

JEAN-LOUP BAER

**Matrice de connexion minimale d'une matrice  
de précedence donnée**

*Revue française d'informatique et de recherche opérationnelle. Série rouge, tome 3, n° R1 (1969), p. 65-73*

[http://www.numdam.org/item?id=M2AN\\_1969\\_\\_3\\_1\\_65\\_0](http://www.numdam.org/item?id=M2AN_1969__3_1_65_0)

© AFCET, 1969, tous droits réservés.

L'accès aux archives de la revue « Revue française d'informatique et de recherche opérationnelle. Série rouge » implique l'accord avec les conditions générales d'utilisation (<http://www.numdam.org/conditions>). 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/>

## MATRICE DE CONNEXION MINIMALE D'UNE MATRICE DE PRECEDENCE DONNEE.

par Jean-Loup BAER (1)

---

Résumé. — *Un algorithme pour générer une matrice de connexion minimale d'une matrice de précédence donnée est présenté. Il se décompose en quatre phases : détermination des sous-graphes fortement connexes maximum ; compression de la matrice de précédence en une matrice pour un graphe sans circuit ; détermination de la matrice de connexion minimale du graphe sans circuit ; extension de la matrice minimale.*

### INTRODUCTION

Étant donné un graphe orienté  $G(X, U)$  où  $X$  est l'ensemble des sommets ( $X$  a  $n$  éléments) et  $U$  l'ensemble des arcs de  $G$ , on peut lui associer une matrice booléenne carrée  $Z(n, n)$  dite *matrice de connexion* de  $G$  (ou encore matrice associée à  $G$ ). Cette matrice  $Z$  est telle que  $z_i^j = 1$  ( $i$  représentant l'indice des lignes et  $j$  celui des colonnes) si et seulement si l'arc  $u_k = (x_j, x_i)$  appartient à  $U$ . Dans le cas contraire,  $z_i^j = 0$ . Le problème de trouver la matrice booléenne carrée  $D(n, n)$  définie par

$$(1) \quad D = \bigcup_{k=1}^{\infty} Z^k = \bigcup_{k=1}^n Z^k,$$

dite *matrice de précédence* de  $G$  a été traité par de nombreux auteurs (un problème équivalent est celui de déterminer la fermeture transitive de  $G$ ). Une comparaison de ces différents algorithmes (2) montre que celui de Warshall (6) est le plus puissant.

Dans cet article, nous résolvons le problème inverse. Étant donné la

---

(1) Université de Californie à Los Angeles. Ce travail a été réalisé dans le cadre du projet AT(11-1) Gen. 10, Proj. 14 de l'Atomic Energy Commission.

matrice  $D$ , trouver une matrice  $Z$  telle que la relation (1) ci-dessus soit vraie. Notons qu'une solution triviale peut être obtenue en prenant  $Z = D$ . Aussi chercherons-nous une matrice  $Z$  minimale, c'est-à-dire telle que le nombre de ses éléments non nul est minimum. En général il existe plusieurs solutions qui, comme nous le verrons, peuvent être déduites les unes des autres très facilement. La seule restriction que nous imposons est que  $D$  soit transitive, c'est-à-dire que si  $d_i^j = 1$  et  $d_j^k = 1$  alors  $d_i^k = 1$ . Une solution a été donnée par Simoes-Pereira [5]. Cependant l'algorithme décrit dans cette référence peut être considérablement simplifié en utilisant la théorie des graphes.

Nous suivrons les notations de Berge [1] et Roy [4] et en particulier nous dirons que :

Un *précédent* de  $x_i$  est un sommet  $x_j$ ,  $x_j \in X$ , tel que l'arc  $(x_j, x_i) \in U$ .

Un *ascendant véritable* de  $x_i$  est un sommet  $x_j$ ,  $x_j \in X$ , tel qu'il existe un chemin allant de  $x_i$  à  $x_j$ . Les *ascendants* de  $x_i$  sont ses ascendants véritables et  $x_i$  lui-même.

On définit de la même façon *suivant*, *descendant véritable* et *descendant* d'un sommet.

Notons alors qu'une ligne  $Z_i$  de  $Z$  est la représentation sous forme de vecteur booléen des précédents de  $x_i$  et une colonne  $Z^j$  celle des suivants de  $x_j$ . De même  $D_i$  représente les ascendants véritables de  $x_i$  et  $D^j$  les descendants véritables de  $x_j$ . Imposer  $D$  transitive est donc simplement imposer que  $D$  soit une matrice de précedence d'un graphe.

On sait qu'un *sous-graphe* de  $G(X, U)$  est un graphe  $H(Y, V)$  où  $Y \subset X$  et  $V$  est l'ensemble des arcs de  $U$  ayant leur deux extrémités dans  $Y$ . Un sous-graphe est *fortement connexe* si deux sommets distincts admettent toujours au moins un ascendant et un descendant communs. Étant donné  $x_i \in X$ , un sous-graphe fortement connexe  $H(Y, V)$  de  $G(X, U)$  est dit *sous-graphe fortement connexe maximum* (S.G.F.C.M.) contenant  $x_i$  si  $x_i \in Y$  et s'il n'existe aucun autre sous-graphe fortement connexe  $H(Y', V')$ ,  $x_i \in Y'$ , tel que le nombre d'éléments de  $Y'$ , soit supérieur au nombre d'éléments de  $Y$ . Pour un  $x_i$  donné, le S.G.F.C.M.  $H(Y, V)$ ,  $x_i \in Y$ , est unique [3].

Avant de donner l'algorithme générant  $Z$ , nous résolvons les trois problèmes suivants :

1. Détermination des S.G.F.C.M.
2. Détermination de  $Z$  minimale pour un graphe fortement connexe.
3. Détermination de  $Z$  minimale pour un graphe sans circuit.

### Algorithme 1. Détermination des S.G.F.C.M.

Étant donné la matrice de précedence  $D$  représentative d'une classe de graphes  $G(X, U)$  (en effet si pour  $D$  donnée  $X$  est unique,  $U$  ne l'est pas néces-

sairement cf. fig. 1), les S.G.F.C.M. peuvent être déterminés de la façon suivante (nous appliquons ici une variante de la méthode donnée en [3]):

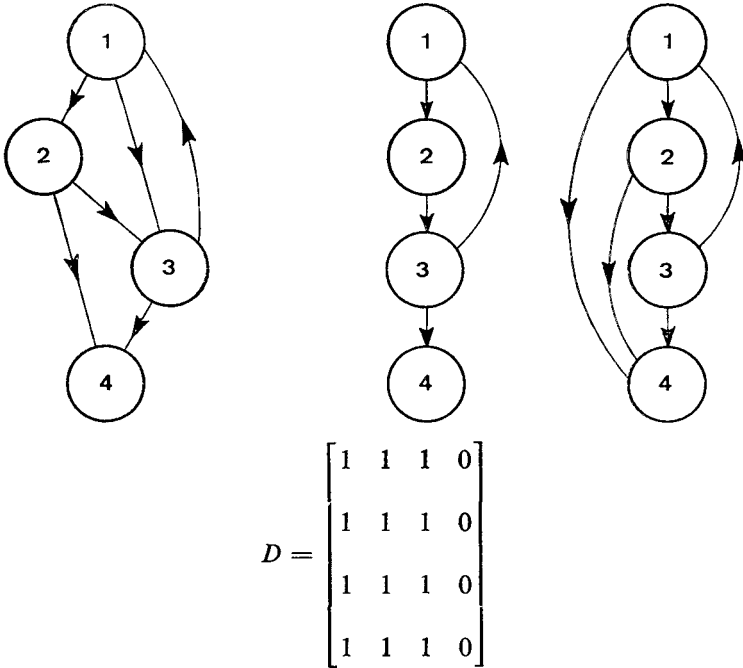


Figure 1

Les 3 graphes ont même matrice de précédence  $D$ .

Nous considérons tout d'abord l'ensemble des sommets qui n'appartiennent à aucun circuit. Si  $x_i$  est l'un d'eux, alors  $d_i^i = 0$ . Soit  $C_0$  l'ensemble de ces sommets. Chaque élément de  $C_0$  est lui-même un S.G.F.C.M.

Considérons maintenant un sommet  $x_i$  tel que  $d_i^i = 1$ . Alors  $D_i$  représente ses ascendants véritables et  $D^i$  ses descendants véritables. L'ensemble  $C_k$  dont la représentation booléenne est  $D_i \cap D^i$  est donc l'ensemble des sommets qui sont ascendants et descendants véritables de  $x_i$ . C'est donc un sous-graphe fortement connexe de  $G(X, U)$ . Il est maximum car si  $x_j \notin C_k$  était un ascendant véritable de  $x_i$ , il appartiendrait à  $D_i$  et s'il était aussi un descendant véritable, il appartiendrait à  $D^i$  donc à leur intersection  $C_k$  ce qui est contraire à l'hypothèse.

L'algorithme est alors:

1. Soit  $V = \{ 1, 1, \dots, 1 \}$  le vecteur unitaire de dimension  $n$  et

$$P = \{ d_1^1, d_2^2, \dots, d_n^n \}$$

le vecteur booléen représentant la diagonale de  $D$ .

2.  $C_0 = \overline{P \cap V}$ . Soit  $n_0$  le nombre d'éléments non-nuls de  $C_0$ .

Si  $n_0 = n$  aller à 5 sinon  $m = 1$ .

3. Soit  $d_i^1$  le premier élément non nul de  $P$ .

$C_m = D_i \cap D^i$  et soit  $n_m$  le nombre de ses éléments non-nuls.  $P = P \cap \overline{C_m}$ .

4. Si  $n_0 + \sum_{k=1}^m n_k = n$  aller à 5 sinon  $m = m + 1$  et aller à 3.

5. Fin.

### Algorithme 2. Détermination de $Z$ minimale pour un graphe fortement connexe

Soit  $G(X, U)$  un graphe fortement connexe. Sa matrice booléenne de précédence  $D$  est pleine puisque par définition tous les éléments de  $X$  sont ascendants et descendants véritables d'un quelconque des éléments de  $X$ . Nous sommes intéressés par le graphe  $G'(X, U')$  de matrice de précédence  $D$  tel que le nombre d'éléments de  $U'$  soit minimum. Puisque le graphe est fortement connexe, ce nombre est nécessairement  $n$  (nombre de sommets du graphe). Si  $X = \{x_1, \dots, x_n\}$ , est l'ensemble des sommets de  $G$  (et de  $G'$ ) alors une définition possible de  $U'$  est :

$$U' = \{(x_1, x_2), (x_2, x_3), \dots, (x_{n-1}, x_n), (x_n, x_1)\}$$

c'est-à-dire que le graphe  $G'(X, U')$  peut être considéré comme un circuit hamiltonien de  $G(X, U)$ . Notons qu'il existe  $n!$  définitions équivalentes de  $U'$  correspondants aux permutations des indices des sommets et donc  $n!$  possibles  $Z$ .

En résumé, si  $D$  est pleine, une matrice  $Z$  minimale peut être définie par :

$$\begin{cases} z_i^j = 1 \text{ si } i = j + 1 (j = 1, 2, \dots, n - 1 ; i = 2, 3, \dots, n) \\ z_1^n = 1 \\ z_i^j = 0 \text{ dans tous les autres cas.} \end{cases}$$

### Algorithme 3. Détermination de $Z$ minimale pour un graphe sans circuit

Soit  $G(X, U)$  un graphe sans circuit. Sa matrice de précédence  $D$  est telle que  $d_i^i = 0, \forall i$ . Ici encore nous voulons trouver  $G'(X, U')$  tel que le nombre d'éléments de  $U'$  soit minimum. Étant donné  $x_i$ , tous ses ascendants véritables sont représentés par  $D_i$ . Soit  $x_j (j = 1, 2, \dots, k)$  les éléments de  $D_i$ . Tout ascendant véritable de  $x_j$  est également un ascendant véritable de  $x_i$ , donc lui-même un élément de  $D_i$ . Pour former  $Z_i$  nous ne voulons garder que les éléments  $x_j$  tels qu'il existe un chemin et un seul allant de  $x_j$  à  $x_i$ , ce chemin devant être de longueur 1. Il suffit donc de supprimer de  $D_i$  tous les éléments  $x_k$  qui sont

eux-mêmes ascendants véritables d'au moins un élément  $x_j$  de  $D_i$ , c'est-à-dire :

$$Z_i = D_i \cap \left( \bigcup_{\{k | x_k \in D_i\}} D_k \right)$$

Nous pouvons maintenant passer à l'algorithme de génération de  $Z$  minimale,  $D$  étant donnée.

**Génération d'une matrice de connexion minimale**

Cet algorithme se décompose en quatre phases.

**Phase 1.** Détermination des S.G.F.C.M. à partir de  $D$  (cf. algorithme 1).

**Phase 2.** Compression de  $D$  en  $D^*$ .

Puisque tous les éléments d'un S.G.F.C.M. ont même ascendants et descendants, nous comprimons chaque S.G.F.C.M. en un seul sommet, par exemple le premier, et supprimons la boucle que ce sommet forme avec lui-même. De ce fait, la matrice comprimée  $D^*$  représente un graphe sans circuit. Pratiquement cette suppression revient à annuler certaines lignes et colonnes de la façon suivante :

1. Soit  $x_\alpha, x_\beta \dots$  les premiers sommets des S.G.F.C.M. dont les représentations sont  $C_1, C_2, \dots$ .

Annuler dans  $D$  toutes lignes et colonnes dont les indices correspondent aux sommets définis par :

$$x_i \notin C_0 \quad ; \quad x_i \neq x_\alpha \quad , \quad x_i \neq x_\beta, \text{ etc.}$$

2.  $d_i^* = 0 \forall i$ .

**Phase 3.** Génération de  $Z^*$  depuis  $D^*$ .

$D^*$  étant la matrice de précédence d'un graphe sans circuit,  $Z^*$  est générée par la méthode décrite plus haut (cf. algorithme 3).

**Phase 4.** Extension de  $Z^*$  en  $Z$ .

Pour chaque élément  $x_\alpha$  correspondant au S.G.F.C.M. dont la représentation est  $C_m = \{x_\alpha, x_{j_1}, \dots, x_{j_k}\}$  on forme  $Z$  comme décrit plus haut (cf. algorithme 2).

**EXEMPLE**

A titre d'exemple, appliquons la méthode à la matrice qu'on trouvera à la page suivante :

**Phase 1.** — Détermination des S.G.F.C.M. Il vient :

$$\begin{aligned} C_0 &= \{x_1, x_7, x_8, x_{11}\} \\ C_1 &= \{x_2, x_3, x_4\} \\ C_2 &= \{x_5, x_6\} \\ C_3 &= \{x_9, x_{10}\} \end{aligned}$$

	1	2	3	4	5	6	7	8	9	10	11
1	0	0	0	0	0	0	0	0	0	0	0
2	1	1	1	1	0	0	0	0	0	0	0
3	1	1	1	1	0	0	0	0	0	0	0
4	1	1	1	1	0	0	0	0	0	0	0
5	1	0	0	0	1	1	0	0	0	0	0
6	1	0	0	0	1	1	0	0	0	0	0
7	1	1	1	1	1	1	0	0	0	0	0
8	1	1	1	1	1	1	1	0	0	0	0
9	1	1	1	1	1	1	1	0	1	1	0
10	1	1	1	1	1	1	1	0	1	1	0
11	1	1	1	1	1	1	1	1	1	1	0

**Phase 2.** — Compression de  $D$  en  $D^*$ .

	1	$\alpha$ 2	3	4	$\beta$ 5	6	7	8	$\gamma$ 9	10	11
1	0	0			0		0	0	0		0
$\alpha$ 2	1	0			0		0	0	0		0
3											
4											
$\beta$ 5	1	0			0		0	0	0		0
$D^*$ 6											
7	1	1			1		0	0	0		0
8	1	1			1		1	0	0		0
$\gamma$ 9	1	1			1		1	0	0		0
10											
11	1	1			1		1	1	1		0

Phase 3. — Génération de  $Z^*$ .

	1	2	3	4	5	6	7	8	9	10	11
1	0	0			0		0	0	0		0
2	1	0			0		0	0	0		0
3											
4											
5	1	0			0		0	0	0		0
6											
7	0	1			1		0	0	0		0
8	0	0			0		1	0	0		0
9	0	0			0		1	0	0		0
10											
11	0	0			0		0	1	1		0

ce qui correspond au graphe de la figure 2.

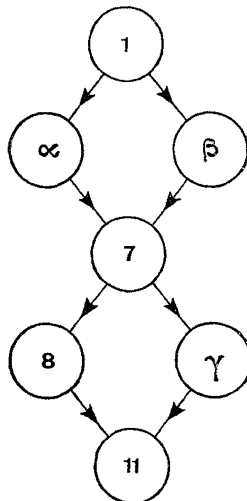


Figure 2  
Graphe correspondant à  $Z^*$ .

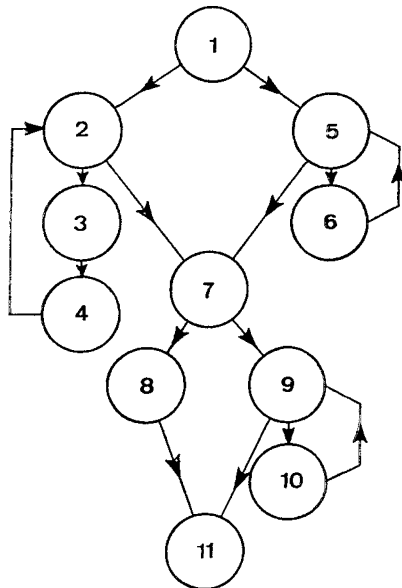


**Phase 4.** — Extension de  $Z^*$  en  $Z$ .

$Z =$

	1	2	3	4	5	6	7	8	9	10	11
1	0	0	0	0	0	0	0	0	0	0	0
2	1	0	0	1	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0	0	0	0
5	1	0	0	0	0	1	0	0	0	0	0
6	0	0	0	0	1	0	0	0	0	0	0
7	0	1	0	0	1	0	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0	0	0
9	0	0	0	0	0	0	1	0	0	1	0
10	0	0	0	0	0	0	0	0	1	0	0
11	0	0	0	0	0	0	0	1	1	0	0

ce qui correspond au graphe de la figure 3.



**Figure 3**  
Graphe correspondant à  $Z$ .

## CONCLUSION

Nous avons présenté un algorithme efficace pour la génération d'une matrice de connexion minimale pour une matrice de précédence donnée. Cet algorithme est particulièrement bien adapté pour sa réalisation sur ordinateur électronique et peut être programmé facilement dans des langages tels qu'Algol 60 et PL/I qui permettent le traitement des vecteurs booléens.

## BIBLIOGRAPHIE

1. C. BERGE, *La théorie des graphes et ses applications*, Dunod, 1958.
2. D. F. MARTIN, On algorithms for the generation of the limiting form of boolean precedence matrices. Comm. privée.
3. C. V. RAMAMOORTHY, « Analysis of graphs by connectivity considerations » *J.A.C.M.* 13, 211-223, avril 1966.
4. B. ROY, « Cheminement et connexité dans les graphes. Application aux problèmes d'ordonnement », *Metra*, série spéciale # 1, 1962.
5. J. M. S. SIMOES PEREIRA, « On the boolean matrix equation  $M' = V_{i=1}^d M^i$  », *J.A.C.M.* 12, 376-382, juillet 1965.
6. S. WARSHALL, « A theorem on boolean matrices », *J.A.C.M.* 9, 11-12 janvier 1962.