

J. P. BORDAT

Complexité de problèmes liés aux graphes sans circuit

Informatique théorique et applications, tome 21, n° 2 (1987),
p. 181-197

http://www.numdam.org/item?id=ITA_1987__21_2_181_0

© AFCET, 1987, tous droits réservés.

L'accès aux archives de la revue « Informatique théorique et applications » 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/>

COMPLEXITÉ DE PROBLÈMES LIÉS AUX GRAPHES SANS CIRCUIT (*)

par J. P. BORDAT (2)

Communiqué par R. CORI

Résumé. – *Les liens entre les deux problèmes suivants :*

- *calcul de la fermeture transitive,*
- *calcul du graphe de Hasse,*

sont étudiés sur plusieurs classes de graphes sans circuit. Dans certains cas, il existe un algorithme de résolution linéaire. Sinon, il est possible d'établir une relation avec le calcul du produit de deux matrices quelconques.

Le problème de la reconnaissance de ces différentes classes est ensuite abordé dans la même optique.

Abstract. – *We study connections between two problems :*

- *finding the transitive reduction,*
- *finding the transitive closure,*

for several classes of acyclic digraphs. We point out linear time algorithmic solutions or elsewhere compare the given problems with matrix multiplication.

The recognition of these classes is then treated with the same tools.

I. INTRODUCTION

La section II contient les rappels et définitions nécessaires à la suite de l'exposé.

Dans la section III, la notion de réductibilité entre problèmes polynomiaux [10] est utilisée afin de clarifier les relations entre deux problèmes :

- calcul du graphe de Hasse,
- calcul de la fermeture transitive,

(*) Reçu janvier 1986, révisé novembre 1986.

(1) C.R.I.M., 860, rue de Saint-Priest, 34100 Montpellier.

concernant les graphes sans circuit, finis et connexes.

Nous noterons \mathcal{U} cette classe, dont certains invariants sont présentés dans [6].

Trois cas particuliers de graphes sans circuit, correspondant à des classes bien connues d'ensembles ordonnés sont étudiés :

– \mathcal{SM} -graphe : correspondant à un ensemble ordonné semi-modulaire.

La semi-modularité peut être supérieure (\mathcal{SMS} -graphe) ou inférieure (\mathcal{SMI} -graphe). Un ensemble ordonné possédant ces deux propriétés est dit modulaire.

– \mathcal{G} -graphe : correspond à un ensemble ordonné gradué. [2].

– \mathcal{JD} -graphe : correspondant à un ensemble ordonné vérifiant la propriété de Jordan Dedekind [2].

La technique envisagée permet d'établir un lien algorithmique étroit avec un problème bien connu (ici le calcul de la fermeture transitive ou du Graphe de Hasse d'un graphe quelconque). Toute amélioration de complexité obtenue par un travail sur ce problème de référence se répercute alors sur chacun des problèmes étudiés.

Dans la section IV sont abordés de la même manière les problèmes de reconnaissance pour les trois catégories.

II. DÉFINITIONS ET NOTATIONS

A. Définitions

Nous utiliserons la notation suivante, pour désigner la réduction entre 2 problèmes Π et Π' ayant chacun un graphe pour instance

$$(\Pi, f(t)) \rightarrow (\Pi', f'(t'))$$

où :

– t désigne la taille d'un graphe ayant un rapport direct avec l'instance du problème (voir ci-après),

– $f(f')$ est une fonction $\mathbb{R}^+ \rightarrow \mathbb{R}^+$,

– \rightarrow signifie : S'il existe un algorithme permettant de résoudre le problème Π' en $O(f'(t'))$ alors il existe un algorithme permettant de résoudre le problème Π en $O(f(t))$ (O notation classique employée dans [12]).

Si $G=(X, U)$ est un graphe simple, fini, connexe, sans circuit

– $G^t = (X, U^t)$ désignera la fermeture transitive de G ($(x, y) \in U^t \Leftrightarrow \exists$ un chemin noté $\sigma(x, y)$, joignant x à y dans G).

– $G_h = (X, U_h)$ désignera le graphe de Hasse (unique) correspondant à G ; ($U_h = U - \{(x, y) \in U \mid \exists$ un chemin $\sigma(x, y)$ de longueur ≥ 2 dans $G\}$).

– On peut décomposer G en niveaux :

X est partitionné en (N_1, N_2, \dots, N_p) , de telle façon que x appartienne à N_i si et seulement si un plus long chemin d'extrémité x contient i sommets.

On définit r , application de X dans \mathbb{N} telle que $x \in N_i \Leftrightarrow r(x) = i$. $r(x)$ est le rang du sommet x .

On note alors $G^r = (X, U^r)$ le graphe sans circuit ayant pour ensemble d'arcs

$$U^r = \{(x, y) \in X \times X \mid r(y) > r(x)\}$$

c'est-à-dire que, dans G^r , chaque sommet est relié par un arc à l'ensemble des sommets des niveaux supérieurs (les partitions en niveaux et la fonction rang sont les mêmes dans G et dans G^r).

La variable t utilisée dans la définition prendra essentiellement les valeurs $|U|, |U^t|, |U_h|, |U^r|$.

La notion de successeur (prédécesseur) sera prise au sens large.

Remarque : Les graphes considérés étant supposés connexes, (parce que correspondant à des ensembles ordonnés), la complexité classique $O(|X| + |U|)$ se réduit à $O(|U|)$ puisque $|U| \geq |X| - 1$, d'où une simplification des notations.

B. Rappels

– Un $\mathcal{SM}\mathcal{S}$ ($\mathcal{SM}\mathcal{I}$)-graphe G est tel que, dans G_h , l'existence de z prédécesseur (successeur) de x et y entraîne l'existence de t successeur (prédécesseur) de x et y .

Exemple de $\mathcal{SM}\mathcal{S}$ -graphe (voir fig. 1) : Les arcs (2,8), (5,9), (3,9) sont des arcs de transitivité.

Un \mathcal{G} -graphe $G = (X, U)$ est tel qu'on puisse associer à $G_h = (X, U_h)$ une application $g : X \rightarrow \mathbb{Z}$ telle que

$$\forall (x, y) \in U_h, \quad g(y) = g(x) + l \quad (g \text{ est une graduation}).$$

Exemple de \mathcal{G} -graphe (voir fig. 2) : Arcs de transitivité (2,8) (5,8).

Un $\mathcal{J}\mathcal{D}$ -graphe $G = (X, U)$ est tel que, dans $G_h = (X, U_h)$, tous les chemins joignant 2 sommets quelconques soient de même longueur.

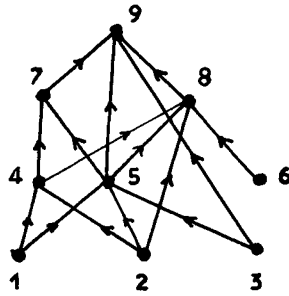


Figure 1.

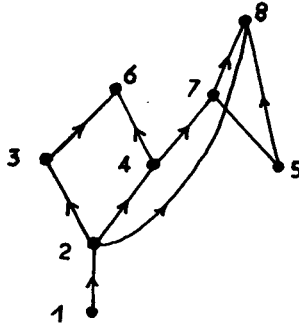


Figure 2.

Exemple de $\mathcal{I}\mathcal{D}$ -graphe (voir fig. 3) : Arcs de transitivité (1,5), (1,6).

Remarques :

1° Tout $\mathcal{S}\mathcal{M}$ -graphe est un \mathcal{G} -graphe, et tout \mathcal{G} -graphe est un $\mathcal{I}\mathcal{D}$ -graphe.

2° Le \mathcal{G} -graphe donné en exemple n'est pas un $\mathcal{S}\mathcal{M}$ -graphe : 6 et 7 ont pour prédécesseur 4 et n'ont pas de successeur commun.

3° Le $\mathcal{I}\mathcal{D}$ -graphe présenté n'est pas un \mathcal{G} -graphe, l'existence de l'arc (2,5) ruinant la possibilité d'existence d'une graduation sur le graphe de Hasse.

III. RÉDUCTIONS ENTRE LE CALCUL DE LA FERMETURE TRANSITIVE ET CELUI DU GRAPHE DE HASSE

Pour désigner les problèmes, nous utilisons les notations : FERTRAN, HASSE, suivies d'un suffixe $\mathcal{S}\mathcal{M}$, $\mathcal{I}\mathcal{D}$ ou \mathcal{G} .

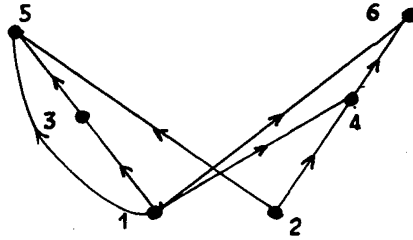


Figure 3.

Exemple : FERTRAN- $\mathcal{J}\mathcal{D}$.

Donnée : Un $\mathcal{J}\mathcal{D}$ -graphe $G=(X, U)$.

Résultat : $G^t=(X, U^t)$.

A. Cas d'un $\mathcal{J}\mathcal{D}$ -graphe

PROPRIÉTÉ 1 :

$$(\text{FERTRAN-}\mathcal{J}\mathcal{D}, |U^t|) \rightarrow (\text{HASSE-}\mathcal{J}\mathcal{D}, |U|).$$

Preuve : Étant donné un $\mathcal{J}\mathcal{D}$ -graphe $G=(X, U)$ pour lequel on souhaite calculer la fermeture transitive, on construit un graphe G' ayant trois fois plus de sommets et un nombre d'arcs de l'ordre de $|U^t|$. Le calcul du graphe de Hasse de G' est alors équivalent à celui de la fermeture transitive de G . Plus précisément, on a : (cette construction est utilisée dans [1], [10] pour G quelconque).

1° Soit la décomposition en niveaux de $G: N_1, N_2, \dots, N_p$, calculée en $O(|U|)$, et soit r le rang associé.

2° Soit $G'=(X, U')$, dans lequel X' est formé de 3 copies de X :

– $X' = X_1 \cup X_2 \cup X_3$ avec

$$\begin{aligned} X_1 &= \{x'_1, x'_2, \dots, x'_n\}, \\ X_2 &= \{x_1, x_2, \dots, x_n\} = X, \\ X_3 &= \{x''_1, x''_2, \dots, x''_n\} \end{aligned}$$

ayant pour ensemble d'arcs

– $U' = U \cup U_1 \cup U_2$ avec

$$U_1 = \bigcup_{x_i \in X} \{(x'_i, x_i), (x_i, x'_i)\}, \quad U_2 = \{(x'_i, x'_j) \mid r(x_j) > r(x_i)\}.$$

Le nombre d'éléments de U' est alors :

$$|U'| = |U| + 2|X| + \sum_{1 < i+1 \leq j \leq p} |N_i| |N_j| \leq 2|X| + |U| + |U'|$$

G' satisfait les propriétés suivantes :

(a) G' est un $\mathcal{J}\mathcal{D}$ -graphe. En effet, G' est sans circuit. Soient $y, z \in X'$ et deux chemins maximaux $\sigma(y, z)$ et $\sigma'(y, z)$ dans G' .

Dans les cas suivants :

$$y, z \in X_2, \quad y \in X_1 \quad \text{et} \quad z \in X_2, \quad y \in X_2 \quad \text{et} \quad z \in X_3$$

Les longueurs de ces chemins sont trivialement égales.

Supposons maintenant $y \in X_1, z \in X_3$. Il existe i, j tels que $y = x'_i$ et $z = x'_j$. Les chemins σ et σ' diffèrent par le premier et le dernier arc, sinon on est ramené aux cas précédents.

D'où $\sigma = (x'_i x'_j)$ ou $\sigma' = (x'_i, x'_j)$, et l'un des deux chemins est constitué par un arc de transitivité.

(b) Si $G'_h = (X', U'_h)$, alors

$$(x, y) \in U^t \Leftrightarrow \text{il existe un chemin dans } G : \sigma(x, y)$$

$$\Leftrightarrow \text{il existe un chemin } [x', \sigma(x, y), y'] \text{ dans } G'$$

$$\Leftrightarrow (x', y') \in U' - U'_h$$

(c) U'_h étant par hypothèse calculable en $O(|U^t|)$, U^t isomorphe à $U' - U'_h$ s'obtient avec cette même complexité, ce qui achève la preuve. ■

Exemple (voir fig. 4) :

Les arcs devant être rajoutés dans G' sont les suivants :

$$(1' 3'') (1' 5'') (1' 4'') (1' 6'') (2' 3'') (2' 5'') (2' 4'') (2' 6'')$$

$$(3' 5'') (3' 6'') (4' 5'') (4' 6'').$$

Seuls les arcs $(4' 5'')$, $(3' 6'')$ et $(2' 3'')$ ne sont pas des arcs de transitivité de G' .

Remarque : La propriété précédente n'est plus vraie si l'on remplace la classe $\mathcal{J}\mathcal{D}$ par la classe \mathcal{G} . En effet, si G est un \mathcal{G} -graphe, G' n'est pas nécessairement dans la même classe, comme le montre l'exemple de la figure 5 :

L'existence de l'arc $(2' 5'')$, par exemple, détruit dans G' la propriété de graduation.

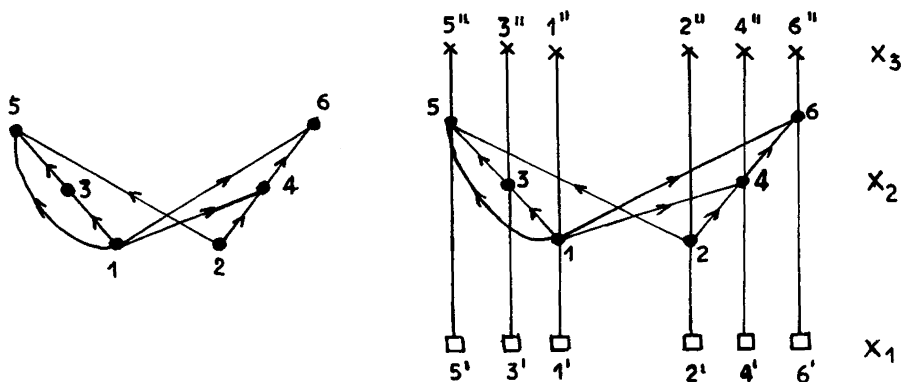


Figure 4.

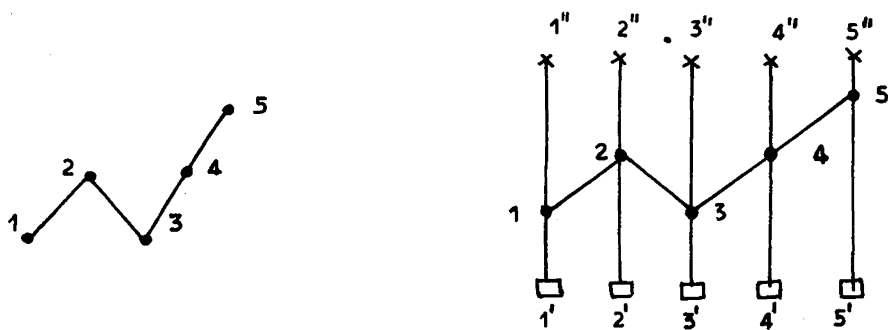


Figure 5.

PROPRIÉTÉ 2 :

$$(\text{HASSE-}\mathcal{G}\mathcal{D}, |U^t|) \rightarrow (\text{FERTRAN-}\mathcal{G}\mathcal{D}, |U^t|)$$

Preuve : Il suffit de remarquer que la réduction suivante est valable sur un $\mathcal{G}\mathcal{D}$ -graphe :

Soit l'algorithme :

1° Calcul de $G^t = (X, U^t) : O(|U^t|)$.

2° Construction du graphe triparti $G' = (X', U')$ tel que

$$X' = X_1 \cup X_2 \cup X_3$$

$$X_1 = X, \quad X_2 = \{x'_i \mid x_i \in X\}, \quad X_3 = \{x''_i \mid x_i \in X\}$$

$$\begin{aligned}
 U' &= U_1 \cup U_2 = U'_h \\
 U_1 &= \{(x_i, x'_j) \mid (x_i, x_j) \in U^t\} \\
 U_2 &= \{(x'_i, x'_j) \mid (x_i, x_j) \in U^t\} \\
 |X'| &= 3|X|, \quad |U'| = 2|U^t|
 \end{aligned}$$

Cette construction se fait donc en $O(|U^t|)$.

3° Calcul de $G'^t = (X', U'^t)$.

Remarquons que :

$$\begin{aligned}
 U'^t &= U' \cup \{(x_i, x'_j) \mid \exists x'_k \text{ tel que } (x_i, x'_k) \in U_1 \text{ et } (x'_k, x'_j) \in U_2\} \\
 U'^t &= U' \cup \{(x_i, x'_j) \mid \exists x_k \text{ tel que } (x_i, x_k) \in U^t \text{ et } (x_k, x_j) \in U^t\} \\
 U'^t &= U' \cup \{(x_i, x'_j) \mid (x_i, x_j) \in U^t - U_h\}.
 \end{aligned}$$

D'où $|U'^t| \leq 3|U^t|$.

G' , triparti, étant un $\mathcal{F}\mathcal{D}$ -graphe, cette étape demande $O(|U^t|)$ opérations.

4° $U_h = U - (U^t - U_h)$.

L'ensemble $(U^t - U_h)$, est isomorphe à $U'^t - U'$. L'obtention de U_h demande donc également $O(|U^t|)$ opérations, ce qui termine la preuve. ■

B. Cas des \mathcal{G} et \mathcal{SM} -graphes

Le calcul du graphe de Hasse pour un \mathcal{SM} -graphe ou un \mathcal{G} -graphe est équivalent au calcul d'une graduation g sur l'ensemble des sommets puisque

$$U_h = \{(x, y) \in U \mid g(y) = g(x) + 1\}.$$

On en déduit les deux propriétés suivantes :

PROPRIÉTÉ 3 : Il existe un algorithme linéaire résolvant HASSE- \mathcal{SM} .

Preuve : Raisonnons sur un \mathcal{SM} -graphe $G = (X, U)$. Le principe de l'algorithme est très simple, puisque dans le cas où le graphe possède la propriété de graduation ainsi que celle de source unique, ce qui est le cas ici, l'application r constitue une graduation [2]. D'où deux étapes, toutes deux linéaires :

1° Calcul de r .

2° $U_h = U - \{(x, y) \in U \mid r(y) > r(x) + 1\}$.

PROPRIÉTÉ 3' : Il existe un algorithme linéaire résolvant HASSE- \mathcal{G} .

Cet algorithme est exposé en détail dans [5]. Nous en donnerons uniquement le principe, qui consiste à augmenter itérativement un ensemble de sommets, noté GRAD, sur lequel a été calculée une graduation g .

Chaque étape se décompose en les actions suivantes :

1° choix d'un sommet $x \in X\text{-GRAD}$, et tel que $D(x) \cap \text{GRAD} \neq \emptyset$.

2° calcul du rang r sur le sous-graphe construit sur les descendants de x n'appartenant pas à GRAD.

3° Pour tout $y \in D(x)\text{-GRAD}$, on pose $\delta(y) = \min \{1 + g(z) - r(t)\}$, le min étant pris sur l'ensemble des arcs (t, z) , $t \in D(x)\text{-GRAD}$, $z \in \text{GRAD}$.

4° GRAD est incrémenté de l'ensemble $D(x)\text{-GRAD}$, avec, pour tout $y \in D(x)\text{-GRAD}$, $g(y) = r(y) + \delta(y)$.

L'ensemble GRAD est initialisé en effectuant le calcul du rang sur l'ensemble des descendants d'un sommet quelconque x_0 , avec $\forall y \in D(x_0)$, $\delta(y) = 0$, et le processus se termine lorsque $\text{GRAD} = X$.

On montre alors que l'application $g: X \rightarrow \mathbb{Z}$ définie par

$$\forall x \in X, \quad g(x) = r(x) + \delta(x)$$

est une graduation.

De plus, le choix de x et le calcul de r à chaque étape peuvent être réalisés en une complexité globale $O(|U|)$, en nombre d'opérations et en place mémoire.

Notons qu'un algorithme différent est exposé dans [9], qui demande plus de place mémoire, et une structure de données plus compliquée.

Propriété 4 :

$$(\text{FERTRAN-}\mathcal{G}, |U^r|) \rightarrow (\text{FERTRAN-}\mathcal{S.M.I.}, |U^r|).$$

Remarquons que l'on a déjà d'après II, 2)

$$(\text{FERTRAN-}\mathcal{S.M.I.}, |U^r|) \rightarrow (\text{FERTRAN-}\mathcal{G}, |U^r|).$$

Preuve : Elle se divise en 3 étapes :

1° $G = (X, U)$ étant un \mathcal{G} -graphe, il est possible de calculer la graduation associée en $O(|U|)$, d'après ce qui précède.

On obtient ainsi une partition de $X: X_1, X_2, \dots, X_p$ et une application g (graduation) : $X \rightarrow \{1, 2, \dots, p\}$ (à une translation près).

2° Construction du graphe $G' = (X, U')$ tel que :

$$X' = X \cup \{y_0, y_1, \dots, y_{p-1}\}$$

$$U' = U \cup U_1$$

$$U_1 = \{\cup \{y_{i-1}\} \times \{X_i \cup y_i\} \mid i = 1, \dots, p-1\} \cup \{\{y_{p-1}\} \times X_p\}$$

(On rajoute un sommet y_i à chaque classe X_i , $i=1, \dots, p-1$, auquel on associe comme successeurs tous les sommets de la nouvelle classe suivante. Un sommet y_0 est relié à tous les éléments de X_1).

Il est facile de montrer que :

- G' est gradué par $g' : g'(x) = g(x), \forall x \in X, g'(y_i) = i, 0 \leq i \leq p$,
- G' est un \mathcal{SMI} -graphe.

Cette construction s'établit en $O(|X|)$.

D'autre part g' donne la fonction rang dans G' (propriété des \mathcal{G} -graphes). On aura donc $|U'^t| \leq 3|U^r|$ en considérant séparément les arcs de U'^t de la forme (x_i, x_j) , ceux ayant un y_i comme extrémité initiale ou finale, et ceux de la forme (y_i, y_j) .

3° Calcul de U'^t

On a alors, si x_i et x_j sont éléments de X :

$(x_i, x_j) \in U'^t \Leftrightarrow$ il existe un chemin $\sigma(x_i, x_j)$ dans G' .

\Leftrightarrow il existe un chemin $\sigma(x_i, x_j)$ dans G .

(Aucun y_k ne peut figurer sur $\sigma(x_i, x_j)$ puisque y_k ne peut avoir un x pour prédécesseur).

$\Leftrightarrow (x_i, x_j) \in U^t$

U^t apparaît donc comme la restriction de U'^t aux arcs de type (x_i, x_j) , ayant leurs deux extrémités dans X .

Si la détermination de U'^t s'effectue en $O(|U'^t|)$, la détermination de U^t s'effectue en $O(|U^r|)$ et la propriété est démontrée. ■

Exemple (voir fig. 6) :

Les arcs de transitivité du type (x_i, x_j) sont les mêmes dans G^t et G'^t .

D. Cas des 3 \mathcal{N} -graphes

Nous désignerons par 3 \mathcal{N} -graphe un graphe possédant exactement 3 niveaux.

PROPRIÉTÉ 5 :

$(\text{HASSE-3 } \mathcal{N}, |U^t|) \rightarrow (\text{FERTRAN-3 } \mathcal{N}, |U^r|)$.

Il suffit d'utiliser la réduction de la propriété 2 en remarquant qu'elle est valable si G et G' sont des 3 \mathcal{N} -graphes.

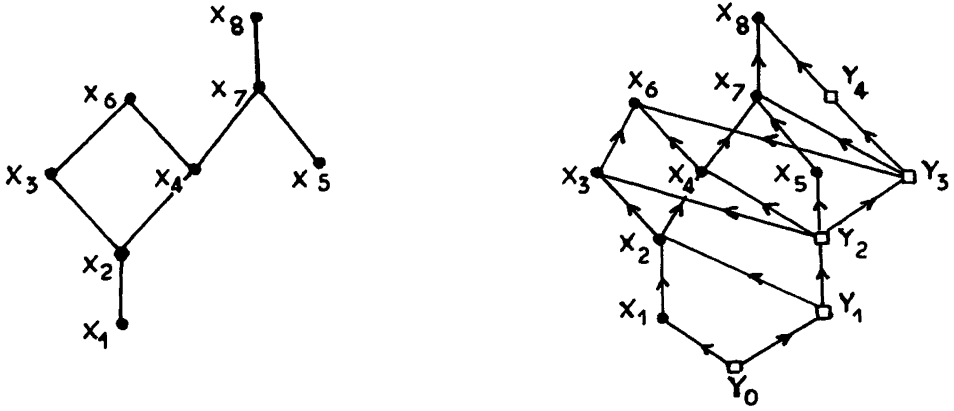


Figure 6.

PROPRIÉTÉ 6 :

$$(FERTRAN-3 \mathcal{N}, |U^r|) \rightarrow (HASSE-3 \mathcal{N}, |U|).$$

L'algorithme est donné par les opérations suivantes :

1° Calcul de la décomposition en niveaux N_1, N_2, N_3 de $G=(X, U)$ ceci en $O(|U|)$.

2° Construction du graphe $G'=(X, U')$ avec $X'=X, U'=U \cup N_1 \times N_3$ en $O(|U^r|)$, puisque $|N_1 \times N_3| \leq |U^r|$ (On rajoute tous les arcs joignant un sommet du premier niveau à un sommet du dernier niveau).

3° Calcul de $G'_h=(X', U'_h)$ en $O(|U^r|)$.

4° Calcul de $U^t = U \cup (N_1 \times N_3 - U'_h)$ en $O(|U^r|)$.

En effet $(x, y) \in U^t - U \Rightarrow (x, y) \in N_1 \times N_3 - U'_h$.

D'autre part

$$\begin{aligned} (x, y) \in N_1 \times N_3 - U'_h &\Rightarrow \exists z \in N_2 \text{ tel que } (x, z) \in U' \text{ et } (z, y) \in U' \\ &\Rightarrow \exists z \in N_2 \text{ tel que } (x, z) \in U \text{ et } (z, y) \in U \\ &\Rightarrow (x, y) \in U^t. \end{aligned}$$

Nous obtenons enfin la dernière réduction :

PROPRIÉTÉ 7 :

$$(FERTRAN-3 \mathcal{N}, |U^r|) \rightarrow (FERTRAN-\mathcal{G}, |U^t|).$$

Soit $G = (X, U)$ un $3\mathcal{N}$ -graphe. On effectuera la réduction suivante :

1° Calcul de la décomposition en niveaux de G en $O(|U|)$.

2° Construction du \mathcal{G} -graphe triparti $G' = (X', U')$ où

$$X' = X \quad U' = U - \{(x_i, x_j) \mid r(x_i) = 0 \text{ et } r(x_j) = 2\}$$

3° Calcul de G'^t en $O(|U'^t|)$ donc en $O(|U^t|)$.

4° Calcul de $U^t = U'^t \cup \{(x_i, x_j) \mid r(x_i) = 0 \text{ et } r(x_j) = 2\}$.

E. Conclusion

Si l'on remplace les complexités observées par la complexité plus générale $|X|^2$, nous avons obtenu la chaîne de réductions suivante : (\rightleftharpoons est noté \leftrightarrow)

$$\begin{aligned} &(\text{HASSE-}\mathcal{J}\mathcal{D}, |X|^2) \leftrightarrow (\text{FERTRAN-}\mathcal{J}\mathcal{D}, |X|^2) \\ &\quad \downarrow \\ &(\text{FERTRAN-}\mathcal{S}\mathcal{M}\mathcal{J}, |X|^2) \leftrightarrow (\text{FERTRAN-}\mathcal{G}, |X|^2) \\ &\quad \downarrow \\ &(\text{HASSE-}3\mathcal{N}, |X|^2) \leftrightarrow (\text{FERTRAN-}3\mathcal{N}, |X|^2) \end{aligned}$$

D'autre part, les articles classiques comparant FERTRAN au calcul d'un produit de matrices présentent les deux réductions :

$$(\text{MAT-}n, n^2) \rightarrow (\text{FERTRAN-}3\mathcal{N}, |X|^2) \quad [7]$$

$$(\text{FERTRAN}, |X|^2) \rightarrow (\text{MAT-}n, n^2) \quad [15]$$

où MAT- n désigne le calcul du produit de deux matrices booléennes d'ordre n . Enfin


$$(\text{HASSE}, |X|^2) \leftrightarrow (\text{FERTRAN}, |X|^2) \quad [10].$$

Considérons la relation sur l'ensemble des problèmes admettant pour instance un élément de \mathcal{U} (\mathcal{U} ensemble des graphes finis, connexes, sans circuit) :

$$(\Pi, |X|^2) \rightarrow (\Pi', |X|^2).$$

Il s'agit évidemment d'une relation de préordre. Tous les problèmes évoqués appartiennent à la même classe \mathcal{C} associée à cette relation. On peut donc établir le tableau terminal suivant, donnant la complexité au sens du plus mauvais cas, ou la classe de complexité des problèmes envisagés :

	$\mathcal{S}\mathcal{M}$	\mathcal{G}	$\mathcal{J}\mathcal{D}$	$3\mathcal{N}$	\mathcal{U}
FERTRAN.	$\in \mathcal{C}$	$\in \mathcal{C}$	$\in \mathcal{C}$	$\in \mathcal{C}$	$\in \mathcal{C}$
HASSE.	linéaire	linéaire	$\in \mathcal{C}$	$\in \mathcal{C}$	$\in \mathcal{C}$

Remarque : Des algorithmes linéaires résolvant HASSE existent pour les arborescences, les graphes série-parallèle [17], ou correspondant aux ensembles ordonnés « *N*-free » (sans  au sens sous-graphe dans le graphe de Hasse, [11]).

Signalons également que FERTRAN est résolu en une complexité $O(|X|^2)$ pour les catégories ci-dessus, à l'exception de celle des ensembles *N*-free (équivalente au cas général), ainsi que pour la classe des ordres de dimension 2, et celle des treillis distributifs [4].

IV. PROBLÈMES DE RECONNAISSANCE

Dans ce paragraphe sont abordés les problèmes de reconnaissance des propriétés de semi-modularité, graduation, et Jordan-Dedekind, les deux premiers cas étant classiques. La terminologie utilisée est la suivante : REC-suffixe.

Exemple : REC- \mathcal{G} .

Donnée : un *graphe de Hasse* $G=(X, U)$.

Résultat : Un booléen vrai si et seulement si G est un \mathcal{G} -graphe.

Il est logique de se restreindre à un graphe de Hasse comme donnée. Il semble en effet difficile au vu des résultats obtenus de trouver, pour chacune des 3 catégories, un algorithme de reconnaissance qui calcule conjointement le graphe de Hasse, comme c'est par exemple le cas pour les graphes série-parallèle [17].

A. Semi-modularité

Soit A la matrice booléenne associée à G : les sommets de G sont alors numérotés de 1 à $|X|$, et soit v cette numérotation.

$$A(i, j) = 1 \iff (v^{-1}(i), v^{-1}(j)) \in U \quad 1 \leq i, j \leq |X|$$

$$= 0 \quad \text{sinon}$$

Il est facile d'établir que la propriété de semi-modularité inférieure par exemple est vraie si et seulement si

$$A \cdot A^t \leq A^t \cdot A \quad [3]$$

(A^t désigne la transposée de A et l'inégalité doit être vérifiée pour chaque terme du produit booléen).

En conséquence

$$(\text{REC-}\mathcal{SM}, |X|^\alpha) \rightarrow (\text{MAT-}n, n^\alpha), \quad \alpha \geq 2$$

On ne peut établir directement la réduction inverse, car REC- \mathcal{SM} est un problème de décision admettant une solution de type booléen.

On se contentera de remarquer que si l'on connaît un algorithme permettant de trouver dans G tous les couples de sommets x, y tels que x et y possèdent un successeur commun, alors on dispose d'un algorithme permettant de résoudre FERTRAN-3 \mathcal{N} avec la même complexité ([3]). Une reconnaissance de semi-modularité basée sur la définition est donc au moins aussi difficile que FERTRAN-3 \mathcal{N} . Par ailleurs, il existe des algorithmes faciles de complexité $O(|X| |U|)$ résolvant REC- \mathcal{SM} ([3]).

B. Graduation

Le problème REC- \mathcal{G} se résout par un algorithme simple, dont le principe est le suivant :

On effectue un parcours du graphe $G' = (X, U \cup U^*)$, avec $U^* = \{(y, x) \mid (-x, y) \in U\}$ en partant d'un sommet quelconque x_0 et en posant $g(x_0) = 0$. A chaque étape est parcouru un arc (x, y) tel que $g(x)$ soit défini. Deux cas peuvent se présenter :

– Si $g(y)$ n'est pas défini, on pose $g(y) = g(x) + 1$ si $(x, y) \in U$ et $g(y) = g(x) - 1$ si $(y, x) \in U$.

– Sinon, on vérifie la cohérence de $g(y)$ par rapport à $g(x)$ suivant le même critère.

La non-existence de cette cohérence pour un sommet donné entraîne le résultat faux pour REC- \mathcal{G} . Dans le cas contraire, g est une graduation.

Ce traitement peut s'effectuer en une complexité $O(|U|)$, grâce à des parcours classiques (en largeur, en profondeur, voir par exemple [5]).

REC- \mathcal{G} est donc linéaire. Signalons qu'on a la même propriété pour la reconnaissance de graphes de Hasse d'ensembles ordonnés « N -free » [11] ou de treillis distributifs [4].

C. Propriété de Jordan-Dedekind

Il est intéressant de rapprocher REC- $\mathcal{J}\mathcal{D}$ du problème de recherche de plus court chemin entre tous les couples de sommets d'un graphe. Une bibliographie abondante concerne ce dernier (voir par exemple [8, 13, 16]).

Nous utiliserons une réduction établie dans [16] :

$$(\text{CHEM} \geq 0, |X|^\alpha) \rightarrow (\text{MAT-}n, n^\alpha), \quad \alpha \geq 2$$

$\text{CHEM} \geq 0$ a pour donnée un graphe $G=(X, U)$ muni d'une valuation positive ou nulle (application de U dans $\mathbb{R}^+ \cup \{0\}$). Il calcule, pour tout couple x, y de $X \times X$, la valeur d'un plus court chemin joignant x à y s'il en existe, un résultat standard (∞ par exemple) sinon.

PROPRIÉTÉ 8 :

$$(\text{REC-}\mathcal{D}, |X|^\alpha) \rightarrow (\text{MAT-}n, n^\alpha), \quad \alpha \geq 2$$

Preuve : A partir d'un algorithme calculant le produit de deux matrices, on peut opérer de la façon suivante :

1° Résolution de $\text{CHEM} \geq 0$ sur G muni de la valuation associant à tout arc de G la valeur 1 : $O(|X|^\alpha)$.

2° Calcul de G' : $O(|X|^\alpha)$.

3° Construction du graphe $G'=(X, U')$:

– $X=X_1 \cup X_2 \cup X_3$ avec

$$X_1 = \{x_1, x_2, \dots, x_n\} = X$$

$$X_2 = \{x'_1, x'_2, \dots, x'_n\}$$

$$X_3 = \{x''_1, x''_2, \dots, x''_n\}$$

– $U' = U_1 \cup U_2$ avec

$$U_1 = \{(x_i, x'_j) \mid (x_i, x_j) \in U\}$$

$$U_2 = \{(x'_i, x''_j) \mid (x_i, x_j) \in U\}$$

Pour simplifier le calcul matriciel qui suit, nous supposons que les sommets de G' possèdent une numérotation v telle que

$$\forall x \in X_1, \quad 1 \leq v(x) \leq |X|$$

$$\forall x \in X_2, \quad |X| < v(x) \leq 2|X|$$

$$\forall x \in X_3, \quad 2|X| < v(x) \leq 3|X|.$$

On associe alors à G' deux matrices carrées A et B de dimension $3|X|$, à coefficients entiers positifs ou nuls.

$$A(i, j) = 1 \Leftrightarrow (v^{-1}(i), v^{-1}(j)) \in U'$$

$$= 0 \quad \text{sinon}$$

$$\begin{aligned}
 B(i, j) = 1 &\Leftrightarrow (v^{-1}(i), v^{-1}(j)) \in U_1 \\
 &= l(v^{-1}(i), v^{-1}(j)) + 1 \Leftrightarrow (v^{-1}(i), v^{-1}(j)) \in U_2 \\
 &= 0 \quad \text{sinon}
 \end{aligned}$$

$l(v^{-1}(i), v^{-1}(j))$ est la longueur d'un plus court chemin dans G joignant $v^{-1}(i)$ à $v^{-1}(j)$ donnée par l'étape 1.

4° Calcul de $C = B^2$, $D = A^2$, produit effectué cette fois avec les opérations classiques $+$, \times .

5° Pour tout i, j , $1 \leq i \leq |X|$, $2|X| < j \leq 3|X|$ on vérifie l'égalité

$$C(i, j) = D(i, j) \times l(v^{-1}(i), v^{-1}(j))$$

$D(i, j)$ est égal au nombre de chemins (de longueur 2) joignant $v^{-1}(i)$ et $v^{-1}(j)$ dans G' . Fixons i et j .

— Si G possède la propriété de Jordan-Dedekind, pour tout k , $|X| < k \leq 2|X|$ tel que $(v^{-1}(i), v^{-1}(k)) \in U_1$ et $(v^{-1}(k), v^{-1}(j)) \in U_2$ $l(v^{-1}(k), v^{-1}(j))$ possède la même valeur et

$$C(i, j) = \sum_k (l(v^{-1}(k), v^{-1}(j)) + 1) = D(i, j) \times l(v^{-1}(i), v^{-1}(j))$$

— Dans le cas contraire, il existe $x, y, z, t \in X$, x et y successeurs de z , t descendant de x et y dans G et tels que la longueur d'un plus court chemin joignant x à t et la longueur d'un plus court chemin joignant y à t soient différentes.

On obtient alors

$$C(v(z), v(t)) > D(v(z), v(t)) \times l(z, t)$$

Les étapes 3 et 5 peuvent être réalisées avec $O(|X|^2)$ opérations élémentaires, et l'étape 4 avec une complexité $O(|X|^n)$, ce qui achève la preuve.

Inversement, on remarquera qu'un algorithme résolvant REC- $\mathcal{J}\mathcal{D}$ en calculant, pour toute source de G la longueur d'au moins un chemin joignant cette source à chacun de ses descendants (ce qui ne paraît pas déraisonnable) permet de résoudre FERTRAN-3 \mathcal{N} .

Si REC- $\mathcal{J}\mathcal{D}$ n'est donc pas plus difficile que MAT- n , il semble peu probable d'obtenir un algorithme de complexité $O(|X|^2)$.

Signalons enfin qu'un algorithme simple, de complexité

$$O(\min(|\mathcal{S}|, |\mathcal{P}|) |U|),$$

où $|\mathcal{S}|$ et $|\mathcal{P}|$ sont les cardinaux des ensembles respectifs de sources et de puits de G consiste à résoudre REC- $\mathcal{J}\mathcal{D}$ sur le sous-graphe des descendants de chacune des sources (resp. des puits) de G .

BIBLIOGRAPHIE

1. A. V. AHO, M. R. GAREY et J. D. ULLMAN, *The Transitive Reduction of a Directed Graph*, S.I.A.M. J. Comput., vol. 1, n° 2, juin, 1972, p. 131-137.
2. M. BARBUT et B. MONJARDET, *Ordre et Classification, Algèbre et Combinatoire*, 2 tomes, Hachette, Paris, 1970.
3. J. P. BORDAT, *Parcours dans les graphes : un outil pour l'algorithmique des ensembles ordonnés*, Discr. Appl. Math., vol. 12, 1985, p. 215-231.
4. J. P. BORDAT, *Propriétés algorithmiques des treillis distributifs*, Rapp. Rech., C.R.I.M., Montpellier (en préparation).
5. J. P. BORDAT et O. COGIS, *Parcours dans les graphes sans circuit*, Rapp. rech., n° 11, C.R.I.M., Montpellier, mai 1985.
6. G. CHATY et M. CHEIN, *Invariants liés aux chemins dans les graphes sans circuit*, Coll. Int. Th. Comb. Rome, 3-15 sept. 1973, p. 287-308.
7. M. J. FISCHER et A. R. MEYER, *Boolean Matrix Multiplication and Transitive Closure*, Conference Record, I.E.E.E. 12th Annual Symposium on Switching and Automata Theory, 1971, p. 129-131.
8. M. L. FREDMAN, *New Bounds on the Complexity of the Shortest Path Problem*, S.I.A.M. J. Comput., vol. 5, n° 1, mars 1976, p. 83-88.
9. A. GORALCIKOVA et V. KOUBEK, *A Reduct-and-Closure Algorithm for Graphs*, Proceedings of M.F.C.S. 79, Springer-Verlag, Berlin, Heidelberg, New York, 1979, p. 301-307.
10. M. HABIB, M. HAMROUN et R. JEGOU, *Linear equivalences for transitivity in Graphs*, Rapp. Rech. n° 83-10, E.N.S.M. Saint-Étienne, Discrete Applied Mathematics (soumis).
11. M. HABIB et R. JEGOU, *N-free Posets as Generalizations of Series-Parallel Posets*, Discr. Appl. Math., vol. 12, 1985, p. 279-291.
12. D. E. KNUTH, *Big Omicron and Big Omega and Big Theta*, Sigact News, avril-juin 1976, p. 18-24.
13. K. MELHORN, *Data Structures and Algorithms 2 : Graph Algorithms and NP completeness*, Springer-Verlag, 1984.
14. B. MONJARDET, *Caractérisations métriques des ensembles ordonnés semi-modulaires*, Math. Sci. Hum., vol. 56, 1976, p. 77-87.
15. I. MUNRO, *Efficient Determination of the Transitive Closure of a Directed Graph*, Dept. of Computer Science, University of Toronto, U.S.A., Inform. Proc. Lett., vol. 1, 1971, p. 56-58.
16. F. ROMANI, *Shortest Path Problem is not harder than Matrix Multiplication*, Inform. Proc. Lett., vol. 11, n° 3, 1981, p. 134-136.
17. J. VALDES, R. E. TARJAN et E. L. LAWLER, *The Recognition of Series-Parallel Digraphs*, S.I.A.M. J. Comput., vol. 11, n° 2, 1982, p. 298-313.