \lfloor n/2^{k-i} \rfloor$. On a alors $n_0 = 1$, $n_{i+1} = 2n_i + w_{k-i}$ et $n_k = n$. Cela permet de calculer successivement les matrices $U^{(n_i)}$, en partant de $U^{(n_0)} = W$ et en utilisant la relation

$$U^{(n_{i+1})} = \begin{cases} U^{(n_i)} \otimes U^{(n_i)} \text{ si } w_{k-i} = 0 \\ U^{(n_i)} \otimes U^{(n_i)} \otimes W \text{ si } w_{k-i} = 1 \end{cases} .$$

1.4.

Montrons par récurrence sur k que, pour tout $i, j \in N$ et $k \in N \cup \{0\}$, les deux propriétés suivantes sont vérifiées : (i) si $V_{ij}^{(k)} \neq -\infty$, il existe un chemin entre i et j de poids $V_{ij}^{(k)}$; (ii) si c est un chemin entre i et j dont les nœuds *intermédiaires* sont tous distincts et dans $\{1, \ldots, k\}$, alors $w(c) \leq V_{ij}^{(k)}$.

Pour $k = 0$, $V_{ij}^{(0)} = W_{ij}$ est le poids maximal d'un chemin entre i et j de longueur 0 ou 1 (donc sans nœuds intermédiaires), donc les propriétés (i) et (ii) sont satisfaites.

Supposons les propriétés (i) et (ii) satisfaites par $V_{ij}^{(k-1)}$ pour tous $i, j \in N$ et pour un $k \in N$ donné. La matrice $V^{(k)}$ est définie par $V_{ij}^{(k)} = \max(V_{ij}^{(k-1)}, V_{ik}^{(k-1)} + V_{kj}^{(k-1)})$. Si $V_{ij}^{(k)} = V_{ij}^{(k-1)} \neq -\infty$, alors par hypothèse de récurrence il existe un chemin entre i et j de poids $V_{ij}^{(k)}$; si $V_{ij}^{(k)} = V_{ik}^{(k-1)} + V_{kj}^{(k-1)} \neq -\infty$, alors par hypothèse de récurrence il existe un chemin entre i et k de poids $V_{ik}^{(k-1)}$ et un chemin entre k et j de poids $V_{kj}^{(k-1)}$, donc en les concaténant on obtient un chemin entre i et j de poids $V_{ij}^{(k)}$: la propriété (i) est satisfaite.

Soit c un chemin entre i et j, dont les nœuds intermédiaires sont tous distincts et dans $\{1, \ldots, k\}$. Si c ne passe pas par k, alors par l'hypothèse de récurrence $w(c) \leq V_{ij}^{(k-1)} \leq V_{ij}^{(k)}$. Si c passe par k, on peut le décomposer en un chemin c_1 de i à k et un chemin c_2 de k à j, et l'hypothèse de récurrence s'applique à c_1 et c_2, de sorte qu'on a $w(c) = w(c_1) + w(c_2) \leq V_{ik}^{(k-1)} + V_{kj}^{(k-1)} \leq V_{ij}^{(k)}$.

S'il existe un circuit de poids strictement positif, alors il existe un tel circuit passant au plus une fois par chaque nœud, et donc $V_{ii}^{(n)} > 0$ pour un nœud i situé sur ce circuit. Réciproquement, si $V_{ii}^{(k)} > 0$ alors il existe un circuit c passant par i tel que $w(c) = V_{ii}^{(k)} > 0$.

On obtient l'algorithme voulu en calculant les matrices $V^{(k)}$ pour $1 \leq k \leq n$, puis en testant leurs coefficients diagonaux (ou seulement ceux de $V^{(n)}$). Le calcul de chaque matrice nécessite $O(n^2)$ opérations élémentaires, donc le calcul des n matrices nécessite $O(n^3)$ opérations élémentaires.

Remarque. De nombreux candidats proposent (et "démontrent") l'énoncé erroné suivant : $V_{ij}^{(k)}$ est le poids maximal d'un chemin de i à j dont tous les nœuds intermédiaires sont dans $\{1, \ldots, k\}$ (variante : sont distincts et dans $\{1, \ldots, k\}$). Cet énoncé n'est correct que dans le cas où il n'existe pas de circuit de poids strictement positif, ce qui est justement la propriété que l'on cherche à tester.

1.5.

Considérons l'état du tableau *valeur* à la sortie de l'algorithme. Si l'algorithme retourne 'non', cela signifie que pour tout arc e, $valeur[f(e)] \geq valeur[i(e)] + \varphi(e)$. En particulier, si $c = (e_1, \ldots, e_k)$ est un circuit, on a $valeur[f(e_i)] \geq valeur[i(e_i)] + \varphi(e_i)$ pour les arcs de c, ce qui en sommant sur i donne :

$$\sum_{i=1}^{k} valeur[f(e_i)] \geq \sum_{i=1}^{k} valeur[i(e_i)] + w(c) \ ,$$

mais comme $f(e_i) = i(e_{i+1})$ et $f(e_k) = i(e_1)$, les deux sommes sont égales et il reste $w(c) \leq 0$. On a montré que si l'algorithme retourne 'non', alors tous les circuits sont de poids inférieur ou égal à 0.

Réciproquement, supposons que tous les circuits sont de poids inférieur ou égal à 0. Nous allons montrer qu'à la sortie de l'algorithme, $valeur[u]$ est égal au poids maximal d'un chemin de longueur quelconque se terminant en u. Soit $c = (e_1, \ldots, e_k)$ un chemin tel que $f(e_k) = u$. Si sa longueur est plus grande que n, il passe au moins deux fois par le même nœud, disons $i(e_a) = i(e_b)$ avec $a < b$. Soit alors $c' = (e_1, \ldots, e_{a-1}, e_b, \ldots, e_k)$ et $c_1 = (e_a, \ldots, e_{b-1})$: $w(c') = w(c) - w(c_1) \geq w(c)$ puisque le poids du circuit c_1 est inférieur où égal à 0. Le poids maximal d'un chemin se terminant en u est donc atteint pour un chemin de longueur au plus n.

Soit $c = (e_1, \ldots, e_k)$, $k \leq n$, un chemin tel que $f(e_k) = u$. Montrons tout d'abord qu'après l'itération i ($1 \leq i \leq k$) de la boucle principale de l'algorithme, on a $valeur[f(e_i)] \geq w((e_1, \ldots, e_i))$. Notons que la valeur de chaque élément du tableau *valeur* ne peut que croître au cours du temps, et en particulier qu'elle est positive puisqu'elle est initialisée à 0. Au cours de l'itération 1, l'algorithme exécute (entre autres) $\textsc{Actualisation}(i(e_1), f(e_1), \varphi)$. Après cette actualisation, on a $valeur[f(e_1)] \geq valeur[i(e_1)] + \varphi(e_1) \geq \varphi(e_1) = w((e_1))$. Supposons que $valeur[f(e_{i-1})] \geq w((e_1, \ldots, e_{i-1}))$ après l'itération $i-1$; au cours de l'itération i, l'algorithme exécute (entre autres) $\textsc{Actualisation}(i(e_i), f(e_i), \varphi)$. Après cette actualisation, on a $valeur[f(e_i)] \geq valeur[i(e_i)] + \varphi(e_i) = valeur[f(e_{i-1})] + \varphi(e_i)$. Même si $valeur[f(e_{i-1})]$

a été augmenté entre temps, on a toujours $valeur[f(e_{i-1})] \geq w((e_1, \ldots, e_{i-1}))$, et donc $valeur[f(e_i)] \geq w((e_1, \ldots, e_{i-1})) + \varphi(e_i) = w((e_1, \ldots, e_i))$. La propriété recherchée s'en déduit par récurrence. À la fin de l'exécution de l'algorithme (donc après n itérations de la boucle principale), on a bien $valeur[u] \geq w(c)$.

Inversement, à tout moment il existe un chemin se terminant en u dont le poids est $valeur[u]$. Au départ, il suffit de considérer le chemin de longueur nulle en u. Supposons cette propriété vérifiée avant un appel de ACTUALISATION(u, v, φ). Si le test effectué à l'intérieur de cette fonction est négatif, rien ne change; si ce test est positif, $valeur[v]$ est modifié, et sa nouvelle valeur satisfait $valeur[v] = valeur[u] + \varphi(u, v)$. Il existe un chemin se terminant en u dont le poids est $valeur[u]$; en prolongeant ce chemin par (u, v), on obtient un chemin se terminant en v dont le poids est $valeur[v]$, donc la propriété considérée reste vraie à la sortie de la fonction. Par conséquent, à la fin de l'exécution de l'algorithme $valeur[u]$ est égal au poids d'un chemin se terminant en u, et donc au poids maximal puisque $valeur[u] \geq w(c)$ pour tout chemin c se terminant en u. On a donc, pour tout $(u, v) \in E$, $valeur[v] \geq valeur[u] + \varphi(u, v)$, puisque la quantité de droite est égale au poids d'un chemin particulier se terminant en v. L'algorithme retourne donc 'non'.

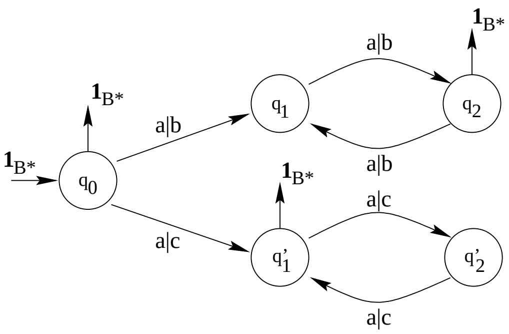
Il reste à déterminer la complexité de l'algorithme. La boucle principale donne lieu à n itérations, et chacune comprend m appels à ACTUALISATION, qui demandent chacun un nombre borné d'opérations élémentaires, soit en tout $O(nm)$ opérations élémentaires. L'initialisation demande $O(n)$ opérations élémentaires, et le test final $O(m)$ opérations élémentaires, donc l'ensemble de l'algorithme demande $O(nm)$ opérations élémentaires.

Remarque. Pour de nombreux candidats, *valeur*$[u]$ après i itérations de la boucle principale est égal au poids maximal d'un chemin de longueur i se terminant en u. Ceci est incorrect (même pour $i = 1$), *valeur*$[u]$ dépendant de l'ordre de passage sur les arcs dans la boucle (*valeur*$[u]$ est bien supérieur ou égal au poids maximal d'un chemin de longueur i, mais l'inégalité peut être stricte).

2 Transducteurs

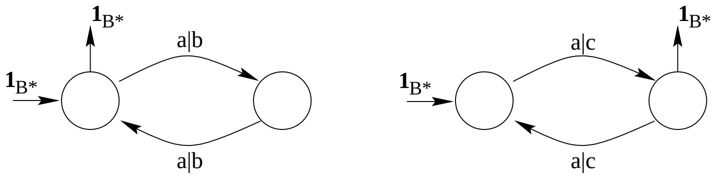
2.1.

Le transducteur suivant convient : $\mathcal{T} = (Q, A, B, T, I, F)$, avec $Q = \{q_0, q_1, q_2, q'_1, q'_2\}$, $\mathrm{dom}(I) = \{q_0\}$, $I(q_0) = \mathbf{1}_{B^*}$, $\mathrm{dom}(F) = \{q_2, q'_1\}$, $F(q_2) = F(q'_1) = \mathbf{1}_{B^*}$, $T(q_0, a, q_1) = T(q_1, a, q_2) = T(q_2, a, q_1) = b$, $T(q_0, a, q'_1) = T(q'_1, a, q'_2) = T(q'_2, a, q'_1) = c$, et T non défini pour les autres triplets.



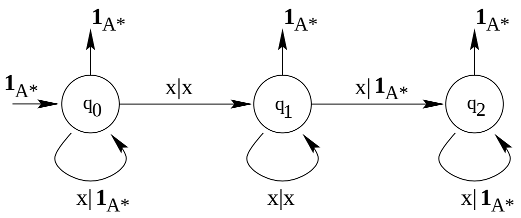
En effet, les seuls chemins réussis sont : le chemin de longueur nulle $q_0 \xrightarrow{\mathbf{1}_{A^*}|\mathbf{1}_{B^*}} q_0$, les chemins de longueur paire $q_0 \xrightarrow{a^{2n}|b^{2n}} q_2$ $(n \geq 1)$ et les chemins de longueur impaire $q_0 \xrightarrow{a^{2n+1}|c^{2n+1}} q_1'$ $(n \geq 0)$.

Il est également possible de proposer un transducteur non-connexe répondant à la question, comme ci-dessous.



2.2.

Le transducteur suivant convient : $\mathcal{T} = (Q, A, A, T, I, F)$, avec $Q = \{q_0, q_1, q_2\}$, $\mathrm{dom}(I) = \{q_0\}$, $I(q_0) = \mathbf{1}_{A^*}$, $\mathrm{dom}(F) = Q$, $F(q_0) = F(q_1) = F(q_2) = \mathbf{1}_{A^*}$, et pour tout $x \in A$, $T(q_0, x, q_0) = \mathbf{1}_{A^*}$, $T(q_0, x, q_1) = T(q_1, x, q_1) = x$, $T(q_1, x, q_2) = T(q_2, x, q_2) = \mathbf{1}_{A^*}$, et T non défini pour les autres triplets.



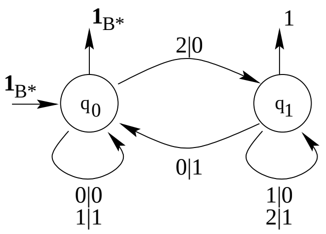
Un chemin réussi dont l'étiquette d'entrée est un mot w de longueur n est de l'une des formes suivantes :
- chemin passant $n+1$ fois par q_0 : $q_0 \xrightarrow{w|\mathbf{1}_{A^*}} q_0$;
- chemin passant i fois par q_0, puis $n-i+1$ fois par q_1, avec $1 \leq i \leq n$: $q_0 \xrightarrow{w|v} q_1$, où $v = w_i w_{i+1} \dots w_n$ est le suffixe de longueur $n-i+1$ de w;
- chemin passant i fois par q_0, puis $j-i+1$ fois par q_1, et enfin $n-j$ fois par q_2, avec $1 \leq i \leq j < n$: $q_0 \xrightarrow{w|v} q_2$, où $v = w_i w_{i+1} \dots w_j$ est le facteur de w de longueur $j-i+1$ commençant à la position i.

On voit ainsi que $f_{\mathcal{T}} = f$.

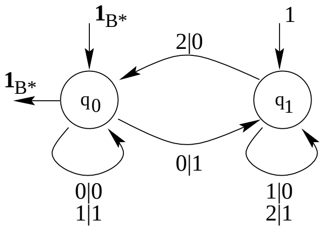
2.3.

L'idée de la construction est d'utiliser l'algorithme usuel de l'addition avec propagation de retenue, en commençant par le chiffre le moins significatif (c'est pour cela qu'on considère les mots miroir de w et de $\langle n+m \rangle$, puisque le chiffre le moins significatif de w ou de $\langle n+m \rangle$ est le dernier). Cet algorithme se traduit en un transducteur à deux états, $\mathcal{T}_1 = (Q, A, B, T, I, F)$, avec $Q = \{q_0, q_1\}$, $A = \{0, 1, 2\}$ et $B = \{0, 1\}$, dans lequel l'état q_0 indique l'absence de retenue et l'état q_1 indique la présence de retenue. L'unique état initial est q_0 (au départ, il n'y a pas de retenue) : $\mathrm{dom}(I) = \{q_0\}$, $I(q_0) = \mathbf{1}_{B^*}$. Les deux états sont terminaux, mais s'il reste une retenue à la fin de la lecture, il convient d'écrire un chiffre 1 supplémentaire : $\mathrm{dom}(F) = Q$, $F(q_0) = \mathbf{1}_{A^*}$, $F(q_1) = 1$. Enfin, les transitions sont données par $T(q_i, a, q_j) = b$

pour tout quadruplet $(a, i, j, b) \in A \times B^3$ tel que $a + i = 2j + b$: la somme a, augmentée de la retenue i, donne le chiffre binaire b et la nouvelle retenue j.



Pour obtenir un transducteur agissant sur w et non sur son mot miroir, il suffit d'inverser le sens des flèches, c'est-à-dire de considérer le transducteur $\mathcal{T}_2 = (Q, A, B, \tilde{T}, \tilde{I}, \tilde{F})$ où $\tilde{I} = F$, $\tilde{F} = I$, et $\tilde{T}(q_j, a, q_i) = T(q_i, a, q_j)$.



En effet, si on a $q_0 \xrightarrow{u|v} q_1$ dans le transducteur $\mathcal{T}_1$, alors on en déduit que l'on a $q_1 \xrightarrow{\tilde{u}|\tilde{v}} q_0$ dans le transducteur $\mathcal{T}_2$, où $\tilde{u}$ et $\tilde{v}$ sont les mots miroirs de u et v.

Remarque. Chercher à obtenir directement le transducteur $\mathcal{T}_2$, sans penser à utiliser $\mathcal{T}_1$, paraît difficile. Certains candidats s'y essaient sans succès.

2.4.

Montrons par récurrence que pour tout $n \in \mathbb{N}$, les chemins de longueur n se terminant en 1 sont de la forme : $1 \xrightarrow{a^n|b^{2n}} 1$, $2 \xrightarrow{a^n|b^{2n+1}} 1$, $3 \xrightarrow{a^n|b^{2n-1}} 1$ ou $4 \xrightarrow{a^n|b^{2n}} 1$.

Si $n = 0$, il n'y a que le chemin $1 \xrightarrow{\mathbf{1}_{A^*}|\mathbf{1}_{B^*}} 1$.

Supposons la proposition vraie pour n. Soit $q_0 \xrightarrow{a|v} q_1 \longrightarrow \cdots \longrightarrow q_{n+1} = 1$ un chemin de longueur $n+1$ se terminant en 1. On discute en fonction de (q_0, q_1), en appliquant l'hypothèse de récurrence à $q_1 \xrightarrow{a^n|w} 1$. Par exemple, si $(q_0, q_1) = (1, 2)$, alors $v = b$ et $w = b^{2n+1}$ et on trouve $vw = b^{2(n+1)}$. Dans chacun des cas, le chemin $q_0 \xrightarrow{a^{n+1}|vw} 1$ est de l'une des formes ci-dessus.

Les chemins réussis sont ceux qui commencent en 1 ou en 4 et qui se terminent en 1; ils ont donc comme étiquette $a^n|b^{2n}$. Le transducteur $\mathcal{T}$ est donc fonctionnel et on a, pour tout n tel que $a^n \in \mathrm{dom}(f_\mathcal{T})$, $f_\mathcal{T}(a^n) = b^{2n}$.

Il reste à déterminer $\mathrm{dom}(f_\mathcal{T})$. Remarquons que $\mathcal{T}$ contient un circuit de longueur 3 et un circuit de longueur 4 allant de 1 en 1. En combinant ces deux circuits, on peut obtenir un chemin réussi de longueur $3\alpha + 4\beta$ pour tous α, β dans $\mathbb{N}$; or $3\mathbb{N} + 4\mathbb{N} = \mathbb{N} \setminus \{1, 2, 5\}$ (preuve : si $n \equiv 0 \pmod 3$, $n = 3\alpha$; si $n \equiv 1 \pmod 3$ et $n \neq 1$, $n = 3\alpha + 4$; si $n \equiv 2 \pmod 3$ et $n \notin \{2, 5\}$, $n = 3\alpha + 8$).

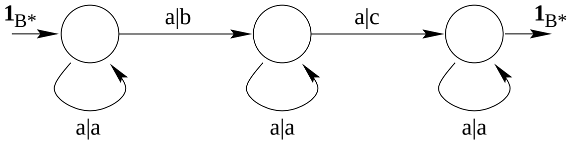
On a aussi des chemins $4 \xrightarrow{a|b^2} 1$ et $4 \xrightarrow{a^5|b^{10}} 1$, donc $\{a, a^5\} \subset \mathrm{dom}(f_{\mathcal{T}})$. En revanche on vérifie facilement qu'aucun chemin de longueur 2 n'est réussi : $\mathrm{dom}(f_{\mathcal{T}}) = \{a^n \ : \ n \in \mathbb{N} \setminus \{2\}\}$.

Une autre méthode de preuve consiste à utiliser la construction de la question 3.8.

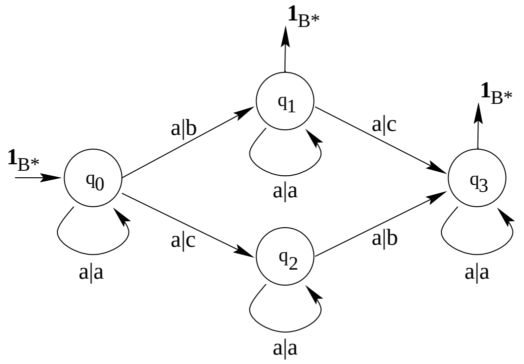
Remarque. Un nombre non négligeable de candidats ne réalisent pas que l'on peut combiner les circuits de longueur 3 et 4 et obtiennent $\mathrm{dom}(f_{\mathcal{T}}) = \{n \ : \ n = 0, 1\ [3] \text{ ou } n = 0, 1\ [4]\}$.

2.5.

L'idée est d'exploiter le fait que $C_n^2 = n(n-1)/2$. Ainsi, le transducteur suivant remplace deux quelconques des lettres du mot d'entrée a^n, $n \geq 2$, par un b puis un c, de sorte que $\mathrm{Card}(f_{\mathcal{T}}(a^n)) = n(n-1)/2$.



En modifiant légèrement ce transducteur, on peut obtenir l'égalité recherchée. Soit $\mathcal{T} = (Q, A, B, T, I, F)$ avec $A = \{a\}$, $B = \{a, b, c\}$, $Q = \{q_0, q_1, q_2, q_3\}$, $\mathrm{dom}(I) = \{q_0\}$, $\mathrm{dom}(F) = \{q_1, q_3\}$, $I(q_0) = F(q_1) = F(q_3) = \mathbf{1}_{B^*}$, $T(q_0, a, q_0) = T(q_1, a, q_1) = T(q_2, a, q_2) = T(q_3, a, q_3) = a$, $T(q_0, a, q_1) = T(q_2, a, q_3) = b$, $T(q_0, a, q_2) = T(q_1, a, q_3) = c$, T non défini pour les autres triplets.



On vérifie que l'on a, pour $n \geq 2$,

$$\begin{array}{rcl} f_{\mathcal{T}}(a^n) & = & \{a^i b a^j \ : \ i + j = n - 1\} \\ & & \cup \{a^i b a^j c a^k \ : \ i + j + k = n - 2\} \\ & & \cup \{a^i c a^j b a^k \ : \ i + j + k = n - 2\} \ . \end{array}$$

Ces trois ensembles sont disjoints, le premier a n éléments et chacun des deux autres a $n(n-1)/2$ éléments, donc au total $\mathrm{Card}(f_{\mathcal{T}}(a^n)) = n^2$. D'autre part, on vérifie que pour $n = 0, 1$, on a également $\mathrm{Card}(f_{\mathcal{T}}(a^n)) = n^2$.

En remplaçant c par bba dans l'exemple ci-dessus, on obtient le résultat voulu avec un alphabet de sortie à deux lettres. Si l'alphabet de sortie B ne contient qu'une lettre, cela n'est plus possible. En effet un mot de B^* est alors entièrement déterminé par sa longueur. Définissons

$$M = \max_{(p,a,q) \in \mathrm{dom}(T)} |T(p, a, q)|, \ M_I = \max_{q \in \mathrm{dom}(I)} |I(q)|, \ M_F = \max_{q \in \mathrm{dom}(F)} |F(q)|$$

(c'est le même M que celui défini à la question 3.6). On a alors

$$\forall v \in f_{\mathcal{T}}(u), |v| \leq M_I + M|u| + M_F$$

donc

$$\mathrm{Card}(f_{\mathcal{T}}(u)) \leq M_I + M|u| + M_F + 1 < |u|^2$$

pour $|u|$ suffisamment grand.

3 Transducteurs séquentiels

3.1

Une distance doit vérifier quatre propriétés : positivité, symétrie, séparation, inégalité triangulaire.

Positivité, symétrie : évidentes au vu de la définition.

Séparation : si $d(u, v) = 0$, alors $|u| + |v| = 2|u \wedge v|$; or par définition $u \wedge v$ est un préfixe de u et de v, donc $|u \wedge v| \leq \min(|u|, |v|) \leq (|u| + |v|)/2$. Il ne peut y avoir égalité que si $|u \wedge v| = |u| = |v|$, ce qui entraîne $u = u \wedge v = v$. Réciproquement, $d(u, u) = 0$ pour tout u.

Inégalité triangulaire : évaluons $\Delta = d(u, v) + d(v, w) - d(u, w)$.

$$\begin{aligned}
\Delta &= |u| + |v| - 2|u \wedge v| + |v| + |w| - 2|v \wedge w| - |u| - |w| + 2|u \wedge w| \\
&= 2(|v| - |u \wedge v| - |v \wedge w| + |u \wedge w|)
\end{aligned}$$

Comme $u \wedge v$ et $v \wedge w$ sont tous deux préfixes de v, on a $|v| \geq \max(|u \wedge v|, |v \wedge w|)$. D'autre part, $u \wedge v$ et $v \wedge w$ étant préfixes d'un même mot, l'un est préfixe de l'autre; par symétrie, supposons que le plus court des deux est $u \wedge v$. Alors $u \wedge v$ est un préfixe de $v \wedge w$ donc de w. Comme $u \wedge v$ est également préfixe de u, il est préfixe de $u \wedge w$, plus grand préfixe commun de u et w. On en déduit que l'on a dans le cas général $|u \wedge w| \geq \min(|u \wedge v|, |v \wedge w|)$. On conclut que $\Delta \geq 0$.

On considère la fonction de la question 2.1. On prend $u = a^{2n}$ et $v = a^{2n+1}$. On a $d(u, v) = 1$, et pourtant $d(f(u), f(v)) = d(b^{2n}, c^{2n+1}) = 4n+1$ n'est pas borné, donc f n'est pas à variation bornée.

3.2.

Quand $\mathrm{Card}(B) = 1$, la propriété de jumelage se réduit à : pour toute paire de chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$, $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$, avec $p, q \in Q$, $i, j \in \mathrm{dom}(I)$, $u_1 \in A^*$, $u_2 \in A^+$, $v_1, v_2, w_1, w_2 \in B^*$, on a $|v_2| = |w_2|$ (car B^* est commutatif et $xw_2 = v_2x$ est équivalent à $xw_2 = xv_2$, soit $w_2 = v_2$). Il suffit donc de comparer des longueurs, et cela peut se faire à l'aide de graphes valués.

Soit G_1 le graphe orienté dont les nœuds sont $Q \times Q$, avec un arc $(p, q) \longrightarrow (p', q')$ si et seulement s'il existe $x \in A$ et $v, w \in B^*$ tels que $p \xrightarrow{x|v} p'$ et $q \xrightarrow{x|w} q'$ sont des transitions de $\mathcal{T}$. Par construction, G_1 contient au plus un arc entre deux nœuds donnés.

Soit G_2 le graphe obtenu en ajoutant à G_1 deux nœuds i_0 et j_0 et des arcs $i_0 \longrightarrow (i, j)$ pour tout $(i, j) \in \mathrm{dom}(I) \times \mathrm{dom}(I)$ et $(p, q) \longrightarrow j_0$ pour tout $(p, q) \in Q \times Q$ (noter la dissymétrie dans les définitions de i_0 et j_0).

Soit G_3 le graphe émondé obtenu en appliquant à G_2 l'algorithme de la question 1.1 avec comme couple de nœuds distingués (i_0, j_0) (on peut simplifier l'algorithme en remarquant que tous les nœuds sont co-accessibles à partir de j_0).

À toute paire de chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$ et $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$ commençant par des états initiaux dans $\mathcal{T}$ correspond un chemin $i_0 \longrightarrow (i,j) \longrightarrow (p,q) \longrightarrow (p,q)$ dans G_3; en effet, ce chemin est par construction dans G_2, et il est conservé lors de l'émondage puisque tous les nœuds utilisés sont accessibles depuis i_0 (et coaccessibles depuis j_0, puisque tous les nœuds le sont). Réciproquement, tout circuit (de longueur éventuellement nulle) $(p,q) \longrightarrow (p,q)$ dans G_3 provient, puisque (p,q) doit être accessible, d'au moins un chemin $i_0 \longrightarrow (i,j) \longrightarrow (p,q) \longrightarrow (p,q)$ dans G_2, et donc d'au moins une paire de chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$ et $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$ dans $\mathcal{T}$.

On munit maintenant G_3 d'une valuation φ définie par

$$\varphi((p,q) \longrightarrow (p',q')) = \max\{|v| - |w| \,:\, v, w \in B^*, \exists x \in A, p \xrightarrow{x|v} p' \text{ et } q \xrightarrow{x|w} q'\} \ ,$$

de sorte que, si $p \xrightarrow{u|v} p'$ et $q \xrightarrow{u|w} q'$ dans $\mathcal{T}$, alors $|v| - |w|$ est inférieur ou égal au poids du chemin de (p,q) à (p',q') associé dans G_3.

Les algorithmes de la partie 1 (par exemple celui de la question 1.5) permettent de tester si G_3 valué par φ contient un circuit de poids strictement positif. Si la réponse est 'oui', alors $\mathcal{T}$ ne vérifie pas la propriété de jumelage, car on a trouvé deux chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$ et $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$ avec $|v_2| - |w_2| > 0$. Si la réponse est 'non', alors tout circuit dans G_3 a un poids négatif ou nul, donc pour toute paire de chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$ et $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$ dans $\mathcal{T}$ on a $|v_2| - |w_2| \leq w((p,q) \longrightarrow (p,q)) \leq 0$, et symétriquement on a aussi $|w_2| - |v_2| \leq w((q,p) \longrightarrow (q,p)) \leq 0$, donc $|v_2| = |w_2|$: $\mathcal{T}$ vérifie la propriété de jumelage.

La construction du graphe G_1 demande $O(n^4)$ étapes élémentaires, alors que celle de G_2 à partir de G_1 en nécessite $O(n^2)$. En utilisant l'algorithme de la question 1.1, on obtient G_3 en partant de G_2 en $O(n^4)$ étapes élémentaires (en effet G_2 possède $n^2 + 2$ nœuds et donc $O(n^4)$ arcs). Enfin, en utilisant l'algorithme de la question 1.5, la détection d'un circuit de poids strictement positif dans G_3 demande $O(n^2 \times n^4) = O(n^6)$ étapes élémentaires.

Remarque. Pour obtenir l'algorithme voulu, il suffit en fait de tester l'existence d'un circuit de poids différent de 0 dans G_3. Ceci peut être fait grâce à un algorithme demandant uniquement $O(n^4)$ étapes élémentaires. En effet, il est moins coûteux de tester l'existence d'un circuit de poids différent de 0 que de tester l'existence d'un circuit de poids strictement supérieur à 0!

D'autre part, on peut également tester à l'aide du graphe G_3, et en $O(n^6)$ étapes élémentaires, la fonctionnalité de $f_{\mathcal{T}}$ (cette construction était mentionnée dans les remarques à la fin de l'énoncé). Appliquons l'algorithme de la question 1.4 au graphe G_3. S'il existe un circuit de poids strictement positif alors $f_{\mathcal{T}}$ n'est pas fonctionnel. Dans le cas contraire, $V^{(n^2)}_{(i,j)(k,l)}$ (notations de la question 1.4) correspond au poids maximal d'un chemin de (i,j) à (k,l) dans G_3. On obtient alors que $f_{\mathcal{T}}$ est fonctionnel si et seulement si : $\forall (i,j) \in \text{dom}(I) \times \text{dom}(I), \forall (k,l) \in \text{dom}(F) \times \text{dom}(F),\ V^{(n^2)}_{(i,j)(k,l)} + |F(k)| - |F(l)| = 0$.

3.3.
Soit $\mathcal{T}$ un transducteur séquentiel réalisant f, M la longueur maximale de l'étiquette de sortie d'une flèche de $\mathcal{T}$, et $M_F = \max_{q \in \mathrm{dom}(F)} |F(q)|$.
Soient u et v deux mots dans $\mathrm{dom}(f)$, et $w = u \wedge v$. On peut écrire $u = wu_1$ et $v = wv_1$, et on a alors $d(u, v) = |u_1| + |v_1|$. Comme $\mathcal{T}$ est séquentiel, il existe un unique chemin $q_0 \xrightarrow{w|w'} q_1$, où q_0 est l'unique état initial, ainsi que deux chemins $q_1 \xrightarrow{u_1|u_1'} q_2$ et $q_1 \xrightarrow{v_1|v_1'} q_3$, avec $q_2, q_3 \in \mathrm{dom}(F)$ et $f(u) = w'u_1'F(q_2)$, $f(v) = w'v_1'F(q_3)$.
On a alors

$$\begin{array}{rcl} d(f(u), f(v)) & = & d(u_1'F(q_2), v_1'F(q_3)) \\ & \leq & |u_1'F(q_2)| + |v_1'F(q_3)| \\ & \leq & Md(u, v) + 2M_F \ . \end{array}$$

(La première égalité découle du lemme suivant : étant donnés trois mots x, y, z, on a $d(zx, zy) = d(x, y)$, qui est lui-même une conséquence immédiate de $zx \wedge zy = z(x \wedge y)$.)
La fonction f est donc à variation bornée.

3.4.
La fonction $f_{\mathcal{T}_1}$ est séquentielle, car $\mathcal{T}_1$ est séquentiel. La fonction $f_{\mathcal{T}_2}$ n'est pas séquentielle, car elle n'est pas à variation bornée. En effet, soit $u = \langle 2^n - 1 \rangle = 1 \ldots 1$ (n fois) et $v = 1 \ldots 12$ ($n - 1$ fois 1). On a $u \wedge v = 1 \ldots 1$ ($n - 1$ fois), et donc $d(u, v) = 2$. Mais $f_{\mathcal{T}_2}(u) = u$ tandis que $f_{\mathcal{T}_2}(v) = \langle 2^n \rangle = 10 \ldots 0$ (n fois 0), de sorte que $d(f_{\mathcal{T}_2}(u), f_{\mathcal{T}_2}(v)) = 2n - 1$, qui n'est pas borné.

3.5.
Par symétrie, il suffit de traiter le cas $w_1 = v_1x$, $xw_2 = v_2x$. On a alors

$$w_1w_2w_3 = v_1xw_2w_3 = v_1v_2xw_3, \text{ et } w_1w_3 = v_1xw_3 \ .$$

Le lemme de la question 3.3 donne

$$d(v_1v_2v_3, w_1w_2w_3) = d(v_1v_2v_3, v_1v_2xw_3) = d(v_3, xw_3) = d(v_1v_3, v_1xw_3) = d(v_1v_3, w_1w_3) \ .$$

3.6.
Nous allons démontrer la propriété par récurrence sur $|u|$. Si $|u| \leq n^2$, on a de manière évidente $|v| \leq Mn^2$ et $|w| \leq Mn^2$, donc $d(v, w) \leq 2Mn^2$.
Supposons la propriété vraie si $|u| < \ell$, pour un certain $\ell > n^2$, et montrons qu'elle est vraie aussi quand $|u| = \ell$. Les deux chemins $i \xrightarrow{u|v} p$ et $j \xrightarrow{u|w} q$ sont de longueur ℓ, on peut donc les écrire de manière extensive

$$p_0 = i \xrightarrow{x_1|y_1} p_1 \xrightarrow{x_2|y_2} \cdots \xrightarrow{x_{\ell-1}|y_{\ell-1}} p_{\ell-1} \xrightarrow{x_\ell|y_\ell} p_\ell = p$$

et

$$q_0 = j \xrightarrow{x_1|z_1} q_1 \xrightarrow{x_2|z_2} \cdots \xrightarrow{x_{\ell-1}|z_{\ell-1}} q_{\ell-1} \xrightarrow{x_\ell|z_\ell} q_\ell = q$$

avec $x_k \in A$, $y_k, z_k \in B$. Parmi les ℓ couples d'états (p_k, q_k), $1 \leq k \leq \ell$, au moins deux sont égaux puisque $\ell > n^2$, disons $(p_h, q_h) = (p_k, q_k)$ avec $h < k$. Posons $u_1 = x_1 \dots x_h$, $u_2 = x_{h+1} \dots x_k$, $u_3 = x_{k+1} \dots x_\ell$, $v_1 = y_1 \dots y_h$, $v_2 = y_{h+1} \dots y_k$, $v_3 = y_{k+1} \dots y_\ell$, $w_1 = z_1 \dots z_h$, $w_2 = z_{h+1} \dots z_k$, $w_3 = z_{k+1} \dots z_\ell$. La propriété de jumelage entraîne alors soit $v_2 = w_2 = \mathbf{1}_{B^*}$, soit

$$\exists x \in A^*, [(w_1 = v_1 x,\ x w_2 = v_2 x) \text{ ou } (v_1 = w_1 x,\ x v_2 = w_2 x)].$$

Dans les deux cas on a $d(v, w) = d(v_1 v_2 v_3, w_1 w_2 w_3) = d(v_1 v_3, w_1 w_3)$, en appliquant la question 3.5. De plus, on a un couple de chemins $i \xrightarrow{u_1 u_3 | v_1 v_3} p$ et $j \xrightarrow{u_1 u_3 | w_1 w_3} q$, ou plus précisément

$$p_0 = i \xrightarrow{x_1|y_1} \cdots \xrightarrow{x_h|y_h} p_h \xrightarrow{x_{k+1}|y_{k+1}} p_{k+1} \xrightarrow{x_{k+2}|y_{k+2}} \cdots \xrightarrow{x_\ell|y_\ell} p_\ell = p$$

et

$$q_0 = j \xrightarrow{x_1|z_1} \cdots \xrightarrow{x_h|z_h} q_h \xrightarrow{x_{k+1}|z_{k+1}} q_{k+1} \xrightarrow{x_{k+2}|z_{k+2}} \cdots \xrightarrow{x_\ell|z_\ell} q_\ell = q$$

avec $|u_1 u_3| < \ell$, donc par hypothèse de récurrence $d(v_1 v_3, w_1 w_3) \leq 2Mn^2$, donc $d(v, w) \leq 2Mn^2$.

3.7.

Supposons que le transducteur $\mathcal{T}$ vérifie la propriété de jumelage. Soient u et u' deux mots dans $\mathrm{dom}(f_\mathcal{T})$; posons $u_0 = u \wedge u'$, $u = u_0 u_1$ et $u' = u_0 u_1'$ de sorte que $d(u, u') = |u_1| + |u_1'|$. Dans $\mathcal{T}$, on a des chemins réussis $i \xrightarrow{u_0|v_0} p \xrightarrow{u_1|v_1} p'$ et $j \xrightarrow{u_0|w_0} q \xrightarrow{u_1'|w_1} q'$, avec $f_\mathcal{T}(u) = v_0 v_1 F(p')$ et $f_\mathcal{T}(u') = w_0 w_1 F(q')$. D'après la question 3.6, $d(v_0, w_0) \leq 2n^2 M$ (où M et n sont définis comme précédemment), et de plus $|v_1| \leq M|u_1|$, $|w_1| \leq M|u_1'|$, $|F(p')| \leq M_F$ et $|F(q')| \leq M_F$ (M_F a été défini à la question 3.3). On a donc

$$\begin{aligned} d(f_\mathcal{T}(u), f_\mathcal{T}(u')) = d(v_0 v_1 F(p'), w_0 w_1 F(q')) &\leq d(v_0, w_0) + |v_1 F(p')| + |w_1 F(q')| \\ &\leq 2n^2 M + M d(u, u') + 2M_F\ , \end{aligned}$$

ce qui montre que $f_\mathcal{T}$ est à variation bornée.

Réciproquement, supposons que $f_\mathcal{T}$ est à variation bornée. Soit une paire de chemins $i \xrightarrow{u_1|v_1} p \xrightarrow{u_2|v_2} p$ et $j \xrightarrow{u_1|w_1} q \xrightarrow{u_2|w_2} q$ dans $\mathcal{T}$, i et j étant des états initiaux. Comme $\mathcal{T}$ est supposé émondé, on a aussi des chemins $p \xrightarrow{u_3|v_3} p'$ et $q \xrightarrow{u_3'|w_3} q'$, avec p' et q' états finaux. On a alors, pour tout entier $h \in \mathbb{N}$, $f_\mathcal{T}(u_1 u_2^h u_3) = v_1 v_2^h v_3 F(p')$ et $f_\mathcal{T}(u_1 u_2^h u_3') = w_1 w_2^h w_3 F(q')$. Comme $d(u_1 u_2^h u_3, u_1 u_2^h u_3') = d(u_3, u_3')$ est indépendant de h et comme $f_\mathcal{T}$ est à variation bornée, il existe un entier K tel que pour tout h, $d(v_1 v_2^h v_3 F(p'), w_1 w_2^h w_3 F(q')) \leq K$, donc $d(v_1 v_2^h v_3, w_1 w_2^h w_3) \leq K + 2M_F$. Alors

$$\begin{aligned} K + 2M_F &\geq d(v_1 v_2^h v_3, w_1 w_2^h w_3) \\ &\geq \big||v_1 v_2^h v_3| - |w_1 w_2^h w_3|\big| \\ &\geq h\big||v_2| - |w_2|\big| - \big||v_1 v_3| - |w_1 w_3|\big|\ , \end{aligned}$$

ce qui n'est possible pour tout h que si $|v_2| = |w_2|$.

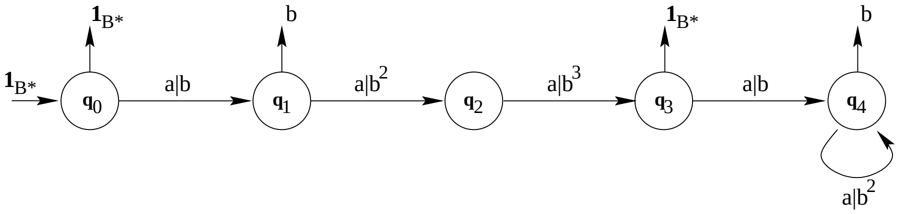
Supposons que l'on n'a pas $v_2 = w_2 = \mathbf{1}_{B^*}$, alors

$$\begin{aligned} |v_1 v_2^h v_3 \wedge w_1 w_2^h w_3| &= \frac{1}{2}\big(|v_1 v_2^h v_3| + |w_1 w_2^h w_3| - d(v_1 v_2^h v_3, w_1 w_2^h w_3)\big) \\ &\geq h|v_2| + \frac{1}{2}(|v_1 v_3| + |w_1 w_3| - K - 2M_F)\ , \end{aligned}$$

et en faisant tendre h vers $+\infty$, on trouve que les mots infinis $v_1v_2v_2v_2\ldots$ et $w_1w_2w_2w_2\ldots$ ont des préfixes communs arbitrairement grands, donc sont égaux. On peut aussi procéder comme suit, sans parler de mots infinis. Supposons par exemple que $|w_1| \geq |v_1|$. Pour h suffisamment grand, $|v_1v_2^hv_3 \wedge w_1w_2^hw_3| \geq |w_1w_2|$ et $|v_1v_2^h| \geq |w_1w_2|$, de sorte que w_1w_2 est préfixe de $v_1v_2^h$. Il existe alors un mot x tel que $w_1 = v_1x$, et xw_2 est préfixe de v_2^h : v_2xw_2 est préfixe de v_2^{h+1}, donc v_2x et xw_2 sont tous deux préfixes de v_2^{h+1} et donc, comme ils ont la même longueur, $v_2x = xw_2$. On conclut que $\mathcal{T}$ vérifie la propriété de jumelage.

3.8.

On ne détaille pas entièrement la construction. On obtient $Q_S = \{\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_2, \mathbf{q}_3, \mathbf{q}_4\}$, $\mathrm{dom}(I_S) = \{\mathbf{q}_0\}$, $I_S(\mathbf{q}_0) = \mathbf{1}_{B^*}$, $T_S(\mathbf{q}_0, a, \mathbf{q}_1) = b$, $T_S(\mathbf{q}_1, a, \mathbf{q}_2) = b^2$, $T_S(\mathbf{q}_2, a, \mathbf{q}_3) = b^3$, $T_S(\mathbf{q}_3, a, \mathbf{q}_4) = b$, $T_S(\mathbf{q}_4, a, \mathbf{q}_4) = b^2$, et T_S n'est pas défini pour les autres triplets. Les états sont : $\mathbf{q}_0 = \{(1, \mathbf{1}_{B^*}), (4, \mathbf{1}_{B^*})\}$, $\mathbf{q}_1 = \{(1, b), (2, \mathbf{1}_{B^*}), (3, b^2)\}$, $\mathbf{q}_2 = \{(2, \mathbf{1}_{B^*}), (3, b^2), (4, b)\}$, $\mathbf{q}_3 = \{(1, \mathbf{1}_{B^*}), (3, b), (4, \mathbf{1}_{B^*})\}$ et $\mathbf{q}_4 = \{(1, b), (2, \mathbf{1}_{B^*}), (3, b^2), (4, b)\}$.



Il reste à adjoindre au quintuplet $\mathcal{T}_S$ une fonction de sortie F_S pour obtenir un transducteur que nous noterons encore $\mathcal{T}_S$. Pour définir F_S, pour tout état $\mathbf{q} \in Q_S$ tel qu'il existe $(q_i, w_i) \in \mathbf{q}$ avec $q_i \in \mathrm{dom}(F)$, on pose $F_S(\mathbf{q}) = w_iF(q_i)$; cette valeur ne doit pas dépendre du couple (q_i, w_i) choisi sinon le transducteur ne serait pas fonctionnel (cette construction sera justifiée à la question 3.10). Ici, on trouve $\mathrm{dom}(F_S) = \{\mathbf{q}_0, \mathbf{q}_1, \mathbf{q}_3, \mathbf{q}_4\}$, $F_S(\mathbf{q}_0) = F_S(\mathbf{q}_3) = \mathbf{1}_{B^*}$, $F_S(\mathbf{q}_1) = F_S(\mathbf{q}_4) = b$. On vérifie aisément que $\mathcal{T}_S$ est séquentiel et que $f_{\mathcal{T}_S} = f_{\mathcal{T}}$ (en particulier, $\mathrm{dom}(f_{\mathcal{T}_S}) = \mathrm{dom}(f_{\mathcal{T}}) = \{a\}^* \setminus \{a^2\}$).

3.9.

Dans cette question et la suivante, on va démontrer que le quintuplet $\mathcal{T}_S$ permet de séquentialiser un transducteur fonctionnel vérifiant la propriété de jumelage. Cela permet d'achever la démonstration du théorème de séquentialité.

Si $\mathbf{q} = \mathbf{q}_0$, alors les éléments de $\mathbf{q}$ sont tous de la forme $(i, \mathbf{1}_{B^*})$, donc il suffit de prendre $(q, w) = (p, v)$: $v \wedge w = v = w = \mathbf{1}_{B^*}$.

Si $\mathbf{q} \neq \mathbf{q}_0$, il existe un état $\mathbf{q}' \in Q_S$ et une lettre $a \in A$ tels que $\mathbf{q} = \mathbf{q}' \cdot a$. Il existe alors $(q_i, w_i) \in \mathbf{q}'$ tel que $(p, w_i^{-1}(\mathbf{q}' \star a)v) \in V(q_i, w_i, a)$, c'est-à-dire tel que dans $\mathcal{T}$, $T(q_i, a, p) = w_i^{-1}(\mathbf{q}' \star a)v$. On a alors $(\mathbf{q}' \star a)v \in W(q_i, w_i, a)$.

Par définition de $\mathbf{q}' \star a$, il existe $v' \in W(q_j, w_j, a)$, avec $(q_j, w_j) \in \mathbf{q}'$, tel que $\mathbf{q}' \star a = ((\mathbf{q}' \star a)v) \wedge v'$. On peut alors poser $w = (\mathbf{q}' \star a)^{-1}v'$. Dans $\mathcal{T}$, il existe q tel que $T(q_j, a, q) = w_j^{-1}v'$, donc $(q, w) \in \mathbf{q}$, et on a bien $v \wedge w = \mathbf{1}_{B^*}$.

On suppose de plus que $\mathcal{T}$ vérifie la propriété de jumelage. Par construction, tous les éléments de Q_S sont obtenus à partir de $\mathbf{q}_0$ en appliquant un nombre fini de fois les transformations $\mathbf{q} \mapsto \mathbf{q} \cdot a$, $a \in A$. Notons $\mathbf{q} \cdot u = (\cdots((\mathbf{q} \cdot a_1) \cdot a_2) \cdots) \cdot a_k$ pour tout mot $u = a_1a_2\ldots a_k \in A^*$.

(Pour que cela ait toujours un sens, il est nécessaire de prolonger la transformation $\mathbf{q} \mapsto \mathbf{q} \cdot a$ par $\mathbf{q} \cdot a = \emptyset$ s'il n'existe pas $(p, v) \in \mathbf{q}$ et $r \in Q$ tels que $(p, a, r) \in \mathrm{dom}(T)$, et $\emptyset \cdot a = \emptyset$.) On a alors $Q_S = \{\mathbf{q}_0 \cdot u \ :\ u \in A^*\} \setminus \{\emptyset\}$. Définissons aussi $\mathbf{q} \star u = (\mathbf{q} \star a_1).\big((\mathbf{q} \cdot a_1) \star a_2\big).\big((\mathbf{q} \cdot a_1 a_2) \star a_3\big) \cdots \big((\mathbf{q} \cdot a_1 a_2 \cdots a_{k-1}) \star a_k\big)$ (on peut aussi donner une définition récurrente : $\mathbf{q} \star \mathbf{1}_{A^*} = \mathbf{1}_{B^*}$ et $\mathbf{q} \star (ua) = (\mathbf{q} \star u).((\mathbf{q} \cdot u) \star a)$ si $u \in A^*$ et $a \in A$), en posant $\mathbf{q} \star a = \mathbf{1}_{B^*}$ quand $\mathbf{q} \cdot a = \emptyset$.
On a alors la caractérisation suivante : $(p, v) \in \mathbf{q}_0 \cdot u$ si et seulement s'il existe $i \in \mathrm{dom}(I)$ tel que $i \xrightarrow{u|(\mathbf{q}_0 \star u)v} p$. La preuve est par récurrence sur $|u|$: si $u = \mathbf{1}_{A^*}$, c'est la définition de $\mathbf{q}_0$; supposons la propriété établie pour $\mathbf{q} = \mathbf{q}_0 \cdot u$, et soit $\mathbf{q}' = \mathbf{q} \cdot a$. Alors par définition $(p, v) \in \mathbf{q}'$ si et seulement s'il existe $(q_j, v_j) \in \mathbf{q}$ tel que $(p, w_j^{-1}(\mathbf{q} \star a)v) \in V(q_j, w_j, a)$, c'est-à-dire $T(q_j, a, p) = w_j^{-1}(\mathbf{q} \star a)v$ dans $\mathcal{T}$. Par hypothèse de récurrence, il existe $i \in \mathrm{dom}(I)$ tel que $i \xrightarrow{u|(\mathbf{q}_0 \star u)w_j} q_j \xrightarrow{a|w_j^{-1}(\mathbf{q} \star a)v} p$, de sorte que $i \xrightarrow{ua|(\mathbf{q}_0 \star (ua))v} p$.
Soit $\mathbf{q} = \mathbf{q}_0 \cdot u \in Q_S$ et $(p, v) \in \mathbf{q}$. D'après le début de la question, il existe $(q, w) \in \mathbf{q}$ tel que $v \wedge w = \mathbf{1}_{B^*}$. D'après la caractérisation ci-dessus, il existe $i, j \in \mathrm{dom}(I)$ tels que $i \xrightarrow{u|(\mathbf{q}_0 \star u)v} p$ et $j \xrightarrow{u|(\mathbf{q}_0 \star u)w} q$. D'après la question 3.6, $d((\mathbf{q}_0 \star u)v, (\mathbf{q}_0 \star u)w) \le 2n^2 M$. Or $d((\mathbf{q}_0 \star u)v, (\mathbf{q}_0 \star u)w) = d(v, w) = |v| + |w|$, donc $|v| \le 2n^2 M$.
On en déduit que Q_S est inclus dans l'ensemble des parties de $Q \times (\bigcup_{i \le 2n^2 M} B^i)$, qui est fini, donc Q_S est fini.

3.10.

Puisque $\mathcal{T}$ est fonctionnel, si $i \xrightarrow{u|v} p$ et $j \xrightarrow{u|w} q$ sont deux chemins réussis on a nécessairement $vF(p) = wF(q) = f_{\mathcal{T}}(u)$. Soit $\mathbf{q} = \mathbf{q}_0 \cdot u \in Q_S$; s'il existe $(p, v) \in \mathbf{q}$ tel que $p \in \mathrm{dom}(F)$, on pose $F_S(\mathbf{q}) = vF(p)$. Cette définition ne dépend pas du choix de (p, v) dans $\mathbf{q}$ puisque, si un autre choix (q, w) convient, on a d'après la caractérisation de la question 3.9, $i \xrightarrow{u|(\mathbf{q} \star u)v} p$ et $j \xrightarrow{u|(\mathbf{q} \star u)w} q$ avec $i, j \in \mathrm{dom}(I)$, et donc, comme remarqué ci-dessus, $(\mathbf{q} \star u)vF(p) = (\mathbf{q} \star u)wF(q)$, donc $vF(p) = wF(q)$. On retrouve la fonction F_S utilisée à la question 3.8.
Notons à nouveau $\mathcal{T}_S = (Q_S, A, B, T_S, I_S, F_S)$: comme Q_S est fini d'après la question 3.9, c'est un transducteur, et il est séquentiel puisque $\mathrm{dom}(I_S) = \{\mathbf{q}_0\}$ est un singleton et, pour tous $\mathbf{p} \in Q_S$ et $a \in A$,

$$\{\mathbf{q} \in Q_S \ :\ (\mathbf{p}, a, \mathbf{q}) \in \mathrm{dom}(T_S)\} = \begin{cases} \{\mathbf{p} \cdot a\} \text{ si } \mathbf{p} \cdot a \neq \emptyset \\ \emptyset \text{ sinon} \end{cases}$$

Dans $\mathcal{T}_S$, les chemins sont tous de la forme $\mathbf{q} \xrightarrow{u|\mathbf{q} \star u} \mathbf{q} \cdot u$.
Il reste à vérifier que $f_{\mathcal{T}_S} = f_{\mathcal{T}}$. Si $u \in \mathrm{dom}(f_{\mathcal{T}_S})$, alors $\mathbf{q} = \mathbf{q}_0 \cdot u \in \mathrm{dom}(F_S)$, donc il existe $(p, v) \in \mathbf{q}$ tel que $p \in \mathrm{dom}(F)$. On a alors $f_{\mathcal{T}_S}(u) = (\mathbf{q}_0 \star u)F_S(\mathbf{q}) = (\mathbf{q}_0 \star u)vF(p) = f_{\mathcal{T}}(u)$.
Réciproquement, si $u \in \mathrm{dom}(f_{\mathcal{T}})$, il existe un chemin réussi $i \xrightarrow{u|v} p$ dans $\mathcal{T}$. Alors il existe $w \in B^*$ tel que $(p, w) \in \mathbf{q}_0 \cdot u$, et on a $F_S(\mathbf{q}_0 \cdot u) = wF(p)$ et $v = (\mathbf{q}_0 \star u)w$. Alors $f_{\mathcal{T}}(u) = vF(p) = (\mathbf{q}_0 \star u)F_S(\mathbf{q}_0 \cdot u) = f_{\mathcal{T}_S}(u)$.

Proposons maintenant un algorithme ayant la propriété voulue. On part d'un transducteur $\mathcal{T}$, supposé émondé, et on lui applique le procédé itératif pour construire $\mathcal{T}_S$. À chaque étape, après avoir construit un nouvel état $\mathbf{q} = \mathbf{q}_0 \cdot u = \{(q_1, w_1), \ldots, (q_k, w_k)\}$, on construit $E = \{w_i F(q_i) \ :\ (q_i, w_i) \in \mathbf{q}, q_i \in \mathrm{dom}(F)\}$. Si cet ensemble E contient au moins deux éléments distincts, on répond 'non' et on sort de l'algorithme : en effet, on a établi que $\mathcal{T}$ n'est pas fonctionnel ($f(u)$ a au moins deux éléments). Si E n'a qu'un seul élément, on pose $\{F_S(\mathbf{q})\} = E$; si E est vide, alors $\mathbf{q} \notin \mathrm{dom}(F_S)$.

À chaque étape, on teste également si tous les w_i satisfont $|w_i| \leq 2n^2M$. Si ce n'est pas le cas, alors on répond 'non' et on sort de l'algorithme, car on a établi que $\mathcal{T}$ ne vérifie pas la propriété de jumelage d'après la question 3.9.

Si l'algorithme ne s'interrompt pas sur une réponse 'non', la construction se termine nécessairement au bout d'un temps fini puisque le second test ne laisse qu'un nombre fini de possibilités pour les états $\mathbf{q}$. On retourne alors le transducteur $\mathcal{T}_S = (Q_S, A, B, T_S, I_S, F_S)$.

Justifions cet algorithme. Si $f_{\mathcal{T}}$ est une fonction séquentielle, l'algorithme ne répond jamais 'non' et on a vu au début de la question que le transducteur retourné $\mathcal{T}_S$ est bien un transducteur séquentiel réalisant $f_{\mathcal{T}}$.

Réciproquement, supposons que $f_{\mathcal{T}}$ retourne un transducteur $\mathcal{T}_S$. Par construction, ce transducteur est séquentiel. Si $i \xrightarrow{u|v} p$ est un chemin réussi dans $\mathcal{T}$, alors $\mathbf{q}_0 \xrightarrow{u|\mathbf{q}_0\star u} \mathbf{q}_0 \cdot u$ est un chemin réussi dans $\mathcal{T}_S$, et $vF(p) = (\mathbf{q}_0\star u)F_S(\mathbf{q}_0\cdot u) = f_{\mathcal{T}_S}(u)$. On a donc $\mathrm{dom}(f_{\mathcal{T}}) \subset \mathrm{dom}(f_{\mathcal{T}_S})$ et pour tout $u \in \mathrm{dom}(f_{\mathcal{T}})$, $f_{\mathcal{T}}(u) = \{f_{\mathcal{T}_S}(u)\}$. Mais si $\mathbf{q}_0 \xrightarrow{u|\mathbf{q}_0\star u} \mathbf{q}$ est un chemin réussi dans $\mathcal{T}_S$, alors il existe $(p, w) \in \mathbf{q}$ tel que $p \in \mathrm{dom}(F)$, donc il existe un chemin réussi $i \xrightarrow{u|v} p$ dans $\mathcal{T}$, donc $\mathrm{dom}(f_{\mathcal{T}_S}) \subset \mathrm{dom}(f_{\mathcal{T}})$. On en conclut que $f_{\mathcal{T}} = f_{\mathcal{T}_S}$ est bien une fonction séquentielle.

Bilan de l'épreuve

La première partie n'est bien traitée que par une petite minorité des candidats. La question 1.1 est déjà très sélective : de nombreux candidats n'arrivent pas à proposer un algorithme correct et bien trop rares sont ceux qui justifient leur algorithme. La seconde partie a été mieux traitée dans l'ensemble. Enfin la troisième partie n'a été réellement abordée que dans peu de copies. À titre indicatif, voici une évaluation de la proportion de candidats ayant répondu correctement aux différentes questions des deux premières parties :

Question	**1.1**	**1.2**	**1.3**	**1.4**	**1.5**	**2.1**	**2.2**	**2.3**	**2.4**	**2.5**
%	25	70	45	10	2	65	50	20	25	5

Notes bibliographiques

Pour plus de détails sur la première partie du sujet, on pourra se référer à [7], chapitre *Algorithmes sur les graphes*. En particulier, les algorithmes des questions 1.4 et 1.5 y sont détaillés sous le nom respectivement d'algorithmes de Floyd-Warshall et de Bellman-Ford.

Pour un exposé général sur les transducteurs, on pourra consulter [3, 8]. On y verra en particulier une extension du théorème de Kleene : l'équivalence entre les graphes des transductions de A^* vers B^* réalisées par un transducteur, d'une part, et les *parties rationnelles* du monoïde $A^* \times B^*$, d'autre part.

La terminologie la plus classique dans la littérature est légèrement différente de celle adoptée dans le sujet : nos *transducteurs séquentiels* sont en général appelés *transducteurs sous-*

séquentiels. Le théorème de (sous-)séquentialité qui forme le cœur de la troisième partie est du à Choffrut. Il est démontré dans [5], la notion de jumelage est aussi introduite dans [4] et la caractérisation à l'aide de la distance d dans [6]. La preuve proposée dans le sujet est également inspirée de [1]. L'existence d'algorithmes polynomiaux pour décider de la fonctionnalité et de la séquentialité (mentionnés à la fin du sujet) a été démontrée récemment, voir [1, 2, 9] pour plus de détails.

Remerciements

Les auteurs tiennent à remercier Alain Darte (ENS Lyon, CNRS) pour avoir participé à l'élaboration du sujet ainsi que pour avoir apporté ses commentaires et suggestions éclairés sur le présent corrigé.

Références

[1] M.-P. Béal et O. Carton. Determinization of transducers over infinite words. *ICALP'2000, Lect. Notes Comput. Sci.* 1853, pp. 561–570, Springer Verlag, 2000. Aussi : rapport de recherche IGM 99-12, Institut Gaspard Monge, Université de Marne-la-Vallée.

[2] M.-P. Béal, O. Carton, C. Prieur et J. Sakarovitch. Squaring transducers: An efficient procedure for deciding functionality and sequentiality. *LATIN'2000*, *Lect. Notes Comput. Sci.* 1776, pp. 397–406, Springer Verlag, 2000.

[3] J. Berstel. *Transductions and context-free languages*. B. G. Teubner, 1979.

[4] C. Choffrut. Une caractérisation des fonctions séquentielles et des fonctions sous-séquentielles en tant que relations rationnelles. *Elektronische Informations-verarbeitung und Kybernetik*, 5:324–338, 1977.

[5] C. Choffrut. *Contribution à l'étude de quelques familles remarquables de fonctions rationnelles*. Thèse d'état, Université Paris 7, 1978.

[6] C. Choffrut. A generalization of Ginsburg and Rose's characterisation of g-s-m mappings. In *ICALP 79*, number 71 in LNCS, pages 88–103. Springer-Verlag, 1979.

[7] T. Cormen, C. Leiserson et R. Rivest. *Introduction à l'algorithmique*. Dunod, 1994. Édition originale : *Introduction to Algorithms*, MIT Press, 1990.

[8] S. Eilenberg. *Automata, languages and machines*, volume A. Academic Press, New York, 1974.

[9] C. Prieur. *Fonctions rationnelles de mots infinis et continuité*. Thèse de doctorat, Université Paris 7, 2000.