Sujet

Les parties I et II sont indépendantes. La partie III peut être traitée en admettant les résultats de la partie I.

Définitions et notations

- Un *graphe* $G = (S, A)$ est constitué d'un ensemble fini de sommets S et d'un ensemble d'arcs $A \subseteq S \times S$. On considère donc les graphes orientés sans arcs multiples mais pouvant contenir des arcs bouclant sur un sommet.
- Un *chemin* c dans un graphe est une suite de sommets

$$c = (s_0, s_1, \ldots, s_n)$$

  telle que $n \in \mathbb{N}$ et pour tout $i \in \mathbb{N}$ tel que $i \leq n-1$, $(s_i, s_{i+1}) \in A$. Le chemin c est appelé un *chemin de s_0 à s_n*.
- Un graphe *pondéré* (S, A, ω) consiste en un graphe (S, A) muni d'une fonction de pondération $\omega : A \to H$ où H est un ensemble quelconque. $\omega(a)$ est appelé *poids* de l'arc a. Si $(H, \otimes)$ est un monoïde, on peut étendre ω à l'ensemble des chemins :

$$\omega(s_0, s_1, \ldots, s_n) = \omega((s_0, s_1)) \otimes \omega((s_1, s_2)) \otimes \ldots \otimes \omega((s_{n-1}, s_n))$$

  Le poids d'un chemin vide (ne contenant pas d'arc) est évidemment $\mathbf{1}$, élément neutre du monoïde.
- Un *semi-anneau* $(H, \oplus, \otimes)$ avec zéro $\mathbf{0}$ et unité $\mathbf{1}$ est une structure munie des deux opérations $\oplus$ et $\otimes$ et respectant les quatre axiomes suivants.

  (S_1) $(H, \oplus)$ est un semi-groupe commutatif avec $\mathbf{0}$ pour élément neutre,
  (S_2) $(H, \otimes)$ est un semi-groupe avec $\mathbf{1}$ pour élément neutre,
  (S_3) $\otimes$ est distributif par rapport à $\oplus$,
  (S_4) $\mathbf{0}$ est absorbant pour $\otimes$.
- *Semi-anneaux de matrices* : si $(H, \oplus, \otimes)$ est un semi-anneau, on peut munir l'ensemble $H^{n \times n}$ des matrices de taille $n \times n (n \geq 1)$ d'une structure de semi-anneau, comme en algèbre linéaire classique. La matrice zéro contient uniquement la valeur $\mathbf{0}$, et la matrice unité I contient des $\mathbf{1}$ sur la diagonale et des $\mathbf{0}$ partout ailleurs.

I Problème du chemin algébrique

Le problème du chemin algébrique peut être formulé de la façon suivante : étant donné un graphe orienté pondéré $G = (S, A, \omega)$, avec $\omega : A \to H$, où H est un semi-anneau comme défini précédemment, trouver pour tout couple (i, j) de sommets la valeur $d_{i,j}$ définie par

$$d_{i,j} = \bigoplus_{\substack{c \text{ chemin} \\ \text{de } i \text{ à } j \\ \text{dans } G}} \omega(c)$$

La solution à ce problème n'existe pas obligatoirement car l'ensemble des chemins de i à j peut être infini. Cependant, pour certains semi-anneaux, une solution peut exister même pour un ensemble infini de chemins. Pour pouvoir parler de sommes infinies de façon consistante, on supposera que les deux propriétés suivantes sont vérifiées.

(S_5) Soient I et J deux ensembles dénombrables, alors l'égalité

$$\bigoplus_{(i,j)\in I\times J} a_i \otimes b_j = \bigoplus_{i\in I} a_i \otimes \bigoplus_{j\in J} b_j$$

est vérifiée dès que les deux sommes de la partie droite sont définies.

(S_6) Soient I, J et I_j des ensembles dénombrables tels que $\{I_j \mid j \in J\}$ est une partition de I, alors l'égalité

$$\bigoplus_{i\in I} a_i = \bigoplus_{j\in J} \bigoplus_{i\in I_j} a_i$$

est vérifiée dès que la somme en partie gauche est définie.

Nous aurons besoin de calculer des sommes infinies de la forme suivante, pour lesquelles nous introduisons la notation

$$c^* = \bigoplus_{i\geq 0} c^i = \mathbf{1} \oplus c \oplus (c \otimes c) \oplus (c \otimes c \otimes c) \oplus \dots$$

Applications du problème du chemin algébrique

Question 1. *Donner une structure de semi-anneau pour laquelle le problème du chemin algébrique se ramène à calculer la relation d'accessibilité dans un graphe orienté.*

Question 2. *Donner une structure de semi-anneau pour laquelle le problème du chemin algébrique se ramène au problème du calcul des plus courts chemins dans un graphe orienté.*

Représentation matricielle À un graphe pondéré (S, A, ω) on peut associer une matrice de taille $|\ S\ | \times |\ S\ |$

$$M = (m_{ij}),\ \text{où}\ m_{ij} = \begin{cases} \omega((i,j)) \text{ si } (i,j) \in A \\ \mathbf{0} \text{ si } (i,j) \notin A \end{cases}$$

Question 3. *Montrer que pour tout entier p,*

$$M^p = (m_{ij}^{(p)}),\ \textit{où}\ m_{ij}^{(p)} = \bigoplus_{\substack{c \text{ chemin de } i \text{ à } j \\ \text{contenant exacte-} \\ \text{ment } p \text{ arcs}}} \omega(c)$$

Question 4. *Montrer que calculer l'inverse d'une matrice à coefficients dans $\mathbb{R}$ peut se ramener au problème du chemin algébrique, pour un semi-anneau que l'on précisera. Les problèmes liés à l'existence d'une solution devront être soigneusement pris en compte.*

Version algorithmique On se propose de trouver un algorithme qui résout le problème du chemin algébrique (sous l'hypothèse qu'une solution existe). Soient i et j deux sommets, et k un entier tel que $0 \leq k \leq n$. On note $\mathcal{C}_{ij}^{(k)}$ l'ensemble de *tous* les chemins de i à j dont tous les sommets intermédiaires x vérifient $1 \leq x \leq k$ et

$$p_{ij}^{(k)} = \bigoplus_{c \in \mathcal{C}_{ij}^{(k)}} \omega(c)$$

Les sommets intermédiaires d'un chemin sont tous les sommets sauf le sommet initial et le sommet final. Le chemin constitué d'un seul arc de i à j est contenu dans $\mathcal{C}_{ij}^{(0)}$ (et donc dans tous les $\mathcal{C}_{ij}^{(k)}$). Par convention, le chemin vide qui commence et finit au nœud i n'est pas contenu dans $\mathcal{C}_{ii}^{(i-1)}$. En revanche, il est contenu dans $\mathcal{C}_{ii}^{(i)}$.

Question 5. *Donner les formules de récurrence permettant de calculer les $p_{ij}^{(k)}$, en considérant les valeurs respectives de i, j et k et en partitionnant $\mathcal{C}_{ij}^{(k)}$ de façon adéquate.*

Question 6. *Montrer précisément comment à partir de ces formules de récurrence on peut se ramener à l'algorithme suivant, constitué de trois phases, où N représente le nombre de sommets du graphe.*

```
PHASE 1:
      pour i de 1 a N
      pour j de 1 a N
      faire
          pour k de 1 a min(i,j) - 1
                  p_ij := p_ij ⊕ p_ik ⊗ p_kj ;
          si i = j alors p_ii := p*_ii ;
          si i > j alors p_ij := p_ij ⊗ p_jj ;
      fait;
PHASE 2:
      pour i de 1 a N
      pour j de 1 a N
      faire
          si i < j alors p_ij := p_ii ⊗ p_ij ;
          pour k de min(i,j) + 1 a max(i,j) - 1
                  p_ij := p_ij ⊕ p_ik ⊗ p_kj ;
          si i < j alors p_ij := p_ij ⊗ p_jj ;
      fait;
PHASE 3:
      pour i de 1 a N
      pour j de 1 a N
      faire
          si i > j alors p_ij := p_ii ⊗ p_ij ;
          pour k de max(i,j) + 1 a N
                  p_ij := p_ij ⊕ p_ik ⊗ p_kj ;
      fait;
```

II Réseaux systoliques

Un réseau systolique est une architecture spécialisée qui permet d'implanter un algorithme effectuant un grand nombre de calculs simples de nature identique. Nous ne donnons pas ici de définition formelle d'un réseau systolique, mais les trois principales propriétés qui le caractérisent :

- **parallélisme massif** : un réseau systolique est constitué d'un grand nombre de cellules (processeurs) élélementaires ; une cellule peut effectuer quelques calculs simples et dispose de quelques registres ; les cellules peuvent être de plusieurs types, mais ce nombre de types différents reste très petit ;
- **localité** : les cellules sont reliées entre elles, mais uniquement de façon locale : chaque cellule ne peut être connectée qu'aux cellules voisines (au sens spatial du terme) ; seules les cellules situées à la frontière du réseau communiquent avec le monde extérieur (un processeur-hôte qui alimente le réseau en données d'entrée et récupère les résultats en sortie) ;
- **synchronisme** : les cellules évoluent en parallèle, sous le contrôle d'une horloge globale ; les calculs sont effectués simultanément dans tout le réseau ; à chaque cycle d'horloge, chaque cellule reçoit des données en provenance des cellules voisines (éventuellement complétées par des données de ses registres internes), effectue un calcul simple, puis transmet les résultats (toujours aux cellules voisines) un temps de cycle plus tard.

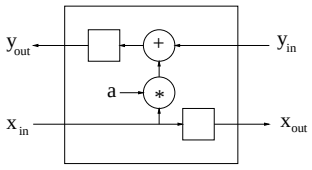
Le fonctionnement d'un réseau systolique peut être vu comme celui d'un ensemble de données qui se déplacent dans le réseau, en étant modifiées par les cellules ou alors simplement transmises. Une même cellule peut présenter des comportements différents selon les temps d'horloge, grâce à des mécanismes de contrôle que nous n'aborderons pas ici.

Pour une cellule donnée, chaque cycle d'horloge ne correspond pas obligatoirement à une donnée valide : il peut y avoir un ou plusieurs cycles ¡¡ à vide ¿¿ entre le passage de deux données consécutives. Même si physiquement parlant il y a bien calcul ou transmission de données à chaque cycle, les données non significatives — car ne participant pas au calcul du résultat souhaité — ne sont pas prises en compte. Pour un cycle donné, une cellule sera dite *active* si elle participe au calcul ou à la tranmission d'une donnée valide, et *passive* sinon.

Convolution non récursive. Notre but est d'implanter un algorithme permettant de résoudre le problème suivant : étant donnée une suite $(x_i)_{i\geq 1}$ de données réelles et une suite finie $(a_1, \ldots, a_k)$ de valeurs réelles fixées, calculer pour tout $i \geq k$ la valeur de

$$y_i = \sum_{j=1}^{k} a_j \times x_{i-j+1}$$

On propose un réseau systolique à une dimension, dont les cellules ont la forme suivante.

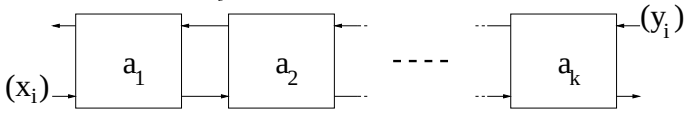


$$x_{out}(t+1) = x_{in}(t)$$
$$y_{out}(t+1) = y_{in}(t) + a \times x_{in}(t)$$

où t est un entier représentant un cycle d'horloge.

L'élément représenté par un rectangle est un registre : il permet de mémoriser une valeur et introduit un délai d'un cycle d'horloge entre son entrée et sa sortie. Le cœfficient a correspond à l'un des a_j.

Le réseau proprement dit est constitué de la juxtaposition et de la connexion de k cellules de type identique (qui ne diffèrent que par la valeur de a). Les cellules sont numérotées de 1 à k (de la gauche vers la droite), et sont pourvues du cœfficient a_j adéquat.



Les données $(x_i)_{i\geq 1}$ se déplacent de la gauche vers la droite, et les valeurs des $(y_i)_{i\geq k}$ en sens inverse, et avancent d'une cellule à chaque cycle d'horloge. Les données sont bien entendu introduites dans le réseau dans l'ordre croissant de leurs indices, et les y_i sont initialisés à 0 avant leur entrée.

Question 7. *On suppose que l'instant $t = 1$ correspond au cycle d'horloge où la donnée x_1 entre dans la cellule 1. Quelle valeur de la suite (y) doit se trouver au même instant dans la même cellule ?*

Question 8. *On se place à l'instant t où la donnée x_i arrive dans la cellule 1. Où se trouve alors la donnée x_{i-1} ? Quelle est la valeur de (y) correspondante ?*

Question 9. *Pour p entier compris entre 1 et k, à quel instant la donnée x_i doit-elle se trouver dans la cellule p ? Même question pour y_i.*

Question 10. *Quel est le débit (rapport entre le nombre de sorties valides et le nombre de cycles d'horloge) du réseau ?*

Question 11. *En supposant que les données sont introduites dans le réseau conformément aux résultats des questins précédentes, montrer que le réseau calcule effectivement le résultat souhaité.*

Convolution récursive. On veut maintenant calculer les valeurs $(y_i)_{i \geq k+1}$ définies par

$$y_i = \sum_{j=1}^{k} a_j \times y_{i-j}$$

où $(y_1, \ldots, y_k)$ sont des valeurs initiales données.

Question 12. *En supposant que les cellules disposent d'un mécanisme qui leur permet à certains instants de transmettre des données de droite à gauche sans les modifier, proposer une modification du réseau précédent qui permet de résoudre ce problème, et prouver sa validité.*

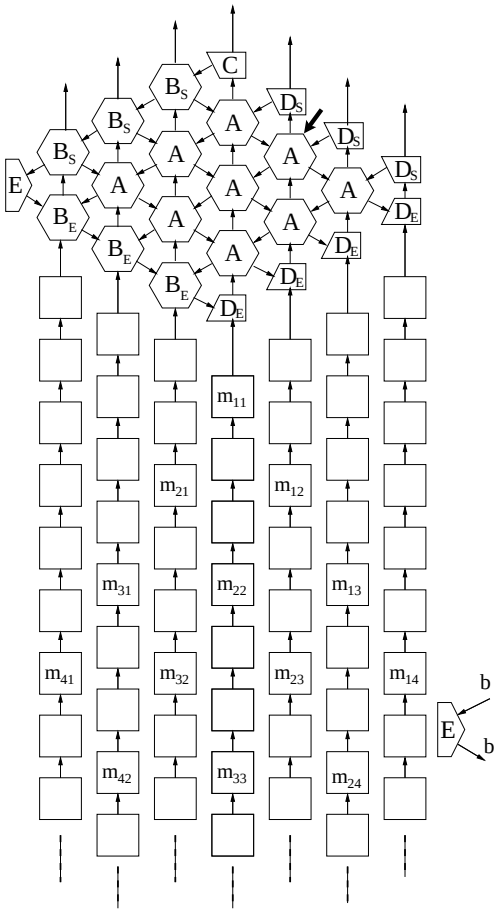
Amélioration du débit

Question 13. *Proposer un réseau* pour la convolution non récursive *où les (x_i) et les (y_i) circulent dans le même sens, et qui améliore le débit de la première version.*

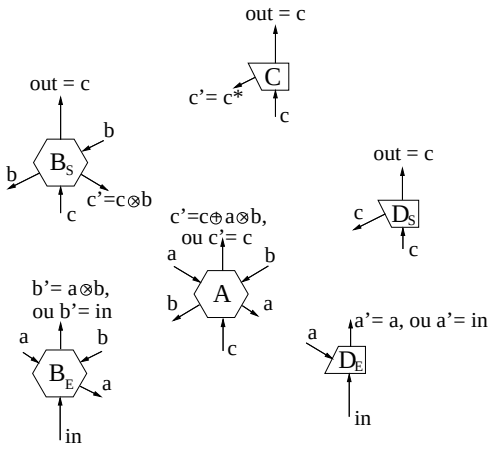
III Réseau systolique pour le problème du chemin algébrique

Notre but est d'implanter l'algorithme de la partie I à l'aide d'un réseau systolique. Nous conservons les définitions et conventions de la partie II pour tout ce qui concerne les caractéristiques générales d'un réseau systolique. On notera N le nombre de sommets du graphe.

Le réseau proposé est un réseau bidimensionnel de taille $(N+1) \times (N+1)$, composé de trois types de cellules. Les cellules sont disposées à l'intérieur d'un losange, et ont chacune au maximum six voisins. L'organisation complète du réseau est donnée en figure 1(a), pour la valeur exemple $N = 4$. On pourra donc repérer une cellule du réseau par deux coordonnées : le numéro z de sa colonne ($-N \leq z \leq N-1$), et le numéro p de la cellule dans sa colonne, compté à partir du bas ($1 \leq p \leq N - \mid z \mid +1$ pour une valeur de z donnée). Par exemple, la cellule signalée par une flèche sur la figure correspond aux coordonnées $(z = 1, p = 3)$.



(a) Le réseau et les entrées–sorties, à $t = 0$. Les rectangles vides correspondent à des données non valides (donc à de simples délais).



(b) Les cellules. Chaque fil de sortie est muni d'un registre qui assure le décalage d'un cycle d'horloge entre entrées et sorties. Chaque cellule dispose en outre d'un mécanisme de contrôle (non détaillé ici), qui permet de choisir entre calcul ou simple transmission des données (pour les cellules de tyoe A et B_E), ou entre plusieurs directions possibles pour la transmission des données (pour les cellules de type B_E et D_E).

FIG. 1 – Le réseau et les différents types de cellules.

Les trois types de cellules correspondent aux trois types de calculs suivants (cf.figure 1(b)) :
- type A : $c' := c \oplus (a \otimes b)$
- type B (avec les variantes B_E et B_S) : $b' := a \otimes b$ ou $c' := c \otimes b$
- type C : $c' = c^*$

À ceci s'ajoutent des cellules de type D_E, D_S et E, qui ne font que transmettre les données avec un délai d'un cycle d'horloge. Les entrées et sorties des cellules sont précisées figure 1(b).

On suppose que tous les registres sont initialisés à la valeur zéro. Puis la matrice est introduite dans le réseau comme spécifié en figure 1(a).

Lorsque des cellules de type B_E ou D_E reçoivent des cœfficients de la matrice, ceux-ci sont transmis vers le haut sans être modifiés. Ces cœfficients sont ensuite acheminés et modifiés à travers le réseau. Lors du passage d'un cœfficient dans une cellule, trois cas se présentent :
- la donnée est *passive* : le cœfficient est simplement transmis d'une entrée vers une sortie ;
- la donnée est *écrite* : le cœfficient est modifié par un calcul avant d'être transmis ;
- la donnée est *lue* : le cœfficient n'est pas modifié avant transmission, mais sa valeur est utilisée pour modifier une autre donnée qui arrive au même instant dans la même cellule.

Les données peuvent donc être écrites dans deux types de situations : soit lors du passage par une cellule de type A, mais uniquement du bas vers le haut, soit lors du passage par une cellule de type B (accompagné par un changement de direction). La circulation d'une donnée dans le réseau peut être intuitivement expliquée de la façon suivante (cf. figure 2) : la donnée entre par un bord inférieur ; elle circule alors de bas en haut en restant dans la même colonne ; pendant ce trajet, elle est d'abord passive, puis écrite : ceci correspond à la première phase de l'algorithme ; arrivée au bord supérieur, elle est transmise par une cellule D sur une diagonale descendante jusqu'au bord inférieur opposé ; pendant cette phase elle est lue et sa valeur utilisée pour modifier d'autres cœfficients ; puis intervient la phase 2 de l'algorithme, à nouveau du bord inférieur au bord supérieur sur une même colonne ; puis une nouvelle phase de transmission et lecture en diagonale ; et enfin une dernière phase quasi identique à la première, sur la même colonne que la colonne d'origine : lors de cette dernière phase, la donnée est passive dans les cellules où elle était écrite initialement, et réciproquement. Les écritures effectuées lors de cette dernière phase correspondent à la phase 3 de l'algorithme. Les phases 1 et 3 de l'algorithme sont donc exécutées sur la même colonne, dans des ensembles de cellules complémentaires.

À la fin de cette dernière phase, les données sont sorties vers le haut du réseau en passant par les cellules B_S, D_S ou C.

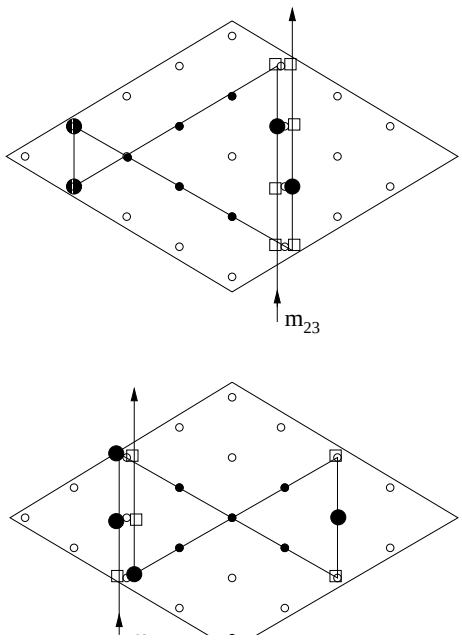


FIG. 2 – Chemin des données dans le réseau.

Question 14. *On suppose que l'instant $t = 1$ correspond au cycle d'horloge où le cœfficient $m_{1,1}$ entre dans la cellule $(z = 0, p = 1)$. En distinguant les cas $i < j$ et $i \geq j$, déterminer pour chaque cœfficient m_{ij} la date où il parvient*

1. *à l'entrée du réseau*
2. *en haut de sa première colonne (fin de la phase 1 de l'algorithme)*
3. *en bas de sa première diagonale*
4. *en haut de sa deuxième colonne (fin de la phase 2 de l'algorithme)*
5. *en bas de sa deuxième diagonale*
6. *à nouveau en haut de sa première colonne (un cycle avant sa sortie du réseau)*

Question 15. *À quel instant le cœfficient $m_{1,1}$ sort-il du réseau ? Même question pour le cœfficient $m_{n,n}$. Dans quel ordre de ses cœfficients et sous quelle forme la matrice résultat est-elle fournie en sortie ?*

Question 16. *Montrer qu'il ne peut pas y avoir de conflit transmission/écriture dans les cellules B_E, c'est-à-dire que les deux actions suivantes*

- *une donnée arrive par le bas (en entrée) et est transmise vers le haut,*
- *une donnée arrive par la diagonale, est écrite et transmise vers le haut,*

ne peuvent avoir lieu au même instant.

Question 17. *Même question pour les cellules de type D_E.*

Question 18. *Pour i et j données, et pour k compris entre 1 et N, montrer qu'il existe un instant et un cellule (et préciser lesquels) où le cœfficient m_{ij} rencontre les cœfficients m_{ik} et m_{kj}. On distinguera suivant les valeurs respectives de i et j et les phases de l'algorithme. On suppose que le mécanisme de contrôle des cellules est spécifié de telle façon quela donnée m_{ij} est mise à jour (écrite) à cet instant.*

Question 19. *En déduire — sous l'hypothèse que les mécanismes de contrôle sont corrects — que le réseau implante effectivement l'algorithm de la partie I.*

Question 20. *Si l'on considère qu'une cellule n'est active que lorsqu'elle écrit (modifie) une donnée, donner pour cahque instant entre le moment où la prmière donnée entre dans le réseau et celui où la dernière en sort, le nombre de cellules actives.*

Question 21. *Quel est le rendement (nombre de cellules actives / nombre total de cellules) moyen du réseau sur cet intervalle de temps ?*

Question 22. *Proposer un ou plusieurs moyens d'améliorer ce rendement.*