

S. R. SCHWER

## Dépliage d'un graphe pointé en un cactus

*Informatique théorique et applications*, tome 21, n° 4 (1987),  
p. 469-477

[http://www.numdam.org/item?id=ITA\\_1987\\_\\_21\\_4\\_469\\_0](http://www.numdam.org/item?id=ITA_1987__21_4_469_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/>

## DÉPLIAGE D'UN GRAPHE POINTE EN UN CACTUS \*

par S. R. SCHWER <sup>(1)</sup>

Communiqué par J.-E. PIN

---

*Résumé. – Le but de cette note est de montrer un résultat sur les graphes, dont l'origine est une question relative aux réseaux de Petri : pour tout graphe étiqueté  $G$ , dont on a distingué un sommet, il existe un graphe ayant les mêmes étiquettes de chemins et possédant une forme particulière que nous baptisons cactus. Cette transformation se fait de façon constructive, par un algorithme de type Mac-Naughton et Yamada [5]. Le résultat peut être utilisé dans différents domaines de l'informatique. Citons par exemple les schémas de programmes [4] pour leur structuration, les réseaux de Petri afin de transformer le graphe de couverture en un graphe plus facile à manipuler [6].*

*Abstract. – We show that every labelled graph, with a distinguished node, can be transformed into a graph with the same set of labels and a particular shape that we call cactus. This transformation is a constructive one with a Mac-Naughton and Yamada's algorithm. Applications are a lot, as: program's scheme for their structuration, coverly graph of Petri nets for their handling.*

### PRÉLIMINAIRES

Nous ne considérons que des graphes orientés et étiquetés et utilisons librement le vocabulaire classique de la théorie des graphes [2] : chemin, cycle...

NOTATIONS : Un graphe étiqueté est un triple  $\langle X, \mathbb{Q}, \delta \rangle$  où

$X$  est un alphabet fini : celui des étiquettes

$\mathbb{Q}$  est un ensemble fini : celui des sommets

---

(\*) Reçu Septembre 1986, révisé en mars 1986.

(<sup>1</sup>) Université de Bordeaux-I, Département d'Informatique, 351, cours de la Libération, 33405 Talence Cedex.

$\delta \subseteq \mathbb{Q} \times X \times \mathbb{Q}$  est l'ensemble des arcs : si  $(q, x, q') \in \delta$ , il existe un arc d'étiquette  $x$  joignant  $q$  à  $q'$ ,

Nous nous limitons aux graphes connexes.

Un graphe est dit *émondé* par rapport à un sous-ensemble de sommets  $F$  si et seulement si tout sommet du graphe est à la fois accessible et coaccessible à partir de  $F$ . Nous noterons  $Acc(q)$  l'ensemble des sommets accessibles à partir de  $q$  autre que  $q$  et  $Coacc(q)$  l'ensemble des sommets coaccessibles à partir de  $q$ , autre que  $q$ . Un graphe pointé  $(G, q)$  est un graphe étiqueté dont on a distingué le sommet  $q$ .

DÉFINITION 1 : Soient  $G_1 = \langle X, \mathbb{Q}_1, \delta_1 \rangle$  et  $G_2 = \langle X, \mathbb{Q}_2, \delta_2 \rangle$  deux graphes. Le graphe pointé  $(G_1, q_1)$  est un *dépliage* du graphe pointé  $(G_2, q_2)$  s'il existe une application surjective  $v$  de  $\mathbb{Q}_1$  sur  $\mathbb{Q}_2$  qui vérifie

- (i)  $v(q_1) = q_2$
- (ii)  $(q, x, q') \in \delta_1 \Rightarrow (v(q), x, v(q')) \in \delta_2$
- (iii)  $(\forall (q, x, q') \in \delta_2) (\forall p \in \mathbb{Q}_1 \text{ t. q. } v(p) = q) (\exists p' \in \mathbb{Q}_1 | v(p') = q' \text{ et } (p, x, p') \in \delta_1)$

Ce dépliage de graphe duplique les sommets tout en conservant l'ensemble des étiquettes des chemins; plus précisément on a la

PROPOSITION 1 : *Pour tout chemin de  $G_2$  d'étiquette  $f$ , il existe un chemin de  $G_1$  ayant même étiquette et réciproquement.*

*Preuve :* Par récurrence sur la longueur du chemin.

*Exemples :*

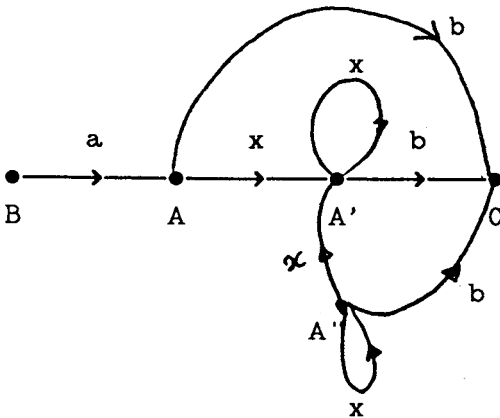


Fig. 1

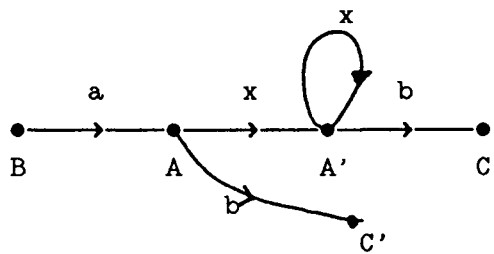


Fig. 2

sont deux dépliages de

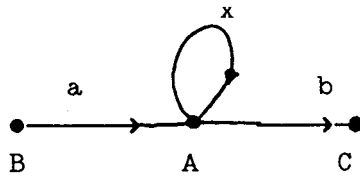


Fig. 3

Pour la figure 1  $v(A)=v(A')=v(A'')=A$ ,  $v(B)=B$ ,  $v(C)=C$ .

Pour la figure 2  $v(A)=v(A')=A$ ,  $v(B)=B$ ,  $v(C)=v(C')=C$ .

Dans la suite nous désignerons du même nom tous les sommets ayant la même image, ceci par commodité d'écriture.

*Remarque* : Pour les automates finis déterministes, le passage de l'automate à l'automate fini minimal correspond à l'inverse d'un dépliage : si  $G_1$  est le graphe sous-jacent d'un automate fini déterministe et  $G_2$  le graphe sous-jacent de l'automate minimal associé, alors  $G_1$  est un dépliage de  $G_2$ .

DÉFINITION 2 : Soient  $(G_1, q_1)$  et  $(G_2, q_2)$  deux graphes pointés avec

$$G_1 = \langle X, Q_1, \delta_1 \rangle \quad \text{et} \quad G_2 = \langle Y, Q_2, \delta_2 \rangle \quad Q_1 \cap Q_2 = \emptyset$$

et soit  $q' \in Q_1$ .

La greffe de  $(G_2, q_2)$  sur le graphe  $(G_1, q_1)$  au sommet  $q'$  est le graphe pointé  $(G, q_1)$  ainsi défini

$$G = \langle X \cup Y, Q, \delta \rangle \quad \text{avec} \quad Q = Q_1 \cup Q_2 \setminus \{q_2\}$$

$$\delta = \delta_1 \cup \delta_2 \mid Q_2 \setminus \{q_2\} \cup \{(q', x, p) \mid (q_2, x, p) \in \delta_2\}$$

$$\cup \{(q, x, q') \mid (q, x, q_2) \in \delta_2\}.$$

Nous notons  $(G_1, q_1)$  greffe  $(G_2, q' = q_2)$

*Remarques* : 1° Cette définition de greffe revient à considérer l'union des deux graphes  $G_1$  et  $G_2$  puis à identifier  $q'$  et  $q_2$ .

2° Cette définition généralise celle usuelle de greffe d'arbres [1].

DÉFINITION 3 : Soit  $(G, q)$  un graphe pointé et  $p$  un sommet de  $G$ . La taille de  $(G, q)$  en  $p$  consiste à éliminer les sommets de  $\text{Acc}(p)$  et les arcs s'y rattachant.

DÉFINITION 4 : Un graphe pointé  $(G, q)$  est un *cactus-fleur* de racine  $q$  si et seulement si

soit il est réduit à un cycle élémentaire.

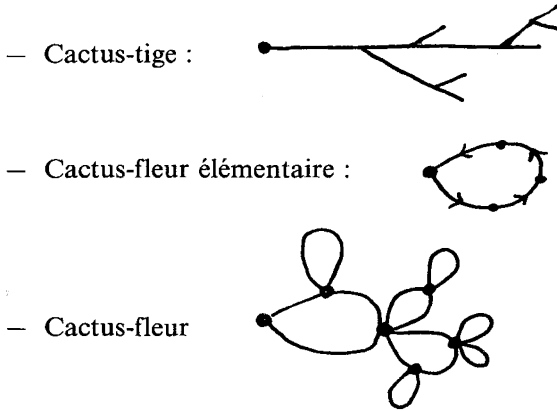
soit il est dans la clôture par greffes de cycles élémentaires de l'ensemble des cycles élémentaires contenant  $q$ .

Un graphe pointé  $(G, q)$  est un *cactus-tige* de racine  $q$  si et seulement si c'est un arbre de racine  $q$ .

Un graphe pointé  $(G, q)$  est un *cactus* s'il est obtenu par greffe de cactus-fleurs sur un cactus-tige de racine  $q$ .

*Nota bene* : Si un cactus-tige est réduit à un sommet sans arc, nous l'appelons *cactus-point*.

*Exemples* :



— Les « flower automaton » définis en [3] sont des cactus.

*Remarque* : Le terme de cactus a été employé par Berge [2] et correspond à une classe de graphes moins restreinte mais qui n'a pas été exploitée.

#### PROPRIÉTÉS DES CACTUS

- (i) Tout sommet est atteint à partir de  $q$  par un chemin.
- (ii) Il n'existe pas deux chemins élémentaires menant d'un sommet à un autre.
- (iii) Deux cycles élémentaires distincts ont au plus un sommet commun.
- (iv) Toute partie fortement connexe du graphe peut-être disconnecté du graphe initial par au plus un sommet.

**PROPOSITION 2** : *Tout graphe pointé vérifiant (i), (ii), (iii), (iv) est un cactus.*

*Preuve :*

– si  $(G, q)$  est un graphe pointé possédant les propriétés (i), (ii), (iii), (iv) alors la taille de  $(G, q)$  en  $p$  donne un graphe pointé  $(G', q)$  qui possède les propriétés (i), (ii), (iii), (iv).

– taillons donc  $(G, q)$  en chaque sommet de connexion d'une composante fortement connexe de  $(G, q)$  [unique point d'après (iv)]. Soit  $(G_1, q)$  le graphe obtenu par cette taille. Il ne possède plus aucun cycle et (i) et (ii) en font un arbre i. e. un cactus tige.

Montrons enfin que chaque composante connexe est un cactus-fleur pointé en son sommet de connexion.

Raisonnons par récurrence sur le nombre de cycles élémentaires.

– S'il est réduit à un cycle élémentaire nous avons un cactus-fleur élémentaire.

– Supposons la propriété vraie pour  $n$  cycles élémentaires. D'après (iii), ils ont au plus un sommet en commun. Montrons qu'il existe un cycle élémentaire ayant un unique sommet commun avec l'ensemble des autres cycles élémentaires.

Dans le cas contraire, soient  $C_1, \dots, C_{n+1}$   $n+1$  cycles élémentaires réduits à leurs sommets.

$$\forall i \in [1, n+1],$$

$$\exists j, k \in [1, n+1] \setminus i \quad \{q_{i,j}\} = C_i \cap C_j \quad \{q_{i,k}\} = C_i \cap C_k$$

le cardinal de  $\{q_{i,j} \mid j \in [1, n+1] \setminus i, \{q_{i,j}\} = C_i \cap C_j\}$  est fini il existe donc un cycle  $C$  passant par cet ensemble i. e. cardinal  $C_i \cap C \geq 2$  ce qui contredit (iii) et (iv).

On peut donc tailler en l'unique point l'un de ces cycles et obtenir une partie fortement connexe ayant  $n$  cycles élémentaires et vérifiant (i), (ii), (iii), (iv), i. e. c'est un cactus-fleur.

La composante fortement connexe est obtenue par greffe du cycle disconnecté sur la partie connexe au point de taille, ce qui donne un cactus-fleur.

Le graphe entier est obtenu par greffe de chaque composante fortement connexe en son point de taille sur le cactus-tige, ce qui donne un cactus.

## RÉSULTAT PRINCIPAL

**PROPOSITION 3 :** *Pour tout graphe pointé  $(G, q)$ , il existe un cactus  $(G', q)$ , de racine  $q$ , qui est un dépliage de  $(G, q)$ .*

*Preuve* : Nous proposons une preuve constructive fondée sur l'algorithme suivant :

*Données* : un graphe  $G = \langle X, \mathbb{Q}, \delta \rangle$  et un sommet  $q_0$ .

$(c, p) = \text{cactus-point } q_0$

*Pour tout*  $q \in \mathbb{Q}$  *faire*

$(c', q') = \text{cactus}(G, q_0, q)$

$(c, p) = (c, p)$  greffe  $(c', p = q')$ .

*fait*

*fin algorithme*

à partir du cactus point  $q_0$ , nous calculons le cactus du graphe issu de  $G$  dont on a supprimé tous les sommets n'appartenant pas à  $\text{Acc}(q_0)$  et  $\text{Coacc}(q)$  puis on greffe ce cactus en  $q$  au cactus déjà obtenu.

Dans la procédure cactus on fait le calcul suivant

Si  $q_d = q_f$  nous aurons un cactus-fleur. Sinon sur chaque chemin élémentaire de  $q_d$  à  $q_f$  nous construisons le cactus de  $q_d$  à  $q_f$  en greffant sur le cactus-tige le cactus-fleur correspondant à chaque sommet  $q_j$  après avoir émondé  $Q$  pour  $q_j$  en ne lisant sur la tige que les sommets déjà traités.

$(c, q) = \text{cactus}(G, q_d, q_f)$

paramètres : le graphe  $G = \langle X, \mathbb{Q}, \delta \rangle$  et deux sommets non forcément distincts  $q_d$  et  $q_f$  de  $\mathbb{Q}$ .

$(c, q) = \text{cactus-point } q_d$

Si  $q_d = q_f$  alors  $(c, q) = \text{cactus-fleur}(G, q_d)$

sinon

● 1 ● *Pour* chaque chemin élémentaire  $ch : \{q_1 = q_d, q_2, \dots, q_s = q_f\}$

*faire*

● 11 ●  $(c', q') = \text{cactus-tige } ch$  de racine  $q_d$

● 12 ● *Pour*  $j = 1$  à  $s$  *faire*

● 121 ● écrire  $G_q$  : sous graphe de  $G$ , dont les sommets sont dans  $\mathbb{Q} \setminus \{q_{j+1}, \dots, q_s\}$ , émondé pour  $q_j$

● 122 ● Si  $G_q \neq \text{cactus-point } q_j$  alors

$(c'_0, q'_0) = \text{Cactus-fleur}(G_{q_p}, q_j)$

$(c', q') = (c', q')$  greffe  $(c'_0, q'_0 = q'_0)$

*fin de si*

*fait*

● 13 ●  $(c, q) = (c, q)$  greffe  $(c, q = q')$

*fait*

*fin de si*

*fin Cactus*

Un cactus-fleur est un cactus dont on a identifié la racine et les sommets  $p$  tels que  $\text{Acc}(p) = \emptyset$  (sommets terminaux), donc on calcule cactus-fleur en dédoublant la racine en un sommet racine et un sommet terminal, et l'on calcule le cactus correspondant à ces deux sommets.

$(c, q) = \text{Cactus-fleur}(G, q_0)$

paramètres :  $G = \langle X, \mathbb{Q}, \delta \rangle$  un graphe émondé en  $q_0 \in \mathbb{Q}$  et  $q_0$ .

● 1 ● Construire  $A = \langle X, \mathbb{Q}_A, \delta_A \rangle$  le graphe ainsi construit

$$\mathbb{Q}_A = \mathbb{Q} \cup \{\bar{q}_0\}$$

$$\delta_A = (\delta \setminus \{(q, x, q_0) \in \delta\}) \cup \{(q, x, \bar{q}_0) \mid (q, x, q_0) \in \delta\}$$

● 2 ●  $(c, q) = \text{cactus-point } q_0$

● 3 ● Pour chaque arc  $(q_0, x, q) \in \delta_A$  faire

$$(c_1, q_1) = ((q_0, x, q), q_0)$$

● 31 ● pour chaque chemin élémentaire de  $q$  à  $\bar{q}_0$  faire

● 311 ● construire  $A'$  issu de  $A$  et émondé par rapport au chemin élémentaire

● 312 ●  $(c', q') = \text{Cactus}(A', q, \bar{q}_0)$

● 313 ●  $(c'', q'') = (c_1, q_1)$  greffe  $(c', q_1 = q')$

● 314 ●  $(c, q) = (c, q)$  greffe  $(c'', q = q'')$

fait

fait

● 4 ● Identifier les sommets  $q_0$  et  $\bar{q}_0$  dans  $(c, q)$

fin cactus-fleur.

## TERMINAISON DE L'ALGORITHME

Elle est due à la remarque suivante : Cactus-fleur  $(G, q; G$  ayant  $n$  sommets) appelle Cactus  $(G', q'_1, q'_2; G'$  ayant  $n$  sommets au plus) qui appelle Cactus-fleur  $(G'', q_1; G''$  ayant au plus  $n-1$  sommets). (En effet le graphe émondé en  $\bar{q}_0$  issu de  $A$  est réduit au sommet  $\bar{q}_0$ ).

## VALIDITÉ DE L'ALGORITHME

Les procédures cactus et cactus-fleurs produisent bien des cactus et il est clair que toutes les opérations effectuées sont bien des dépliages du graphe initial.

*Remarque* : Les cactus ainsi obtenus sont tels que le cactus-tige est un arbre dont seul la racine admet plusieurs fils.

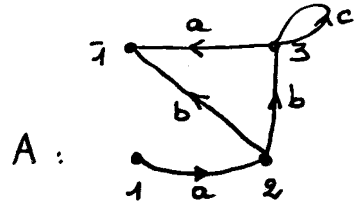
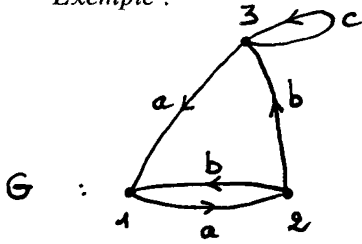
## CONCLUSION

La numérotation locale des sommets dans le sens de parcours dans *Cactus* fait que les cycles sont repoussés le plus loin possible de la racine.

Une numérotation locale inversée aurait fait apparaître les cycles le plus près possible de la racine. Une numérotation globale fixe des sommets du graphe donnerait d'autres cactus équivalents dont les cycles seraient répartis différemment tout comme l'algorithme de MacNaughton et Yamada donne des expressions rationnelles différentes selon la numérotation des sommets choisis.



Exemple :



	$(c, q)$	$(c_1, q_1)$	$(c', q')$	$(c'', q'')$	$A'$
• 2 •	• 1				
• 3 •					
• 311 •					
• 312 •					
• 313 •					
• 314 •					
• 3 •					
• 311 •					
• 312 •					
• 313 •					
• 314 •					
• 4 •					

Fig. 4 : cactus-fleur (G, 1)

## BIBLIOGRAPHIE

1. J. M. AUTEBERT, *Théorie des langages*, Masson, Paris, 1987.
2. C. BERGE, *Graphes et hypergraphes*, Dunod, Paris, 1970.
3. J. BERSTEL et D. PERRIN, *Theory of Codes*, Acad. Press., 1985.
4. S. RAO KOSARAJU, *Analysis of Structured Programs*, J.C.S.S., (74), vol. 9, p. 232-285.
5. R. MACNAUGHTON et H. YAMADA, *Regular Expressions and State Graphs for Automata*, Trans IRE EC-9 39-47, 1960.
6. S. R. SCHWER, *Décidabilité de l'algèbricité des langages associés aux Réseaux de Petri*, Thèse de 3<sup>e</sup> cycle de l'Université Paris-VII, 1985.