

RAIRO

INFORMATIQUE THÉORIQUE

O. COGIS

M. HABIB

Nombre de sauts et graphes série-parallèles

RAIRO – Informatique théorique, tome 13, n° 1 (1979), p. 3-18.

http://www.numdam.org/item?id=ITA_1979__13_1_3_0

© AFCET, 1979, tous droits réservés.

L'accès aux archives de la revue « RAIRO – Informatique théorique » implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/legal.php>). Toute utilisation commerciale ou impression systématique est constitutive d'une infraction pénale. Toute copie ou impression de ce fichier doit contenir la présente mention de copyright.

NUMDAM

*Article numérisé dans le cadre du programme
Numérisation de documents anciens mathématiques*

<http://www.numdam.org/>

NOMBRE DE SAUTS ET GRAPHES SÉRIE-PARALLÈLES (*)

O. COGIS et M. HABIB ⁽¹⁾

Communiqué par R. CORI

Résumé. — Le nombre de sauts d'un graphe sans circuit peut se définir comme le nombre minimal d'arcs à ajouter à ce graphe pour qu'il possède un chemin hamiltonien mais demeure sans circuit. Le problème de la détermination du nombre de sauts d'un graphe sans circuit est à ce jour non classé.

En partie 1, nous étudions une technique de décomposition compatible avec la détermination du nombre de sauts d'un graphe. En partie 2 nous donnons un résultat qui permet de réduire, dans certains cas, le choix des arcs à ajouter.

Enfin en partie 3, nous montrons que chacune de ces deux approches fournit un algorithme polynômial ($O(n^2)$) pour la recherche du nombre de sauts, lorsque le graphe considéré appartient à la classe des graphes série-parallèles.

INTRODUCTION

Soit $X = \{x_1, x_2, \dots, x_n\}$ un ensemble de tâches, et soit U l'ensemble des couples $x_i x_j$ tels que x_j ne puisse être exécutée qu'après x_i . Le graphe $G = (X, U)$ est appelé le graphe des contraintes de précédence (G est raisonnablement sans circuit). Si ces n tâches doivent être exécutées par une seule machine qui ne peut en exécuter qu'une à la fois, il faut déterminer un ordre τ d'exécution de ces n tâches. Ceci peut se faire en ajoutant des contraintes artificielles, non contradictoires avec les contraintes initiales, de telle sorte que seul demeure réalisable un ordre total τ (autrement dit, on détermine un ensemble V de couples de tâches tel que $G' = (X, U \cup V)$ soit sans circuit et hamiltonien).

On note $s(G, \tau)$ le nombre minimal de contraintes artificielles déterminant τ et $s(G)$ la valeur minimale prise par $s(G, \tau)$ lorsque τ décrit l'ensemble $\Omega(G)$ des ordres possibles d'exécution de ces n tâches.

Une illustration de ce problème est celui du choix de l'ordre dans lequel seront exposés les différents chapitres d'un cours, une fois ceux-ci déterminés par leur contenu. Certains chapitres ne pouvant être exposés qu'après certains autres, on

(*) Reçu mai 1977, révisé juillet 1978.

(1) Institut de Programmation, Université Pierre-et-Marie-Curie, Paris.

peut ainsi définir un graphe de précédence et, si τ est l'ordre dans lequel les chapitres seront exposés, $s(G, \tau)$ est le nombre de ruptures de lien logique qui apparaîtront au fil de l'exposé du cours. On préférera souvent choisir un ordre τ de coût minimal.

En 1972, M. Chein et P. Martin [3] ont défini l'invariant $s(G)$ ou nombre de sauts d'un graphe sans circuit, et ils ont proposé un algorithme en $O(n^{5/2})$ pour déterminer $s(G)$ lorsque le graphe est une forêt. Par la suite, M. Habib [6] a présenté un algorithme qui effectue la recherche du nombre de sauts d'un graphe qui est une forêt en $O(n^2)$, d'un graphe unicyclique en $O(n^3)$ et d'un graphe dont les cycles sont deux à deux disjoints en $O(n^4)$.

Enfin, dans le cas particulier des graphes bipartis, si le problème de la recherche du nombre de sauts n'a pu être résolu de façon polynômiale, G. Chaty, M. Chein, P. Martin, G. Petolla [1] en ont montré l'équivalence (polynômiale) avec le problème de la recherche d'un couplage sans cycle alterné et de cardinal maximal dans un graphe biparti (problème indépendamment posé par S. Krogdahl [8] à propos de changement de bases dans un matroïde).

Mais, le problème de la détermination du nombre de sauts d'un graphe sans circuit demeure à ce jour non classé, et la seule méthode générale connue énumère brutalement l'ensemble $\Omega(G)$, dont le cardinal peut croître exponentiellement en fonction du nombre de sommets de G .

C'est pourquoi nous présentons ici des résultats de deux types :

- réduction, quand c'est possible, du nombre d'opérations effectuées par l'algorithme brutal;
- exhibition d'une nouvelle famille de graphes pour laquelle il existe un algorithme polynômial résolvant le problème.

Deux approches distinctes ont conduit à des résultats du premier type. En partie 1, nous adaptions une idée de décomposition d'un graphe G en une famille $(G_i)_{i \in I}$ de sous-graphes disjoints deux à deux, idée introduite par A. Ducamp [4] en vue de la simplification de la recherche de la dimension d'un ordre partiel – cette décomposition est polynômiale et ramène le calcul de $s(G)$ à celui de $(s(G_i))_{i \in I}$.

En partie 2, nous montrons que $s(G)$ peut être calculé par énumération d'un sous-ensemble $\Omega^*(G)$ de $\Omega(G)$ et nous proposons un algorithme en $O(n^2)$ permettant de construire un élément de $\Omega^*(G)$.

Les résultats du premier type sont limités, car certains graphes sont indécomposables, et pour certains graphes G , on a $\Omega^*(G) = \Omega(G)$. A l'opposé l'étude des cas particuliers favorables, conduit curieusement à un même résultat du premier type : en partie 3, nous montrons que les graphes série-parallèles

introduits par E. L. Lawler et R. E. Tarjan [9], sont les graphes décomposables en sous-graphes réduits chacun à un sommet et que d'autre part, l'algorithme exposé en partie 2 calcule directement le nombre de sauts des graphes série-parallèles (nous signalons en remarque, que cet algorithme détermine, pour ces graphes, des ordres totaux de coût minimal pourvu que la fonction coût remplisse une condition donnée).

DÉFINITIONS : Dans tout ce qui suit, les graphes considérés seront finis, simples, orientés et, sauf mention particulière, sans circuit.

Soit $G=(X, U)$ un graphe – donc sans circuit – et soit $X = \{x_1, \dots, x_n\}$ l'ensemble de ses sommets. Enfin, soit τ un ordre total sur X , que nous convenons de noter comme un mot construit sur l'alphabet X , c'est-à-dire : $\tau = x_{\pi(1)} \dots x_{\pi(n)}$ où π est la permutation de l'ensemble $\{1, 2, \dots, n\}$, (que nous noterons $[1, n]$), telle que l'on ait : $x_{\pi(1)} < x_{\pi(2)} < \dots < x_{\pi(n)}$.

Nous appelons $\Omega(G)$ l'ensemble des ordres totaux sur X compatibles avec G (ou encore supérieurs à G) c'est-à-dire, en convenant de noter xy le couple $(x, y) : \tau \in \Omega(G)$ ssi $\forall x, y \in X, xy \in U \Rightarrow x$ précède y relativement à τ . G étant sans circuit, il est facile de se convaincre que $\Omega(G)$ est non vide.

Soit $\tau = x_{\pi(1)} \dots x_{\pi(n)} \in \Omega(G)$, nous appelons nombre de sauts de l'ordre τ par rapport au graphe G , le nombre noté $s(G, \tau)$ de couples $(x_{\pi(i)}, x_{\pi(i+1)})$ tels que $x_{\pi(i)} x_{\pi(i+1)} \notin U$ pour $i \in [1, n-1]$. Lorsque $n=1$, nous posons $s(G, \tau)=0$.

Nous appelons nombre de sauts du graphe G , le nombre noté $s(G)$ et défini par : $s(G) = \min_{\tau \in \Omega(G)} (s(G, \tau))$.

Enfin nous posons : $\tilde{\Omega}(G) = \{ \tau \in \Omega(G) / s(G, \tau) = s(G) \}$.

1. DÉCOMPOSITIONS COMPATIBLES AVEC LE NOMBRE DE SAUTS

1.1. Décomposition en joint

Nous noterons $G^t=(X, U^t)$ la fermeture transitive d'un graphe $G=(X, U)$.

PROPRIÉTÉ 1.1 [3] : $\Omega(G)=\Omega(G^t)$ et $\forall \tau \in \Omega(G), s(G, \tau)=s(G^t, \tau)$. En particulier, $\tilde{\Omega}(G)=\tilde{\Omega}(G^t)$ et $s(G)=s(G^t)$.

Autrement dit, le nombre de sauts est invariant pour une classe de graphes admettant la même fermeture transitive.

DEFINITION : Soient $H = (X_H, U_H)$ un graphe et $(G_i)_{i \in I} = ((X_i, U_i))_{i \in I}$ une famille de graphes vérifiant :

- (i) $\forall i, j \in I, i \neq j \Rightarrow X_i \cap X_j = \emptyset$;
- (ii) il existe une bijection σ de X_H dans I .

Nous appelons joint de la famille $(G_i)_{i \in I}$ par le graphe H , suivant la bijection σ , le graphe (X, U) noté $J_{H, \sigma}((G_i)_{i \in I})$ avec .

$$X = \bigcup_{i \in I} X_i \text{ et, } U = \bigcup_{i \in I} U_i \cup \left\{ \bigcup_{x_1 \in U_H} (X_{\sigma(x_1)} \times X_{\sigma(y)}) \right\}.$$

Cette opération se définit exactement de la même manière sur des graphes admettant des circuits; elle a été surtout étudiée jusqu'à présent dans le cas non orienté : J. L. Jolivet [7] et G. Sabidussi [12].

THÉORÈME 1 [6] : Si $G = J_{H, \sigma}((G_i)_{i \in I})$ et si $\tau_H = x_1, \dots, x_k \in \tilde{\Omega}(H)$, $\tau_i \in \tilde{\Omega}(G_i)$ pour $i \in I$, alors :

- (i) $\tau_{\sigma(x_1)}, \dots, \tau_{\sigma(x_k)} \in \tilde{\Omega}(G)$;
- (ii) $s(G) = s(H) + \sum_{i \in I} s(G_i)$.

On remarquera que dans le théorème précédent la valeur du nombre de sauts ne dépend pas du choix de la permutation σ sur I

Soit maintenant $\{G_i\}$ pour $i \in [1, k]$, l'ensemble des composantes connexes du graphe G . En remarquant que $G = J_{H, \sigma}((G_i)_{i \in [1, k]})$ où $H = ([1, k], \emptyset)$ et σ est une permutation quelconque de $[1, k]$, on tire immédiatement du théorème précédent :

COROLLAIRE 1.1 [3] : Si $\{G_i\}$ pour $i \in [1, k]$ est l'ensemble des composantes connexes de G , si σ est une permutation quelconque de $[1, k]$, et si $\tau_i \in \tilde{\Omega}(G_i)$ pour $i \in [1, k]$, alors :

- (i) $\tau = \tau_{\sigma(1)} \dots \tau_{\sigma(k)} \in \tilde{\Omega}(G)$;
- (ii) $s(G) = k - 1 + \sum_{i \in [1, k]} s(G_i)$.

DÉFINITION : Le graphe d'incomparabilité $G^I = (X, U^I)$ du graphe $G = (X, U)$ est donné par : $U^I = \{xy \in X^2 / \text{il n'existe aucun chemin dans } G \text{ de } x \text{ à } y \text{ ni de } y \text{ à } x\}$.

Le graphe G^I est manifestement symétrique. Notons T_k le graphe associé à l'ordre naturel strict sur l'ensemble $[1, k]$, [i.e. $T_k = ([1, k], \{ij/i, j \in [1, k], i < j\})$]. Le résultat suivant va nous permettre d'obtenir un résultat analogue au corollaire 1.1 [pour $Y \subseteq X$, on note $G(Y)$ le sous-graphe de G engendré par Y].

PROPRIÉTÉ 1.2 : Si $\{X_i^I\}$ pour $i \in [1, k]$ est la partition de X associée aux composantes connexes de G^I , il existe une permutation σ de $[1, k]$ telle que :

$$G^I = J_{\tau_k, \sigma}((G(X_i^I))_{i \in [1, k]}).$$

Démonstration : Nous avons,

1° si $p, q \in [1, k]$ avec $p \neq q$, alors : $\forall x \in X_p^I, \forall y \in X_q^I, xy \notin U^I \Rightarrow yx \in U^I$. Sinon, x et y seraient incomparables et l'on aurait $p = q$.

De plus B. Montjardet ([11], p. II, 18) a montré que les composantes connexes de G^I sont exactement les composantes fortement connexes de $G \cup G^I$.

Nous en tirons deux conséquences :

2° si $p, q \in [1, k]$ avec $p \neq q$, alors : $\forall x, x' \in X_p^I, \forall y, y' \in X_q^I, xy \in U^I \Rightarrow x'y' \in U^I$;

3° le graphe quotient $G/\{X_i^I\}_{i \in [1, k]}$ est sans circuit [où $G/\{X_i^I\}_{i \in [1, k]}$ est le graphe $([1, k], V)$ avec $V = \{ij/\exists x \in X_i^I, \exists y \in X_j^I, xy \in U\}$].

Le résultat annoncé découle immédiatement de 1°, 2° et 3°. ■

COROLLAIRE 1.2 : Si $\{X_i^I\}_{i \in [1, k]}$ est la partition de X associée aux composantes connexes de G^I , si σ est la permutation de $[1, k]$ dont la propriété 1.2 assure l'existence, et si $\tau_i \in \tilde{\Omega}(G(X_i^I)), \forall i \in [1, k]$ alors :

(i) $\tau = \tau_{\sigma(1)} \dots \tau_{\sigma(k)} \in \tilde{\Omega}(G)$;

(ii) $s(G) = \sum_{i \in [1, k]} s(G(X_i^I))$.

1.2. Décomposition de Ducamp

Dans le but de simplifier la détermination de la dimension d'un ordre partiel, A. Ducamp a eu l'idée de décomposer un ordre suivant ses composantes connexes puis celles-ci à leur tour suivant les composantes connexes du graphe d'incomparabilité, et d'itérer le processus tant que c'est possible.

Afin d'adapter cette méthode de décomposition à la détermination du nombre de sauts, à tout graphe $G = (X, U)$ nous associons une arborescence $A(G)$ dite de décomposition. Cette arborescence est plane, orientée (i.e. l'ensemble des successeurs de chaque sommet est muni d'un ordre total) et étiquetée aux sommets. Elle est définie par les règles suivantes :

(i) les sommets de $A(G)$ sont étiquetés par des sous-graphes de G ; de plus tout sommet x non pendent de $A(G)$ possède un poids entier positif ou nul $p(x)$;

(ii) la racine de $A(G)$ est étiquetée par G ;

(iii) soit x un sommet de $A(G)$ étiqueté par $G(Y)$:

(a) si $G(Y)$ n'est pas connexe, soient $G(Y_1), \dots, G(Y_k)$ ses composantes

connexes; x possède k successeurs x_1, \dots, x_k ordonnés dans un ordre quelconque et respectivement étiquetés $G(Y_1), \dots, G(Y_k)$; de plus $p(x) = k - 1$;

(b) si $G^l(Y)$ n'est pas connexe, soient $G^l(Y_1), \dots, G^l(Y_k)$ ses composantes connexes; x possède k successeurs x_1, \dots, x_k ordonnés suivant l'ordre total $x_{\sigma(1)}, \dots, x_{\sigma(k)}$ où σ est la permutation dont l'existence est assurée par la propriété 1.2, et respectivement étiquetés $G(Y_1), \dots, G(Y_k)$; de plus $p(x) = 0$;

(c) si $G(Y)$ et $G^l(Y)$ sont connexes, x est pendent.

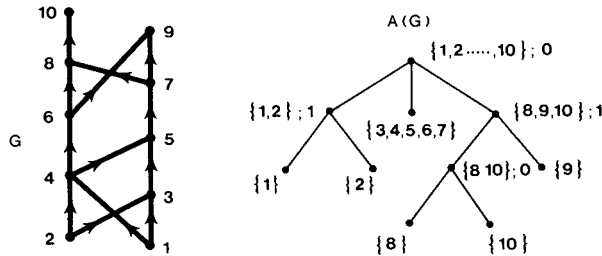


Figure 1.

La figure 1 fournit un exemple d'arborecence de décomposition où les sommets y sont étiquetés par Y au lieu de $G(Y)$, et les successeurs d'un sommet sont rangés par ordre croissant de gauche à droite.

Il est clair qu'il existe des graphes pour lesquels l'arborecence précédente est réduite à un seul sommet. Cependant, lorsque tel n'est pas le cas, le théorème suivant, qui se déduit facilement des corollaires 1.1 et 1.2, permet de simplifier la détermination du nombre de sauts.

On convient de définir un ordre total sur les sommets pendants d'une arborecence plane orientée de la façon suivante : si x_1 et x_2 sont pendants, soient x'_1 et x'_2 leur ascendants respectifs successeurs d'un même sommet z . x_1 et x_2 sont ordonnés comme x'_1 et x'_2 le sont dans l'ordre total associé aux successeurs de z ; autrement dit sur la figure 1, les sommets pendants sont rangés de gauche à droite par ordre croissant.

THÉORÈME 2 : Soit x_1, \dots, x_k la suite ordonnée des sommets pendants de $A(G)$ ayant respectivement pour étiquettes $G(X_1), \dots, G(X_k)$, et soit $\{y_1, \dots, y_h\}$ l'ensemble des sommets non pendants de $A(G)$. Alors, on a

$$s(G) = \sum_{j \in [1, h]} p(y_j) + \sum_{i \in [1, k]} s(G(X_i));$$

de plus, si $\tau_i \in \tilde{\Omega}(G(X_i))$ pour tout $i \in [1, k]$, $\tau = \tau_1, \dots, \tau_k \in \tilde{\Omega}(G)$.

Dans le meilleur des cas, les étiquettes $G(X_i)$ des sommets pendants de $A(G)$ sont des graphes tous réduits à un sommet et $A(G)$ suffit à déterminer $s(G)$ tout en fournissant un ordre $\tau \in \tilde{\Omega}(G)$. Nous disons alors que G est *totalemt décomposable*. Nous donnons en partie 3.1 une caractérisation des graphes totalement décomposables.

2. $\Omega^*(G)$, UN SOUS-ENSEMBLE DE $\Omega(G)$ SUFFISANT POUR LE NOMBRE DE SAUTS

2.1. Les ensembles $\mathcal{C}(G)$, $\mathcal{L}(G)$ et $\mathcal{S}(G)$

DÉFINITIONS : $\mathcal{C}(G)$ est l'ensemble des suites $C=(c_0, \dots, c_k)$ de chemins de G partitionnant le graphe au sens des sommets et telles que, $\forall i \in [0, k-1]$ l'extrémité du chemin c_i ne soit pas l'origine d'un arc de G ayant pour extrémité l'origine du chemin c_{i+1} .

Il existe une bijection canonique entre $\mathcal{C}(G)$ et l'ensemble des ordres totaux sur X . En effet, convenons de noter un chemin de G comme un mot sur l'alphabet X et convenons qu'un sommet constitue un chemin à lui tout seul. Il est clair que :

- si $C=(c_0, \dots, c_k) \in \mathcal{C}(G)$, alors $\tau_C=c_0, \dots, c_k$ est un ordre total sur X ;
- si τ est un ordre total sur X , il existe une décomposition unique de τ en sous-mots, i. e. $\tau=c_0, \dots, c_k$, telle que $\forall i \in [0, k]$, c_i soit un chemin de G et soit maximal pour cette propriété; alors $C_\tau=(c_0, \dots, c_k) \in \mathcal{C}(G)$.

$\mathcal{L}(G)$ est le sous-ensemble de $\mathcal{C}(G)$ canoniquement associé à $\Omega(G)$.

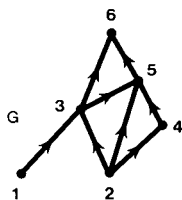


Figure 2.

G étant le graphe de la figure 2, on a

$$\tau = 123456 \in \Omega(G) \quad \text{et} \quad C_\tau = (1, 23, 456) \in \mathcal{L}(G),$$

$$C = (136, 245) \in \mathcal{C}(G) \quad \text{et} \quad \tau_C = 136245 \notin \Omega(G)$$

donc $C \notin \mathcal{L}(G)$.

Nous appelons *chemin libre de G* tout chemin tel que tout arc de G y ayant son extrémité, y ait aussi son origine [i. e. c tel que $\Gamma^-(c) \subseteq c$].

Pour $C=(c_0, \dots, c_k) \in \mathcal{C}(G)$, appelons X_i^C l'ensemble des sommets des chemins c_j , $\forall j \in [i, k]$ et posons $G_i^C = G(X_i^C)$. Nous pouvons alors caractériser aisément l'ensemble $\mathcal{L}(G)$ au moyen du résultat suivant :

PROPRIÉTÉ 2.1 : Soit $C=(c_0, \dots, c_k) \in \mathcal{C}(G)$, alors $C \in \mathcal{L}(G)$ ssi $\forall i \in [0, k]$, c_i est un chemin libre de G_i^C .

Chaque fois qu'il nous sera possible de le faire sans ambiguïté, nous écrirons X_i au lieu de X_i^C et G_i au lieu de G_i^C .

En reprenant l'exemple de la figure 2, on constate bien qu'en posant $C=(c_0, c_1)=(136,245)$, c_0 n'est pas libre dans $G_0 = G$, et donc $C \notin \mathcal{L}(G)$. Par contre c_1 est évidemment libre dans $G_1 = G(\{245\})$.

DÉFINITION : $\mathcal{S}(G)$ est le sous-ensemble de $\mathcal{L}(G)$ canoniquement associé à $\tilde{\Omega}(G)$.

Les éléments de $\mathcal{L}(G)$, respectivement $\mathcal{S}(G)$, s'obtenant par une simple réécriture des éléments de $\Omega(G)$, respectivement $\tilde{\Omega}(G)$, il est clair que lorsque $C=(c_0, \dots, c_k) \in \mathcal{C}(G)$:

- si $C \in \mathcal{L}(G)$, alors $s(G, \tau_C) = k$;
- si $C \in \mathcal{S}(G)$, alors $s(G) = k$.

Autrement dit, la détermination de $s(G)$ peut se ramener à celle d'un élément de $\mathcal{S}(G)$, c'est-à-dire d'un élément de $\mathcal{L}(G)$ comportant le minimum de chemins possible.

2.2. Les ensembles $\mathcal{E}(G)$ et $\Omega^*(G)$

Une stratégie de type « algorithme gourmand » pour déterminer un élément de $\mathcal{S}(G)$ consiste à choisir chaque c_i le plus long possible, i. e. à le choisir de telle sorte qu'il soit impossible de le prolonger sans qu'il perde sa propriété de liberté.

DÉFINITIONS : Nous appelons chemin libre maximal de G tout chemin libre de G qui ne soit pas sous-chemin strict d'aucun chemin libre de G . De plus $\mathcal{E}(G)$ est l'ensemble des suites $C=(c_0, \dots, c_k) \in \mathcal{L}(G)$ telles que, pour $\forall i \in [0, k]$, c_i soit un chemin libre maximal de G_i^C . Enfin $\Omega^*(G)$ est le sous-ensemble de $\Omega(G)$ canoniquement associé à $\mathcal{E}(G)$.

Reprenons l'exemple de la figure 1. On vérifie que $C_\tau=(1, 23, 456) \in \mathcal{E}(G)$. Malheureusement on a $s(G, \tau)=2$, alors que $s(G, \tau_{C'})=1$ avec $C'=(24, 1356)$ donc : $\mathcal{E}(G) \not\subseteq \mathcal{S}(G)$.

Ainsi la détermination de $s(G)$ ne peut se ramener à celle d'un élément quelconque de $\mathcal{L}(G)$. En fait, le rapport de $s(G, \tau_C)$ à $s(G)$, lorsque $C \in \mathcal{L}(G)$, n'est pas borné indépendamment de G , comme le montre l'exemple suivant :

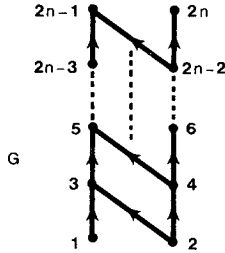


Figure 3.

$$C = (1, 23, 45, \dots, (2n-2)(2n-1), 2n) \in \mathcal{L}(G) \quad \text{et} \quad s(G, \tau_C) = n,$$

$$C' = (246 \dots (2n-2)2n, 135, \dots, (2n-3)(2n-1)) \in \mathcal{L}(G)$$

et

$$s(G, \tau_{C'}) = s(G) = 1.$$

Cependant, l'intérêt de l'ensemble $\mathcal{L}(G)$ réside dans le résultat suivant :

THÉORÈME 3 : $\tilde{\Omega}(G) \cap \Omega^*(G) \neq \emptyset$.

Il est clair que démontrer $\mathcal{L}(G) \cap \mathcal{L}(G) \neq \emptyset$ revient à démontrer ce théorème. Avant de le faire nous allons donner une caractérisation des éléments de $\mathcal{L}(G)$. Pour $C = (c_0, \dots, c_k) \in \mathcal{L}(G)$, notons respectivement o_i et e_i , l'origine et l'extrémité du chemin c_i .

DÉFINITION : On dit qu'une suite $C = (c_0, \dots, c_k) \in \mathcal{L}(G)$ vérifie la propriété P ssi $\forall i, j \in [0, k]$ tels que $i + 1 < j$ et $e_i, o_j \in U$, il existe $h \in]i, j[$ et $t \in c_h$ tels que $to_j \in U$.

PROPRIÉTÉ 2.1 : $\mathcal{L}(G)$ est l'ensemble des éléments de $\mathcal{L}(G)$ qui vérifient P.

Démonstration : Soit $C = (c_0, \dots, c_k) \in \mathcal{L}(G)$ vérifiant P. Rappelons que par définition de $\mathcal{L}(G)$ nous avons : $\forall i \in [0, k-1], e_i, o_{i+1} \notin U$. Ceci implique que si $k \in [1, 2]$, $C \in \mathcal{L}(G)$. De même si $k > 2$, c_{k-1} et c_k sont respectivement libres maximaux dans G_{k-1} et G_k . Considérons alors $i \in [0, k-2]$ et $x \in \Gamma_i^+(e_i)$ (nous notons respectivement Γ_i^+ et Γ_i^+ les fonctions successeur et prédécesseur dans G_i).

Nécessairement $x \in X_{i+1}$; posons $j \in [i+1, k]$ tel que $x \in c_j$. Si $x \neq o_j$, le chemin c_i, x ne peut être libre car $\Gamma_i^-(c_i, x)$ contient le prédécesseur y de x sur c_j , et $y \notin c_i, x$. Si $x = o_j$, d'après la propriété P, $\Gamma_i^-(c_i, x)$ contient un sommet $t \in c_h$ (avec $i < h < j$) et donc $t \notin c_i, x$ et le chemin c_i, x n'est pas libre. Le chemin c_i est donc libre maximal dans G_i et $C \in \mathcal{L}(G)$.

Considérons maintenant $C=(c_0, \dots, c_k) \in \mathcal{E}(G)$ et supposons qu'il existe $i, j \in [0, k]$ tels que $e_i, o_j \in U$. En vertu de la définition de $\mathcal{L}(G)$, $i+1 < j$ et c_i est libre dans G_i et en vertu de celle de $\mathcal{E}(G)$, c_i est libre maximal dans G_i ; il existe nécessairement $t \in X_{i+1}$ vérifiant $to_j \in U$ et $t \in c_h$ avec $i < h < j$. ■

Nous allons démontrer à présent le théorème 3. Soit $C=(c_0, \dots, c_k) \in \mathcal{L}(G) - \mathcal{E}(G)$. Alors il existe $i, j \in [0, k]$ tels que : $i+1 < j$, $e_i, o_j \in U$ et $\forall h \in [i+1, j-1], \forall t \in c_h, to_j \notin U$.

Nous construisons $C'=(c'_0, \dots, c'_k)$ à partir de C de la façon suivante :

- $\forall h \in [0, j-1], h \neq i, \quad c'_h = c_h \quad \text{et} \quad c'_i = c_i o_j;$
- si $c_j = o_j$, on pose $k' = k-1$ et si $j < k, c'_h = c_{h+1}, \forall h \in [j, k'];$
- sinon $c_j = o_j x_{j_1} \dots x_{j_q} e_j$, alors : si $e_{j-1} x_{j_1} \in U$ on pose de même

$$k' = k-1 \quad \text{et} \quad c'_{j-1} = c_{j-1} x_{j_1} \dots x_{j_q} e_j$$

et si $j < k, c'_h = c_{h+1}, \forall h \in [j, k'];$ sinon $e_{j-1} x_{j_1} \notin U$; on pose alors

$$k' = k \quad \text{et} \quad c'_j = x_{j_1} \dots x_{j_q} e_j$$

et si $j < k, c'_h = c_h, \forall h \in [j+1, k']$.

On vérifie aisément que dans tous ces différents cas, on a bien :

- c'_h est un chemin libre de $G_h^C, \forall h \in [0, k'];$
- $e'_h o_{h+1} \notin U, \forall h \in [0, k'-1]$.

Donc $C' \in \mathcal{L}(G)$ et $s(G, \tau_{C'}) = k' \leq k = s(G, \tau_C)$. Autrement dit, si une suite $C \in \mathcal{L}(G)$ ne vérifie pas P , on peut construire une suite $C' \in \mathcal{L}(G)$ telle que $s(G, \tau_{C'}) \leq s(G, \tau_C)$ et telle que $\eta(C') < \eta(C)$ si l'on note $\eta(C)$ l'entier naturel dont l'écriture en base n est : $\eta(C) = |c_k|, \dots, |c_0|$. Une telle construction n'est donc possible qu'un nombre fini de fois, ce qui montre que l'on peut construire, à partir d'une suite $C \in \mathcal{L}(G) - \mathcal{E}(G)$ une suite C^* vérifiant P , donc appartenant à $\mathcal{E}(G)$ et telle que $s(G, \tau_{C^*}) \leq s(G, \tau_C)$. ■

Note : L'exemple de la figure 3, nous montre que l'on peut avoir $\mathcal{E}(G) - \mathcal{L}(G) \neq \emptyset$, soit $\Omega^*(G) - \Omega(G) \neq \emptyset$. L'exemple de la figure 4, nous montre que l'on peut avoir $\mathcal{L}(G) - \mathcal{E}(G) \neq \emptyset$, soit $\Omega(G) - \Omega^*(G) \neq \emptyset$, car $C=(2, 135, 46) \in \mathcal{L}(G) - \mathcal{E}(G)$.

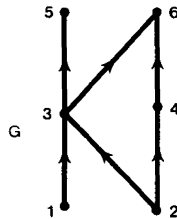


Figure 4.

THÉOREME 4 : $s(G) = \min_{\tau \in \Omega^*(G)} s(G, \tau)$.

Ce résultat est une conséquence immédiate du théorème 3. Ceci signifie en particulier que l'on peut bâtir une procédure arborescente qui n'énumère que les éléments de $\Omega^*(G)$. De plus il est possible de construire un élément de $\Omega^*(G)$ au moyen d'un algorithme polynômial borné en $O(n^2)$, que nous allons maintenant décrire.

2.3. Algorithme pour la construction d'un élément de $\Omega^*(G)$

(A) *Exposé de l'algorithme :*

Soit $G=(X, U)$ un graphe (sans circuit).

Γ^+ représente la fonction successeur du graphe G .

d^- représente la fonction demi-degré intérieur du graphe G .

λ représente le mot vide.

Le graphe G n'est pas modifié par l'algorithme.

Le langage utilisé pour la description de l'algorithme est un langage de type algol. Les étiquettes A, B et C ne sont utiles que pour l'analyse de l'algorithme et peuvent donc être omises pour le déroulement de ce dernier.

Début

```

pour x ∈ X faire ddi(x) ← d-(x);
k ← 0;
τ ← c0 ← λ;
Y ← X' ← X;
tant que X' ≠ ∅ faire
C : si il existe y ∈ Y tel que ddi(y) = 0
    alors A : début ck ← cky;
                X' ← X' - {y};
                Y ← Γ+(y);
                pour x ∈ Y faire ddi(x) ← ddi(x) - 1
            fin
    sinon B : début τ ← τ ck;
                k ← k + 1;
                ck ← λ;
                Y ← X'
            fin;
τ ← τ ck;
s ← k
fin
    
```

A la fin de l'algorithme, $\tau = (c_0, \dots, c_k) \in \Omega^*(G)$.

(B) *Évaluation de l'algorithme :*

Soit $|X| = n$:

Si l'on utilise une structure de liste pour représenter la fonction Γ^+ ainsi que les ensembles X, X' , et Y , il s'ensuit que :

– l'exécution de l'instruction composée A, ainsi que l'évaluation du test « il existe $y \in Y$ tel que $\text{ddi}(y) = 0$ », nécessitent chacune l'exécution d'un nombre d'instructions machine (LOAD, STORE, ADD . . .) majoré en $O(n)$;

– l'exécution de l'instruction composée B nécessitant l'exécution d'un nombre constant d'instructions machine, l'instruction conditionnelle composée C qui constitue le corps de la boucle « tant que » nécessite, pour s'exécuter, l'exécution d'un nombre d'instructions machine majoré en $O(n)$.

Remarquons maintenant, (en raisonnant par induction), qu'à chaque évaluation du test, chacune des trois propositions suivantes est vérifiée [en notant $d_{G(X')}^-$ la fonction demi-degré intérieur associée au graphe $G(X')$] :

- (i) $\forall x \in X' \text{ ddi}(x) = d_{G(X')}^-(x)$;
- (ii) $\forall x \in X' \Gamma^+(x) \subseteq X'$;
- (iii) $Y \subseteq X'$.

Il en découle que si l'exécution de l'instruction C, sous la condition $X' \neq \emptyset$, se ramène à celle de l'instruction B, X' demeurant inchangé, l'instruction C sera à nouveau exécutée, mais cette fois avec $Y = X'$. Du fait que $G(X')$ est sans circuit et que (i) est vérifiée, l'exécution de l'instruction C se ramènera cette fois nécessairement à celle de l'instruction A.

Comme au cours de l'instruction A, $|X'|$ diminue de 1, l'instruction C ne peut être exécutée un nombre de fois supérieur à $2n$ (elle l'est exactement un nombre de fois égal à $n+s$).

L'exécution de la boucle « tant que » nécessitera donc celle d'un nombre d'instructions machine majoré en $O(n^2)$ et on en déduit qu'il en va de même pour l'exécution du programme entier [on notera que l'implantation des données X et Γ^+ peut également s'effectuer au moyen d'un nombre d'instructions machine majoré en $O(n^2)$].

3. UN DOUBLE CAS PARTICULIER : LES GRAPHES SÉRIE-PARALLÈLES

Nous avons exposé en parties 1 et 2 des résultats obtenus en poursuivant deux approches *a priori* différentes. Assez curieusement, chacune des tentatives de simplification du problème de la détermination du nombre de sauts que l'on peut mettre en œuvre à partir de ces résultats est assurée du succès, (dans le sens où chacune fournit un algorithme polynômial) lorsqu'elle est appliquée à un graphe de la famille des graphes série-parallèles. Cette famille de graphes a été étudiée par Lawler et Tarjan [9] et on peut la définir comme suit :

Posons $H_v = (\{1, 2\}, \emptyset)$, $H_p = (\{1, 2\}, \{12\})$, et soit σ_0 l'identité sur $\{1, 2\}$. Nous appelons, respectivement opération de joint-vide et opération de joint-plein, les opérations suivantes :

$$\begin{aligned} (G_1, G_2) &\rightsquigarrow J_{H_v, \sigma_0}(G_1, G_2) \text{ noté } J_v(G_1, G_2), \\ (G_1, G_2) &\rightsquigarrow J_{H_p, \sigma_0}(G_1, G_2) \text{ noté } J_p(G_1, G_2). \end{aligned}$$

Soit X un ensemble fini et soit $\mathcal{G}_0(X)$ l'ensemble des graphes constitués d'un sommet isolé appartenant à X [i. e. $\mathcal{G}_0(X) = \{(\{x\}, \emptyset) / x \in X\}$]. Appelons $\mathcal{G}(X)$ l'ensemble des graphes, avec ou sans circuit, dont les sommets appartiennent à X [i. e. $\mathcal{G}(X) = \{(Y, V) / Y \subseteq X \text{ et } V \subseteq Y^2\}$].

DÉFINITIONS : *Un graphe $G = (X, U)$ est un graphe série-parallèle-transitif, en abrégé SPT, ssi il appartient à la fermeture de $\mathcal{G}_0(X)$ dans $\mathcal{G}(X)$ relativement aux opérations de joint-vide et de joint-plein. De plus un graphe est série-parallèle, en abrégé SP, ssi sa fermeture transitive est SPT.*

Il est facile de vérifier que tout graphe SP est sans circuit. Si nous appelons « enne » tout graphe isomorphe au graphe $(\{1, 2, 3, 4\}, \{13, 14, 24\})$, nous pouvons mentionner une caractérisation des graphes SPT par sous-graphes exclus :

THÉORÈME 5 [9] : *Un graphe transitif est SPT ssi il n'admet aucun « enne » comme sous-graphe.*

Cette caractérisation permet notamment d'établir que les graphes série-parallèles-électriques étudiés par Elgott et Wright dans [5] sont SP.

3.1. Les graphes totalement décomposables de Ducamp

THÉORÈME 6 : *G est un graphe SP ssi il est totalement décomposable.*

Démonstration : En remarquant qu'un graphe est totalement décomposable ssi sa fermeture transitive l'est, nous pouvons supposer le graphe G transitif :

1° soit G un graphe SPT et $A(G)$ son arborescence de décomposition. Soit x un sommet de $A(G)$ étiqueté par un sous-graphe $G(Y)$ non réduit à un sommet. On peut supposer par induction que $G(Y)$ est également SPT. Il existe donc deux graphes SPT G_1, G_2 tels que $G(Y) = J_v(G_1, G_2)$ ou $G(Y) = J_p(G_1, G_2)$. Donc x n'est pas un sommet pendant de $A(G)$ et de plus, chacun de ses successeurs est étiqueté par un graphe SPT (on peut utiliser le théorème 5). G est donc totalement décomposable;

2° soit G un graphe transitif non SPT. Il admet alors un « enne » comme sous-graphe (cf. th. 5) et cet « enne » est entièrement inclus dans une composante connexe de G et dans une composante connexe de G^t . On prouve alors aisément par induction que G n'est pas totalement décomposable.

3.2. Algorithme gourmand pour les graphes série-parallèles

L'algorithme exposé en partie 2.2 peut s'appliquer brutalement aux graphes SP pour en déterminer le nombre de sauts, comme le montre le résultat suivant :

THÉORÈME 7 : *Si G est un graphe SP, alors $\tilde{\Omega}(G) = \Omega^*(G)$.*

Nous allons montrer ce résultat sous une forme équivalente : si G est SP alors $\mathcal{E}(G) = \mathcal{S}(G)$. Notons d'abord que la réciproque est fautive comme le montre l'exemple de la figure 5 :

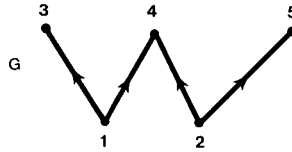


Figure 5.

Le graphe G de la figure 5 n'est pas un graphe SP, car il admet un « enne » comme sous-graphe. Cependant $\forall C \in \mathcal{E}(G), C = (c_0, \dots, c_k) \Rightarrow k = 2$ et manifestement $s(G) = 2$. Alors $\mathcal{S}(G) = \mathcal{E}(G)$, [puisque $\mathcal{E}(G) \subseteq \mathcal{S}(G)$ et que G est biparti].

Établissons quelques résultats avant de démontrer le théorème 7.

Soit $G = (X, U)$ un graphe et $c = x_1 \dots x_q$ un chemin libre de G , alors, $\forall i \in [1, q - 1], x_i x_{i+1}$ n'est pas un arc de transitivité de G (i. e. il n'existe pas d'autre chemin dans G que l'arc $x_i x_{i+1}$ reliant x_i à x_{i+1}). De plus c est libre dans G ssi c est libre dans G' . Le résultat suivant découle immédiatement de ces remarques.

LEMME 1 : *Soit G un graphe, alors $\mathcal{L}(G) = \mathcal{L}(G'), \mathcal{S}(G) = \mathcal{S}(G')$ et $\mathcal{E}(G) = \mathcal{E}(G')$.*

Nous pouvons donc supposer, dans la démonstration du théorème 7, que le graphe G est SPT. Ce faisant, soit $C = (c_0, \dots, c_k) \in \mathcal{L}(G)$ et plaçons-nous tout d'abord dans le cas où $G = J_v(G_1, G_2)$ où G_1 et G_2 sont deux graphes disjoints au sens des sommets. Pour tout $i \in [0, k] c_i$ est soit un chemin de G_1 , soit un chemin de G_2 . Appelons I_1 et I_2 les sous-suites de la suite $(0, 1, \dots, k)$ constituées respectivement des entiers i tels que c_i soit un chemin de G_1 et des entiers i tels que c_i soit un chemin de G_2 .

On pose

$$\alpha(C) = (c_i)_{i \in I_1} \quad \text{et} \quad \beta(C) = (c_i)_{i \in I_2}.$$

Considérons maintenant le cas où $G = J_p(G_1, G_2)$ où G_1 et G_2 sont deux graphes disjoints au sens des sommets. Soient $x \in G_1, y \in G_2, i, j \in [0, k]$ tels que : $x \in c_i$ et $y \in c_j$. Nécessairement $i \leq j$ (car $xy \in U$) et de plus deux cas sont à considérer :

- 1° $i=j$, alors x est un antécédent de y sur le chemin $c_i=c_j$;
- 2° $i \neq j$ et si $x = e_i$, alors $y \neq o_{i+1}$.

Par conséquent, il existe $i_m \in [0, k]$ tel que :

- si $i_m > 0, c_0, \dots, c_{i_m-1}$ sont des chemins de G_1 ;
- si $i_m < k, c_{i_m+1}, \dots, c_k$ sont des chemins de G_2 .

De plus, en posant $c_{i_m} = x_1, \dots, x_q$ il existe $j_m \in [2, q]$ tel que : $c' = (x_1, \dots, x_{j_m-1})$ soit un chemin de G_1 et $c'' = (x_{j_m}, \dots, x_q)$ soit un chemin de G_2 . On définit alors :

$$\alpha(C) = (c_0, \dots, c_{i_m-1}, c') \quad \text{ou} \quad \alpha(C) = (c') \quad \text{lorsque } i_m = 0,$$

et

$$\beta(C) = (c'', c_{i_m+1}, \dots, c_k) \quad \text{ou} \quad \beta(C) = (c'') \quad \text{lorsque } i_m = k.$$

Il est maintenant aisé de vérifier dans les deux cas, que pour $C \in \mathcal{L}(G)$ on a : $\alpha(C) \in \mathcal{L}(G_1)$ et $\beta(C) \in \mathcal{L}(G_2)$. De plus, si $C_1 \in \mathcal{L}(G_1)$ et $C_2 \in \mathcal{L}(G_2)$, il existe $C \in \mathcal{L}(G)$ tel que $C_1 = \alpha(C)$ et $C_2 = \beta(C)$ [il y a unicité de C dans le cas où, $G = J_p(G_1, G_2)$].

On en conclut alors facilement le résultat suivant :

LEMME 2 : Soient G_1 et G_2 deux graphes disjoints au sens des sommets, et $G = J_i(G_1, G_2)$ où $i \in \{v, p\}$. Soit $C \in (G)$ on a alors :

- (i) $C \in \mathcal{E}(G) \Leftrightarrow \alpha(C) \in \mathcal{E}(G_1)$ et $\beta(C) \in \mathcal{E}(G_2)$;
- (ii) $C \in \mathcal{S}(G) \Leftrightarrow \alpha(C) \in \mathcal{S}(G_1)$ et $\beta(C) \in \mathcal{S}(G_2)$;
- (iii) $s(G) = s(G_1) + s(G_2) + \varepsilon$, avec $\varepsilon = 1$ si $i = v$ et $\varepsilon = 0$ si $i = p$.

Nous pouvons maintenant aborder la démonstration du théorème 7 en raisonnant par induction sur le nombre de sommets d'un graphe G SPT.

Si $n = 1$, alors évidemment $\mathcal{E}(G) = \mathcal{S}(G)$.

Supposons, $n > 1$, il existe alors deux graphes SPT, G_1 et G_2 disjoints au sens des sommets et tels que : $G = J_1(G_1, G_2)$ avec $i \in \{v, p\}$.

Comme G_1 et G_2 ont respectivement moins de sommets, au sens strict, que G , nous obtenons :

$$C \in \mathcal{S}(G) \Leftrightarrow \left\{ \begin{array}{l} \alpha(C) \in \mathcal{S}(G_1) \Leftrightarrow \alpha(C) \in \mathcal{E}(G_1) \\ \beta(C) \in \mathcal{S}(G_2) \Leftrightarrow \beta(C) \in \mathcal{E}(G_2) \end{array} \right\} \Leftrightarrow C \in \mathcal{E}(G). \quad \blacksquare$$

REMARQUE : Signalons enfin que le résultat de ce théorème 7 peut s'étendre à d'autres fonctions coût définies sur les ordres totaux compatibles avec un graphe sans circuit.

Soit G un tel graphe et soit $t : \Omega(G) \rightarrow \mathbf{N}$, une telle fonction coût. Posons, $t(G) = \min \{ t(G, \tau) / \tau \in \Omega(G) \}$, $\Omega'(G) = \{ \tau \in \Omega(G) / t(G, \tau) = t(G) \}$ et enfin $\mathcal{F}(G) = \{ C \in (G) / \tau_C \in \Omega'(G) \}$.

Si le résultat (ii) du lemme 2 est encore vérifié, i. e. si l'on a

$$C \in \mathcal{F}(G) \Leftrightarrow \alpha(C) \in \mathcal{F}(G_1) \quad \text{et} \quad \beta(C) \in \mathcal{F}(G_2),$$

le résultat suivant est également vrai :

THÉORÈME 7 bis : Si G est un graphe SP, alors $\Omega'(G) = \Omega^*(G)$.

BIBLIOGRAPHIE

1. G. CHATY, M. CHEIN, P. MARTIN et G. PETOLLA, *Some Results About the Number of Jumps of an Acircuit Digraph*, Proc. 5th. S-E Conf. Combinatorics, Graph Theory and Computing, p. 267-279, Winnipeg, Utilitas Math., 1974.
2. G. CHATY, M. CHEIN, P. MARTIN et G. PETOLLA, *Number of Jumps and Ordered Matchings in Bipartite Digraphs* (à paraître).
3. M. CHEIN et P. MARTIN, *Sur le nombre de sauts d'une forêt*, C. R. Acad. Sc. Paris, t. 275, série A, 1972, p. 159-161.
4. A. DUCAMP, *Sur la dimension d'un ordre partiel*, Théorie des graphes, Journées internationales d'étude, Rome, 1967, Dunod, Paris, p. 103-112.
5. C. C. ELGOT et J. E. WRIGHT, *Series-Parallel Graphs and Lattices*, Duke Math. J., vol. 26, 1959, p. 325-338.
6. M. HABIB, *Partitions en chemins des sommets et sauts dans les graphes sans circuit*, Thèse de 3^e cycle, Université Paris-VI, 1975.
7. J. L. JOLIVET, *Problèmes de connexité et problèmes hamiltoniens en théorie des graphes. Joint d'une famille de graphes et généralisation de la notion de graphe parfait*, Thèse Sc. Math., Université Paris-VI, 1975.
8. S. KROGDALH, *The Dependance Graph for Basis in Matroids*, Discrete Math., vol. 19, 1977, p. 47-59.
9. E. L. LAWLER et R. E. TARJAN, *Analysis and Isomorphism of Serie-Parallel Digraph* (à paraître).
10. E. L. LAWLER, *Sequencing Jobs to Minimize Total Weighted Completion Time Subject to Precedence Constraints*, Problèmes Combinatoires et Théorie des Graphes, Orsay, 1976.
11. B. MONTJARDET, *Problèmes de transversalité dans les hypergraphes, les ensembles ordonnés et en théorie de la décision collective*, Thèse Sc. Math., Université Paris-VI, 1974.
12. G. SABIDUSSI, *Graph Derivative*, Math. Zeitschr., vol. 76, 1961, p. 385-401.