

# Théorie des graphes

par **Robert CABANE**

*Professeur de mathématiques spéciales au lycée Michel-Montaigne (Bordeaux)  
Ancien élève de l'École normale supérieure*

<b>1. Généralités</b> .....	AF 205	-	3
1.1 Concepts orientés.....	—		3
1.2 Concepts non orientés.....	—		4
1.3 Hypergraphes.....	—		4
1.4 Suites d'arêtes. Connexité.....	—		5
<b>2. Modes de représentation</b> .....	—		6
2.1 Listes de succession.....	—		6
2.2 Matrice d'adjacence.....	—		7
2.3 Matrice d'incidence.....	—		8
<b>3. Nombres et ensembles caractéristiques des graphes</b> .....	—		9
3.1 Degrés attachés à un graphe.....	—		9
3.2 Rang et nombre cyclomatique.....	—		9
3.3 Nombre de stabilité interne (externe).....	—		10
3.4 Noyau.....	—		10
3.5 Points et ensembles d'articulation. Nombre de connexité.....	—		11
3.6 Centres d'un graphe.....	—		12
3.7 Nombre chromatique.....	—		12
3.8 Indice chromatique.....	—		13
3.9 Problème de l'isomorphisme et problèmes NP-difficiles.....	—		14
<b>4. Chemins et circuits</b> .....	—		14
4.1 Réseaux.....	—		14
4.2 Chemins et circuits eulériens.....	—		15
4.3 Chemins et circuits hamiltoniens.....	—		16
4.4 Recherche du plus court chemin.....	—		18
4.5 Problème du voyageur de commerce (PVC).....	—		20
4.6 Algorithmes distribués.....	—		21
<b>5. Arbres et arborescences</b> .....	—		21
5.1 Définition des arbres.....	—		21
5.2 Arbres couvrants sur un graphe.....	—		22
5.3 Notion d'arborescence.....	—		26
<b>6. Réseaux de transport</b> .....	—		26
6.1 Flots dans un réseau de transport.....	—		27
6.2 Méthode de recherche du flot maximal : algorithme de Ford et Fulkerson.....	—		27
6.3 Problème des flots de coût minimal.....	—		30
<b>7. Graphes bipartis et couplages</b> .....	—		31
7.1 Position du problème et vocabulaire.....	—		31
7.2 Problèmes d'affectation.....	—		32
<b>8. Graphes planaires</b> .....	—		34
8.1 Généralités.....	—		34
8.2 Test de planarité d'un graphe.....	—		35
8.3 Coloriage des graphes planaires.....	—		36
<b>9. Implantations d'un graphe dans un autre</b> .....	—		36
<b>Pour en savoir plus</b> .....	Doc. AF 205		

Chacun a vu une fois au moins un plan de métro, une carte de lignes ferroviaires ou aériennes, un plan électrique ou un circuit électronique ; ainsi, tout le monde sait plus ou moins intuitivement ce qu'est un graphe. Toutefois, entre la vague notion d'un schéma incluant des « points » et des « trajets » unissant ces points et la théorie mathématique des graphes, il y a une longue élaboration des concepts, et en particulier le choix d'une terminologie.

Un **graphe** n'est pas a priori autre chose qu'un ensemble fini de « points » (ses « **sommets** ») reliés par des « traits » ou des « flèches » (ses « **arêtes** »).

Nous trouvons aussi des graphes dans deux activités de l'ingénieur : les diagrammes fonctionnels et les problèmes d'ordonnancement des tâches.

On comprendra mieux l'apparition des graphes dans ce dernier problème par un exemple (figure **A**). Supposons que cinq équipes de football nommées AB, OR, LF, DT, PZ jouent dans un tournoi. A un certain moment, les matchs joués ont été les suivants :

AB-OR 1-0	AB-DT 0-0
AB-LF 1-2	OR-PZ 0-0
LF-DT 1-1	PZ-DT 2-0

On peut modéliser l'ensemble de ces matchs de manière graphique, comme on le voit dans la figure **A**. L'information sur les résultats des matchs n'a pas été reportée, on pourrait l'attacher aux traits (« arêtes » : ce sont les matchs) qui unissent les « sommets » (qui sont ici les équipes). Grâce à cette représentation, on perçoit d'emblée un certain nombre de faits, par exemple le nombre d'équipes en retard.

On trouve aussi une situation de théorie des graphes avec le réseau Internet, dans lequel les sommets sont les fichiers (ou leurs adresses) et les arcs sont les liens entre ces documents.

Un dernier exemple, très différent, peut être trouvé avec la modélisation financière des systèmes internationaux de droits de douane et taxes diverses.

La théorie des graphes est un sujet d'étude relativement récent en Mathématiques. Le texte le plus ancien semble être dû à Euler, en 1736 (dans le problème des ponts de Königsberg, il s'agissait de parcourir les sept ponts sur la rivière Pregel une et seule fois chacun), et le suivant à Monge, en 1781 (à propos d'un problème de déplacement de terre de déblai et de remblai). On remarque ensuite les contributions de Kirkman et Hamilton en 1856 au sujet des parcours désormais appelés chemins hamiltoniens. Mais c'est vers la fin du XIX<sup>e</sup> siècle, avec Cayley, que la théorie des graphes commença à prendre son essor et à fixer sa terminologie.

Du fait de cette jeunesse, le développement théorique, la mise au point des algorithmes, le choix d'une terminologie ne sont pas entièrement achevés. On trouvera, dans cet article, un exposé de quelques méthodes et applications importantes de la théorie des graphes. Le lecteur trouvera tout le nécessaire pour compléter son information en compulsant les ouvrages fondamentaux de Claude Berge, bien que ceux-ci soient très orientés vers la Combinatoire. Les besoins plus spécifiques de l'Algorithmique sont développés dans des ouvrages plus récents qui sont également cités en bibliographie.

Terminons en signalant que la théorie des graphes est parfois « classée » comme sous-discipline de l'Informatique théorique (en anglais « computer science ») en raison de l'importance des algorithmes, parfois comme sous-discipline des Mathématiques en raison de sa proximité avec la Combinatoire. En réalité, la théorie des graphes se trouve « à la croisée des chemins », entre Mathématiques, Recherche opérationnelle, Informatique et Gestion des réseaux de distribution ou de transport, s'appliquant aussi bien en économie, biologie ou en psychologie.

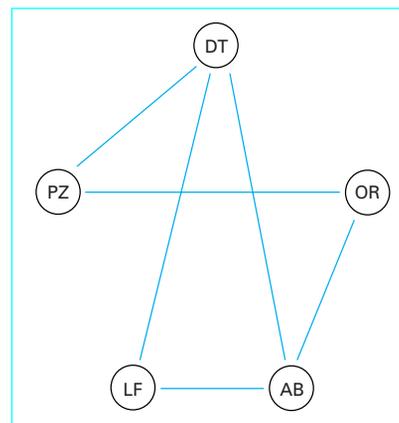


Figure A – Graphe de tournoi

# 1. Généralités

Dans tout ce qui suit, pour un ensemble  $X$  on notera  $|X|$  le nombre de ses éléments.

L'ensemble réduit au seul élément  $x$  sera noté  $\{x\}$ .

L'ensemble des parties d'un ensemble  $X$  sera noté  $\mathcal{P}(X)$ .

La matrice transposée d'une matrice  $A = (a_{ij})$  ( $i$  indice de ligne,  $j$  indice de colonne) sera notée  ${}^tA = (a_{ji})$ .

## 1.1 Concepts orientés

### 1.1.1 Vocabulaire

#### Définition 1.

On appelle **graphe orienté** ou **digraphe**  $G = (X, A)$  la donnée d'un ensemble  $X$ , dont les éléments sont appelés **sommets**, et d'une partie  $A$  de  $X \times X$ , dont les éléments sont appelés **arcs** ou **arêtes**.

En présence d'un arc  $a = (x, y)$  qui peut être noté simplement  $xy$ , on dit que  $x$  est l'**origine** et  $y$  l'**extrémité** de  $a$ , que  $a$  est **sortant** en  $x$  et **incident** en  $y$ , et que  $y$  est un **successeur** de  $x$  tandis que  $x$  est un **prédécesseur** de  $y$ . On dit aussi que  $x$  et  $y$  sont **adjacents**.

On dit que le graphe  $G$  est **sans boucle** si  $A$  ne contient pas d'arête de la forme  $(x, x)$ , c'est-à-dire joignant un sommet à lui-même.

Une arête (arc) se notera  $\bar{u} \in A$ .

On peut aussi définir les graphes d'une manière un peu différente.

**Définition 2.** Soit un graphe  $G = (X, A)$ . Pour tout sommet  $x \in X$  l'ensemble de ses successeurs est noté  $\Gamma x$ . L'application  $\Gamma$  (qui va de  $X$  dans l'ensemble des parties  $\mathcal{P}(X)$ ) est appelée **l'application successeur**.

On note aussi  $G = (X, \Gamma)$  puisque la donnée de  $\Gamma$  permet de connaître toutes les arêtes du graphe.

**Définition 3.** Étant donné un graphe  $G = (X, \Gamma)$ , on note  $\Gamma^{-1}$  l'application de  $X$  dans  $\mathcal{P}(X)$  qui à tout sommet  $x$  associe l'ensemble de ses prédécesseurs ; c'est-à-dire :

$$y \in \Gamma^{-1}x \Leftrightarrow (y, x) \in A \Leftrightarrow x \in \Gamma y$$

#### Définition 4.

On appelle **degré sortant** ou **demi-degré extérieur** d'un sommet  $x$  le nombre d'arcs de la forme  $\bar{u} = (x, y)$  avec  $y \neq x$ , c'est-à-dire le nombre d'éléments de  $\Gamma x \setminus \{x\}$ . On note  $d_s(x)$  ce degré.

On appelle **degré entrant** ou **demi-degré intérieur** d'un sommet  $x$  le nombre d'arcs de la forme  $\bar{u} = (y, x)$  avec  $y \neq x$ , c'est-à-dire le nombre d'éléments de  $\Gamma^{-1}x \setminus \{x\}$ . On note  $d_e(x)$  ce degré.

On appelle **degré** de  $x$  la somme du degré entrant et du degré sortant.

Un sommet de degré entrant non nul et degré sortant 0 est appelé **puits**, tandis qu'un sommet de degré entrant 0 et degré sortant non

**Exemple :** on voit, dans la figure 1, un graphe orienté suivant les définitions antérieures ; il possède quatre sommets (**A,B,C,D**) et six arêtes :

$$(A,A), (B,A), (B,D), (C,D), (D,C), (C,B).$$

Il possède une boucle en **A**. Le sommet **B** a un degré entrant de 1 et un degré sortant de 2, tandis que **D** a un degré entrant de 2 et un degré sortant de 1.

Certains digraphes sont **symétriques**, au sens où toute arête est accompagnée d'une arête unissant les mêmes sommets, mais en sens inverse : si  $(x, y)$  est une arête, alors  $(y, x)$  en est une autre.

**Exemple :** on voit dans la figure 2 un graphe symétrique.

Le concept de graphe symétrique est très proche de celui des graphes non orientés qui sont présentés paragraphe 1.2. En fait, à tout graphe orienté, on peut associer un graphe symétrique, donc un graphe non orienté, en associant à chaque arête  $xy$  une arête  $yx$ .

Nous pouvons aussi extraire d'un graphe une information plus sélective ; par exemple, sur un circuit imprimé, ne considérer que les lignes de masse et, sur une carte de déplacements, ne montrer que les villes de plus de 100 000 habitants.

**Définition 5.** Soit un graphe  $G = (X, A)$  et soit  $X' \subset X$  un sous-ensemble de  $X$ .

Le **sous-graphe** engendré par  $X'$  est  $G' = (X', A')$ ,  $A'$  étant formé des arêtes dont les deux extrémités sont dans  $X'$ .

Si l'on se donne au contraire un sous-ensemble  $A_1$  de  $A$ , le **graphe partiel** engendré par  $A_1$  est  $G_1 = (X, A_1)$ .

Dans le second cas, un sommet de  $G_1$  n'ayant aucune arête incidente appartenant à  $A_1$  est de degré 0 (ou « isolé ») dans  $G_1$ . Si on désire éviter ce fait, il faut considérer le sous-graphe du graphe partiel dont les sommets sont les extrémités des arêtes de  $A_1$ .

**Exemple :** à partir du graphe du réseau routier français, on peut extraire le sous-graphe associé aux préfectures et sous-préfectures ; les routes qui ne relient pas de telles villes seront ignorées. Le graphe partiel des autoroutes contient encore toutes les villes, même celles qui ne sont pas desservies par une autoroute. Le réseau autoroutier strict est un sous-graphe du précédent.

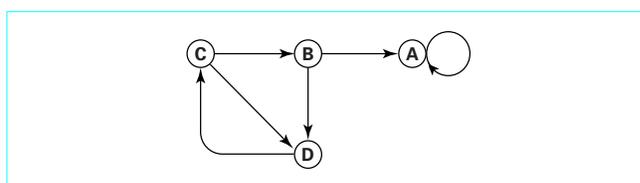


Figure 1 – Exemple de graphe

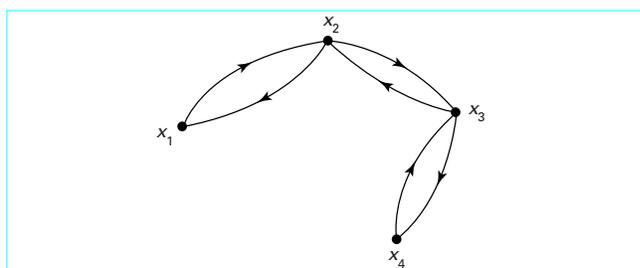


Figure 2 – Graphe symétrique

### 1.1.2 Multigraphes et graphes étiquetés

La notion de graphe peut être généralisée de diverses manières. On peut, par exemple, considérer que plusieurs arcs distincts, et de même sens, peuvent relier un sommet  $x$  à un sommet  $y$ . On dira alors que l'on a affaire à un **multigraphe**. Cela est commode pour de nombreux problèmes. La théorie des graphes a été ainsi utilisée par le mathématicien Polya pour le dénombrement des isomères d'un composé chimique.

En fin de compte, pour définir un multigraphe on ne se contente plus de donner les arêtes, mais on attache à celles-ci une information qui est le nombre d'arêtes de mêmes extrémités ; on peut par exemple noter  $p_{ij}$  le nombre d'arêtes existant entre deux sommets  $x_i, x_j$ . En conséquence, un multigraphe est la donnée de :

$$G = (X, A, F),$$

$F$  étant une application de  $A$  dans  $\mathbb{N}^*$  (ensemble des nombres entiers non nuls) qui, à chaque arête possible, associe le nombre d'arêtes simultanées à considérer. On pourrait même poser  $F(xy) = 0$  si  $x$  et  $y$  ne sont pas adjacents, prolongeant  $F$  à tout le produit cartésien  $X \times X$ .

Si la valeur maximale des  $p_{ij}$  que l'on peut mesurer ( $i = 1, \dots, n ; j = 1, \dots, n$ ) est  $p$ , on dira que l'on a affaire à un  $p$ -graphe.

Plus généralement, on peut envisager d'attacher aux arêtes d'un graphe un nombre non nécessairement entier.

**Définition 6.** Soit un graphe  $G = (X, A)$ .

Un **étiquetage** de  $G$  est la donnée de toute application  $F: X \rightarrow \mathbb{R}$ .

Une **pondération** de  $G$  est la donnée de toute application  $G: A \rightarrow \mathbb{R}$ .

Le graphe **étiqueté** est le triplet  $(X, A, F)$  ; de même le graphe **valué** ou **pondéré** est  $(X, A, G)$ .

## 1.2 Concepts non orientés

### 1.2.1 Vocabulaire

**Définition 7.**

On appelle **graphe non orienté**  $G = (X, A)$  la donnée d'un ensemble  $X$ , dont les éléments sont appelés **sommets** et d'une partie  $A$  de  $X \times X$ , symétrique ( $(x, y) \in A \Leftrightarrow (y, x) \in A$ ), dont les éléments sont appelés arêtes.

En présence d'une arête  $a = (x, y)$  qui peut être notée simplement  $xy$ , on dit que  $x$  et  $y$  sont les **extrémités** de  $a$ , que  $a$  est **incidente** en  $x$  et en  $y$ , et que  $y$  est un **successeur** ou **voisin** de  $x$  (et vice versa).

On dit que le graphe  $G$  est **sans boucle** si  $A$  ne contient pas d'arête de la forme  $(x, x)$ , c'est-à-dire joignant un sommet à lui-même.

La définition des arêtes du graphe non orienté montre qu'on peut considérer celles-ci comme des paires de sommets. Une arête se notera encore  $u \in A$ .

Les applications  $\Gamma$  et  $\Gamma^{-1}$  se définissent comme pour les graphes orientés (cf. définitions 2 et 3). La notion de sous-graphe et graphe partiel se définissent de même (cf. définition 5).

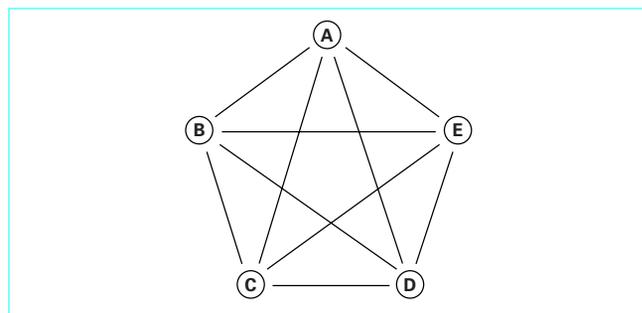


Figure 3 –  $K_5$ , graphe complet

**Définition 8.**

On appelle **degré** d'un sommet  $x$  le nombre d'arêtes aboutissant en  $x$ , l'autre extrémité étant différente de  $x$ . On note  $d(x)$  ce degré.

On dit qu'un sous-graphe  $G = (S, \Gamma_S)$  où  $S \subset X$  est **régulier** de degré  $d$  si le degré de tous les sommets est le même et égal à  $d$ . Cela revient à dire que pour tout  $x \in S$  l'ensemble des successeurs de  $x$ , soit  $\Gamma_S x$ , possède  $d$  éléments.

Un sommet de degré 1 est appelé **terminal** ou **extrémité**.

### 1.2.2 Graphes particuliers

Considérons d'abord les graphes ayant le plus possible d'arêtes.

**Définition 9.**

Un graphe non orienté est dit **complet** si toute paire de sommets de ce graphe est une arête.

On note  $K_n$  le graphe complet à  $n$  sommets. Il est donc pourvu de  $\frac{n(n-1)}{2}$  arêtes.

On appelle **clique** d'un graphe  $G$  tout sous-graphe complet de  $G$ .

La figure 3 donne un exemple de graphe complet à 5 sommets.

Au contraire, on peut considérer des graphes dépourvus d'arêtes.

**Définition 10.**

On appelle **stable** ou **ensemble intérieurement stable** de sommets d'un graphe  $G$  tout sous-graphe de  $G$  sans arêtes.

Comme un sous-graphe s'obtient en ne conservant que les arêtes joignant les sommets sélectionnés, il s'agit en fait d'un ensemble de sommets tels que deux quelconques ne soient pas voisins (adjacents).

## 1.3 Hypergraphes

L'examen d'un plan de métro va nous suggérer une structure un peu plus générale que celle de graphe. En effet, les « correspondances » apparaissent marquées non pas par des traits, mais comme des « bulles » unissant les stations de plusieurs lignes. On peut considérer les « correspondances » comme des arêtes pouvant joindre plus de deux sommets.

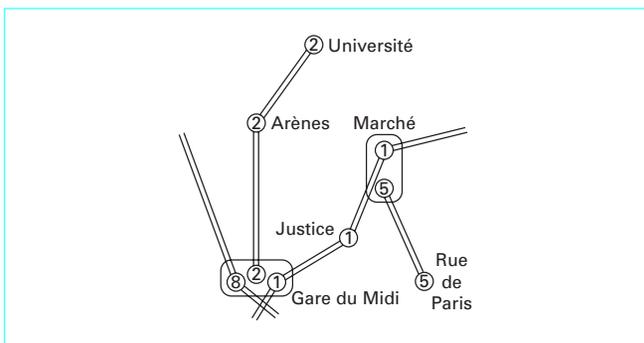


Figure 4 – Hypergraphe

**Définition 11.**

On appelle **hypergraphe** la donnée d'un ensemble fini  $G = \{s_1, \dots, s_n\}$  (les « sommets ») et d'une famille  $H = (A_1, \dots, A_p)$  de parties non vides de  $G$  (les « arêtes ») telle que l'union de cette famille soit  $G$  tout entier.

On appelle **hypergraphe simple** tout hypergraphe  $(G, A)$  tel qu'aucune arête n'en contienne une autre ; soit :

$$A_i \subset A_j \Rightarrow i = j$$

**Exemple :** dans la figure 4, on a représenté les arêtes contenant deux sommets (comme « Université » et « Arènes ») par un trait double (c'est une « interstation sur une ligne »), et les autres arêtes (comme « Marché ») par un contour autour des sommets (ce sont les « correspondances »).

L'avantage de la théorie des hypergraphes est de permettre une parfaite dualité sommets/arêtes, ce que la théorie des graphes ne permet pas.

## 1.4 Suites d'arêtes. Connexité

### 1.4.1 Chemins, chaînes et circuits

**Définition 12.**

On appelle **chemin** dans un graphe orienté toute suite finie de sommets  $(s_1, s_2, \dots, s_k)$  tels que, pour tout  $i$ , il existe une arête de  $s_i$  vers  $s_{i+1}$ .

On appelle **chaîne** dans un graphe non orienté toute suite finie de sommets  $(s_1, s_2, \dots, s_k)$  tels que, pour tout  $i$ , il existe une arête entre  $s_i$  et  $s_{i+1}$ .

On confond le plus souvent la suite des sommets avec la suite des arêtes. Dans le cas des multigraphes, la définition pertinente est plutôt la suite d'arêtes contiguës.

**Définition 13.** Un chemin (ou une chaîne) dans un graphe est **simple** si les arêtes utilisées sont toutes différentes, sinon il est **composé**. Il est **élémentaire** s'il ne rencontre pas deux fois le même sommet. La **longueur** de ce chemin est le nombre d'arêtes suivies (1 de moins que le nombre de sommets visités).

**Définition 14.** On appelle **cycle** ou **circuit** tout chemin ou chaîne fini(e) qui part d'un sommet et aboutit à ce même sommet.

La définition des circuits peut poser un petit problème : si on a un circuit  $(s_1, s_2, \dots, s_k = s_1)$ , alors

$$(s_1, s_2, \dots, s_{k-1}, s_1, s_2, \dots, s_k)$$

est aussi un circuit, mais il n'apporte rien de nouveau. Plus généralement, on convient de ne s'intéresser qu'aux circuits qui n'effectuent aucun retour au point de départ avant leur dernier sommet, soit, dans le cas présent, tels que  $s_j \neq s_1$  pour  $1 < j < k$ .

### 1.4.2 Graphes et sous-graphes connexes

**Définition 15.** Un graphe non orienté est dit **connexe** si deux quelconques de ses sommets peuvent être reliés par une chaîne.

Notons qu'il suffit qu'un sommet particulier soit connecté à tous les autres pour que le graphe soit connexe.

Soit  $x$  un sommet donné et  $C_x$  l'ensemble des sommets pouvant être reliés à  $x$  par une chaîne,  $x$  étant compris dans  $C_x$ .

**Définition 16.** Les éléments du sous-graphe engendré par un ensemble  $C_x$  sont appelés **composantes connexes** du graphe relatives à  $x$ .

Les différentes composantes connexes du graphe  $G$  réalisent une partition de  $X$ , ce qui veut dire :

- $C_x$  n'est pas vide ( $\neq \emptyset$ ) ;
- $C_x \neq C_y$  implique  $C_x \cap C_y = \emptyset$  ;
- $\bigcup_{x \in X} C_x = X$ .

En fin de compte, on peut dire que les composantes connexes sont les classes d'équivalence de sommets pour la relation « être reliés par une chaîne ». On rappelle à ce sujet qu'une relation binaire notée  $\equiv$  est dite d'équivalence si elle vérifie les axiomes suivants :

- a) **réflexivité** :  $\forall x, x \equiv x$  ;
- b) **symétrie** :  $\forall x, y [x \equiv y] \Rightarrow [y \equiv x]$  ;
- c) **transitivité** :  $\forall x, y, z [x \equiv y \text{ et } y \equiv z] \Rightarrow [x \equiv z]$ .

Un graphe est connexe si, et seulement si, il ne possède qu'une composante connexe.

**Exemple :** un circuit électrique est connexe si, et seulement si, en présence d'une tension alternative de fréquence convenable, appliquée en deux points du circuit, il passe un courant non nul.

**Définition 17.** Soit  $G = (X, \Gamma)$  un graphe. On appelle **écart** (ou distance) entre deux sommets  $x, y$  de  $G$  le nombre entier ou infini  $d(x, y)$  tel que :

$$d(x, y) \begin{cases} = 0 & \text{si } x = y \\ = +\infty & \text{si } x \text{ et } y \text{ ne sont pas reliés par un chemin} \\ & \text{soit } (C_x \neq C_y) \\ = \ell & \text{si une chaîne de longueur } \ell \text{ relie } x \text{ et } y \\ & \text{mais qu'aucune chaîne de longueur inférieure} \\ & \text{ne le permet.} \end{cases}$$

**Proposition 1.**

L'écart  $d$  vérifie les relations suivantes :

$$\begin{cases} d(x, x) \geq 0 \\ d(x, y) = 0 \Leftrightarrow x = y \\ d(x, z) + d(z, y) \geq d(x, y) \end{cases}$$

Si le graphe est symétrique (ou non orienté) on a :

$$d(x, y) = d(y, x)$$

et, dans ce cas,  $d$  est une distance au sens topologique de ce terme.

On pourra se reporter à l'article [A 100] *Topologie* (§ 2.1) pour la notion générale de distance. Nous conserverons le nom d'écart pour éviter tout risque de confusion.

### 1.4.3 Graphes et sous-graphes fortement connexes

**Définition 18.** Un graphe orienté est dit **fortement connexe**, si, de tout sommet, on peut trouver un chemin vers n'importe quel autre sommet.

Un graphe fortement connexe est connexe, mais l'inverse n'est pas vrai.

Des deux graphes présentés, le premier (figure 5) est fortement connexe et le second (figure 6) ne l'est pas, car aucun chemin ne peut mener vers  $x_4$ .

Un graphe non fortement connexe peut être partitionné (découpé) en un nombre fini de sous-graphes fortement connexes appelés les **composantes fortement connexes** du graphe initial.

**Exemple :** en reprenant le graphe de la figure 1, on trouve deux composantes fortement connexes, représentées dans la figure 7, dont l'une est réduite à un sommet.

La figure 6 montre un phénomène analogue.

De même que les composantes connexes sont des classes d'équivalence pour la relation d'équivalence entre sommets « être reliés par une chaîne », les composantes fortement connexes sont des classes d'équivalence pour la relation d'équivalence

$x \sim y$  si  $x = y$  ou s'il existe un chemin de  $x$  à  $y$  et un chemin de  $y$  à  $x$ .

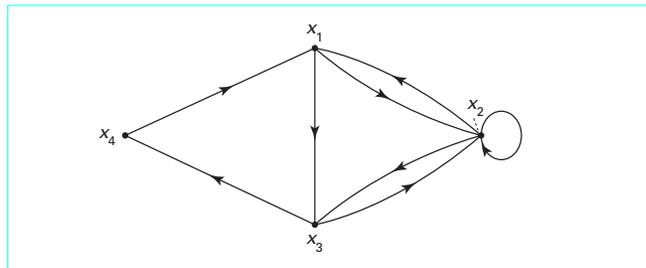


Figure 5 – Graphe fortement connexe

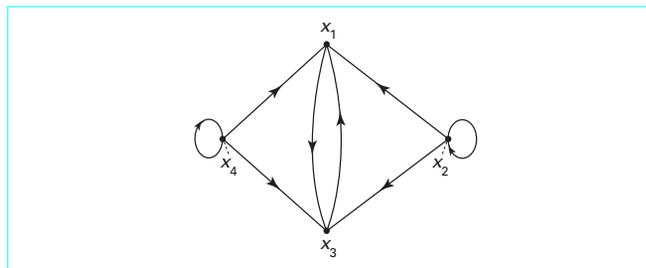


Figure 6 – Graphe non fortement connexe

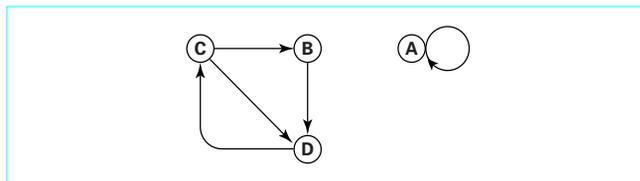


Figure 7 – Deux composantes fortement connexes

Un graphe est connexe (resp. fortement connexe) si, et seulement si, il ne possède qu'une composante connexe (resp. fortement connexe).

Il existe un algorithme simple pour voir si un graphe est fortement connexe : on part d'un sommet, on le « marque » (c'est une sorte de coloriage). On suit toutes les arêtes qui partent de celui-ci et on « marque » les extrémités de ces arêtes. On recommence sur chacune de ces extrémités ; les sommets déjà marqués restent marqués. On arrête le processus lorsqu'on n'a plus rien à faire. On compte le nombre de sommets marqués : le graphe est fortement connexe si, et seulement si, tous les sommets ont été marqués. Sinon, on vient de découvrir un sous-graphe fortement connexe maximal (composante fortement connexe). Physiquement, il suffit d'imaginer les arêtes comme des « tuyaux » par lesquels la « peinture » se répand d'un sommet à l'autre. Ce processus sera évoqué plus loin (§ 5.2) à propos du parcours en largeur d'un graphe.

## 2. Modes de représentation

Dans tout ce qui suit, et jusqu'à la fin de l'article, il ne sera traité que des cas concernant les **graphes finis** (c'est-à-dire ayant un nombre fini d'éléments), qui sont les seuls intéressant l'ingénieur.

Les graphes « abstraits » dont il a été question à la définition 1 doivent être représentables dans un ordinateur, pour permettre un traitement automatique et algorithmique. Il existe différents modes de représentation qui vont être à présent discutés.

### 2.1 Listes de succession

Une première manière consiste à donner, pour chaque sommet, la liste de ses successeurs, ce qui revient à introduire  $X$  et l'application  $\Gamma$  (pour un digraphe, il serait commode de disposer aussi de la liste des prédécesseurs  $\Gamma^{-1}$ ). Le graphe tout entier est alors considéré comme une liste de listes.

Ainsi, le graphe représenté à la figure 1 peut être codé :

— soit comme  $[[A, A], [B, A, D], [C, B, D], [D, C]]$  en considérant que chaque sous-liste donne un sommet et la liste de ses successeurs ;

— soit, plus brièvement, comme  $[[A], [A, D], [B, D], [C]]$  en s'appuyant sur une énumération dans l'ordre alphabétique des sommets (la  $k$ -ième sous-liste représentant alors la liste des successeurs du  $k$ -ième sommet).

Cette représentation est un peu redondante dans le cas des graphes non orientés : elle est assez commode pour parcourir le graphe. L'encombrement de cette représentation est minimal, puisqu'il correspond exactement à la quantité d'information fournie par le graphe (nombre de sommets + nombre d'arêtes).

## 2.2 Matrice d'adjacence

### 2.2.1 Vocabulaire

L'autre manière de représenter les graphes consiste à faire usage des outils de l'algèbre linéaire. Il y a deux démarches possibles, comme on va le voir. Le premier point de vue consiste à considérer chaque arête comme un lien entre deux sommets.

**Définition 19.** Soit un graphe  $G = (X, A)$  ayant  $n$  sommets notés  $x_1, \dots, x_n$ .

La **matrice d'adjacence** de  $G$  est égale à la matrice  $U = (u_{ij})$  de taille  $n \times n$  telle que :

$$u_{ij} = \begin{cases} 1 & \text{si } (i, j) \in A \text{ (c'est-à-dire } (i, j) \text{ est une arête)} \\ 0 & \text{sinon} \end{cases}$$

Plus généralement, on appelle **matrice booléenne** toute matrice carrée dont les éléments sont égaux à 0 ou à 1.

Un graphe orienté quelconque correspond à une matrice booléenne quelconque, tandis qu'un graphe non orienté correspond à une matrice symétrique. L'absence de boucles correspond à une diagonale nulle. Voici la matrice d'adjacence du graphe représenté à la figure 1. :

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

Bien entendu, ce mode de représentation engendre des matrices très « creuses », à moins que le graphe ait beaucoup d'arêtes. Il est donc plutôt inefficace. Cette représentation sera adoptée lorsque le graphe devra être modifié souvent, car, avec des listes de succession, c'est moins facile. En revanche, il n'est pas commode d'ajouter des sommets (en retirer revient à annuler une ligne et une colonne). Nous verrons au paragraphe 2.2.4 que la matrice d'adjacence est un bon outil pour discuter des chemins et des chaînes.

**Généralisation.** Soit un  $p$ -graphe  $G$  (voir § 1.1.2) de sommets  $x_1, x_2, \dots, x_n$ .

La **matrice d'adjacence** associée à  $G$  est la matrice  $U = (u_{ij})$  à  $n$  lignes et  $n$  colonnes,  $u_{ij}$  désignant le nombre d'arcs allant de  $x_i$  à  $x_j$  ( $i =$  numéro de ligne,  $j =$  numéro de colonne).

Plus généralement encore, toute matrice à coefficients réels de taille  $n \times n$  peut être interprétée comme provenant d'un graphe pondéré à  $n$  sommets, chaque coefficient donnant la pondération (ou étiquette) à attacher à chaque arête particulière (pas d'arête si le coefficient est nul).

Si on effectue une symétrie par rapport à la diagonale principale, on obtient la **matrice transposée** ; elle est associée au  $p$ -graphe obtenu en renversant l'orientation de tous les arcs.

On définit également la **matrice complémentaire** par :

$$u'_{ij} = p - u_{ij}.$$

Dans le cas d'un 1-graphe, il suffit de permuter les 0 et les 1. C'est la matrice associée au graphe  $(X, \Gamma')$ , où  $\Gamma'$  est définie par :

$$\forall x \in X \quad \Gamma'x = X - \Gamma x.$$

### 2.2.2 Propriétés de la matrice d'adjacence

Les propriétés de la matrice associée à un graphe découlent immédiatement des définitions.

#### Proposition 2.

Soit  $G$  un graphe sans boucle et soit  $U$  sa matrice d'adjacence.

- a) La somme des éléments de la  $i$ -ième ligne de  $U$  est égale au degré sortant  $d_s(x_i)$  du sommet  $x_i$  de  $G$ .
- b) La somme des éléments de la  $j$ -ième colonne de  $U$  est égale au degré entrant  $d_e(x_j)$  du sommet  $x_j$  de  $G$ .
- c)  $U$  est symétrique si, et seulement si, le graphe  $G$  est symétrique.
- d) Le graphe complet non orienté sans boucle à  $n$  sommets a une matrice d'adjacence  $U$  particulière ; on a :

$$u_{ii} = 0 \text{ pour tout } i,$$

et

$$u_{ij} = u_{ji} = 1 \text{ pour } i \neq j.$$

**Théorème 1.** Soit deux multigraphes ayant même ensemble de sommets ; soit  $U$  et  $V$  leurs matrices d'adjacence.

La matrice  $U + V$  est celle du multigraphe formé par la réunion des arcs des deux premiers.

La matrice  $UV$  est celle du multigraphe obtenue de la façon suivante : d'un sommet  $x_i$  à un sommet  $x_j$  on trace autant d'arcs qu'il existe de chemins distincts allant de  $x_i$  à  $x_j$  et constitués par un arc de  $U$  suivi d'un arc de  $V$ .

**Preuve.**  $\diamond$  La première proposition est évidente.

Examinons la seconde : le nombre de chemins distincts de la forme  $[x_i, x_k, x_j]$  avec  $(x_i, x_k) \in U$  et  $(x_k, x_j) \in V$  est égal à  $u_{ik}v_{jk}$  ; donc, pour aller de  $x_i$  à  $x_j$ , le nombre total de chemins formés d'un arc de  $U$  suivi d'un arc de  $V$  est :

$$\sum_{k=1}^n u_{ik} v_{kj}$$

Il s'agit du produit scalaire de la ligne  $u_i$  et de la colonne  $v_j$ , ce qui est bien le terme général de la matrice-produit  $UV$ .  $\diamond$

### 2.2.3 Opérations sur les matrices booléennes

Dans le cas des 1-graphes, pour lesquels seules comptent les matrices booléennes, il convient de modifier un peu le théorème 1 de manière à avoir un résultat booléen.

**Définition 20.** Soit deux matrices booléennes  $U$  et  $V$ .

La **somme booléenne** de  $U$  et  $V$ , notée  $U \dot{+} V$ , est une matrice dont l'élément  $s_{ij}$  est tel que :

$$s_{ij} = u_{ij} \dot{+} v_{ij} = \begin{cases} 0 & \text{si } u_{ij} = 0 \text{ et } v_{ij} = 0 \\ 1 & \text{si } u_{ij} = 0 \text{ et } v_{ij} = 1 \\ 1 & \text{si } u_{ij} = 1 \text{ et } v_{ij} = 0 \\ 1 & \text{si } u_{ij} = 1 \text{ et } v_{ij} = 1 \end{cases}$$

(on reconnaît ici l'opération « OU logique »).

Le **produit booléen** de  $U$  et  $V$ , noté  $U \dot{\times} V$ , est une matrice dont l'élément  $p_{ij}$  est tel que :

$$p_{ij} = u_{i1}v_{1j} \dot{+} u_{i2}v_{2j} \dot{+} \dots \dot{+} u_{in}v_{nj}$$

où

$$u_{ik}v_{kj} = \begin{cases} 0 & \text{si } u_{ik} = 0 \text{ et } v_{kj} = 0 \\ 0 & \text{si } u_{ik} = 0 \text{ et } v_{kj} = 1 \\ 0 & \text{si } u_{ik} = 1 \text{ et } v_{kj} = 0 \\ 1 & \text{si } u_{ik} = 1 \text{ et } v_{kj} = 1 \end{cases}$$

(on reconnaît ici l'opération « ET logique »).

La **puissance booléenne** se définit à partir du produit.

De manière un peu plus savante, on peut dire que ces opérations reviennent à opérer sur des matrices à coefficients dans l'anneau booléen à deux éléments  $\{0, 1\}$ , muni des lois  $\dot{+}$  et  $\dot{\times}$ .

On peut alors reformuler le théorème 1 pour ce qui concerne la somme des graphes (pour le produit, il n'y a rien à changer).

**Théorème 2.** Soit deux graphes  $G$  et  $G'$ , ayant même ensemble de sommets et ayant des matrices d'adjacence  $U$  et  $V$ .

La matrice  $U \dot{+} V$  est celle du graphe  $G \dot{+} G'$  formé par la réunion des arcs des deux premiers (si une arête  $xy$  se trouve à la fois dans  $G$  et  $G'$ , elle figure une fois dans  $G \dot{+} G'$ ).

En ce qui concerne la puissance booléenne, elle a une interprétation très simple.

**Proposition 3.**

Soit  $U$  la matrice d'adjacence d'un graphe  $G$ ; soit deux sommets  $x_i$  et  $x_j$  de  $G$  et un entier  $p$ . Le coefficient de  $U^p$  (en tant que puissance booléenne) situé à la place  $(i, j)$  est égal à 1 si, et seulement si, il existe au moins un chemin (ou une chaîne) de longueur  $p$  de  $x_i$  vers  $x_j$  dans  $G$ .

**Preuve.**  $\diamond$  On peut procéder par récurrence sur  $p$ : pour  $p = 2$ , on utilise le théorème 1. Si ce résultat est vrai pour la puissance  $p$ -ième, un coefficient de

$$U^{p+1} = UV \text{ avec } V = U^p$$

ne peut être égal à 1 que si

$$\sum_{k=1}^n u_{ik} v_{kj} = 1 \text{ soit } u_{ik} v_{kj} = 1$$

pour au moins une valeur de  $k$ , soit

$$u_{ik} = v_{kj} = 1,$$

ce qui revient à l'existence d'une arête de  $x_i$  vers  $x_k$  et d'un chemin de longueur  $p$  de  $x_k$  vers  $x_j$ . On a donc dans ce cas, et dans ce cas seulement, un chemin de longueur  $p + 1$  de  $x_i$  vers  $x_j$ .  $\diamond$

**2.2.4 Application aux composantes fortement connexes**

**Algorithme de Foulkes**

Pour des graphes d'ordre élevé on se propose de trouver efficacement les composantes fortement connexes grâce à des opérations booléennes sur la matrice d'adjacence  $U$ .

En notant  $I_n$  la matrice-identité de taille  $n$ , on a par opérations booléennes :

$$(I_n \dot{+} U)^2 = I_n \dot{+} U \dot{+} U^2$$

et par récurrence :

$$(I_n \dot{+} U)^r = I_n \dot{+} U \dot{+} U^2 \dot{+} \dots \dot{+} U^r.$$

Le coefficient  $d_{ij}^r$  de cette matrice vaut 1 s'il existe un chemin de longueur inférieure ou égale à  $r$  de  $x_i$  à  $x_j$ , 0 dans le cas contraire.

Soit  $n$  le nombre de sommets, de sorte que tout chemin élémentaire est de longueur inférieure ou égale à  $n - 1$ . La matrice  $(I_n \dot{+} U)^{n-1}$  donne tous les chemins élémentaires ou non de longueur inférieure ou égale à  $n - 1$ . Comme tout chemin est la réunion de chemins élémentaires, la matrice  $(I_n \dot{+} U)^{n+p-1}$  (avec  $p \geq 1$ ) est nécessairement égale à la matrice  $(I_n \dot{+} U)^{n-1}$ . Il suffit donc de calculer cette dernière matrice pour savoir si un chemin existe entre un point  $x_i$  et un autre point  $x_j$ . En réalité, la dernière puissance à calculer de  $I_n \dot{+} U$  correspond au chemin élémentaire le plus long possible dans le graphe, et son numéro est exactement le diamètre du graphe (§ 3.5.1, définition 29).

**Amélioration**

Il s'agit donc de calculer des puissances booléennes d'une matrice donnée. L'algorithme d'exponentiation binaire permet d'accélérer notablement les calculs : on cherche seulement :

$$(I_n \dot{+} U)^2, (I_n \dot{+} U)^4, \dots, (I_n \dot{+} U)^{2^k}$$

et on s'arrête lorsque :

$$(I_n \dot{+} U)^{2^{k+1}} = (I_n \dot{+} U)^{2^k}$$

puisque cette égalité signifie qu'aucun chemin nouveau n'est créé en doublant la longueur. Dès lors, une composante fortement connexe correspond, dans la matrice calculée, à une sous-matrice carrée maximale et remplie de 1.

**2.3 Matrice d'incidence**

La seconde idée de représentation matricielle se fonde sur la relation d'incidence entre arêtes et sommets.

**Définition 21.** Soit  $u_1, u_2, \dots, u_m$ , les arêtes d'un graphe orienté **sans boucle**  $G = (X, \Gamma)$  de sommets  $x_1, x_2, \dots, x_n$ .

On appelle **matrice d'incidence aux arcs** de  $G$  la matrice  $M(G)$  d'éléments  $m_{ij}$  ( $i = 1, 2, \dots, n$  et  $j = 1, 2, \dots, m$ ) avec :

$$m_{ij} = \begin{cases} 1 & \text{si } x_i \text{ est l'extrémité initiale de } u_j \\ -1 & \text{si } x_i \text{ est l'extrémité terminale de } u_j \\ 0 & \text{si } x_i \text{ n'est pas une extrémité de } u_j \end{cases}$$

Pour un graphe non orienté  $G$ , sans boucle, la **matrice d'incidence aux arêtes** est de même définie comme :

$$M = (m_{ij})$$

avec  $m_{ij} = 1$  si  $x_i$  est une extrémité de  $u_j$ ,  $m_{ij} = 0$  sinon.

En fait, quand on considère un graphe non orienté  $G$ , il apparaît, dans certains cas, préférable d'orienter arbitrairement ses arêtes et de considérer la matrice d'incidence aux arcs (dont les coefficients sont égaux à la matrice d'incidence aux arêtes aux signes près).

En ce qui concerne les multigraphes, il n'y a aucune modification à faire pour créer une matrice d'incidence : il suffit de dupliquer les colonnes autant de fois qu'il y a d'arcs (ou d'arêtes) joignant les mêmes sommets.

Le lien entre les deux matrices associées à un graphe est assez simple.

**Théorème 3.** Soit  $M$  la matrice d'incidence aux arcs et  $U$  la matrice d'adjacence d'un multigraphe  $G$  orienté sans boucle ; alors  $M^t M + U$  est une matrice diagonale ; sur la diagonale de celle-ci on trouve les degrés des sommets de  $G$ .

Avec la matrice  $N$  d'incidence aux arêtes, c'est  $N^t N - U$  qui est égale à la matrice diagonale des degrés.

**Preuve.**  $\diamond$  Quand on fait le produit  $M^t M$  on multiplie les lignes de  $M$  par les colonnes de  $^t M$ , donc les lignes de  $M$  entre elles, ce qui revient à calculer le produit scalaire des diverses lignes de  $M$ . Notons que le nombre de lignes de  $M$  est bien égal à celui de  $M^t M$  et de  $U$ . Quand on multiplie (scalairement) une ligne de  $M$  par elle-même, on trouve exactement le degré du sommet associé puisque

le nombre de 1 ou de -1 est égal au nombre d'arêtes entrantes ou sortantes, donc au degré de ce sommet. Autrement, le produit scalaire de deux lignes de numéros  $i$  et  $j$  de  $M$  (associé à deux sommets différents  $x_i$  et  $x_j$  de  $G$ ) ne peut être non nul que si on trouve 1 et -1 dans la même colonne sur ces deux lignes. Cela se produira pour toute arête joignant  $x_i$  à  $x_j$  ou vice versa. Le produit scalaire recherché est donc égal à l'opposé du nombre d'arêtes joignant  $x_i$  et  $x_j$  (dans n'importe quel sens), d'où le résultat.

Pour un graphe non orienté, on a le même phénomène au signe près.  $\diamond$

La matrice  $M^t M$  est appelée **matrice d'adjacence étendue**. Elle a la propriété d'avoir ses lignes et ses colonnes de somme nulle ; c'est donc une matrice singulière. On peut préciser cet aspect en utilisant la théorie des déterminants (pour les règles de calcul des déterminants, voir par exemple l'article [AF 86] *Calcul matriciel*, § 6.4).

**Théorème 4.** Toute matrice d'incidence d'un graphe (orienté ou non) sans boucle est **totalelement unimodulaire**, c'est-à-dire que toute matrice carrée extraite a son déterminant (appelé **mineur**) égal à 0, +1 ou -1.

**Preuve.**  $\diamond$  On démontre ce théorème par récurrence, se basant sur le fait que la matrice d'incidence a, dans chaque colonne, des coefficients tous nuls sauf un 1 et un -1 (ou deux 1 dans le cas non orienté). Pour une taille égale à 2, on ne peut essentiellement trouver que :

$$\begin{vmatrix} 1 & \varepsilon \\ 0 & \pm 1 \end{vmatrix} = \pm 1 \text{ et } \begin{vmatrix} 1 & -1 \\ 1 & -1 \end{vmatrix} = 0 \text{ (au signe près).}$$

Si le théorème est prouvé pour les mineurs de taille inférieure à  $q$ , on considère un mineur  $\Delta$  de taille  $q$ . Quitte à échanger des lignes, des colonnes et multiplier certaines lignes, colonnes par -1, on peut supposer que :

$$\Delta = \begin{vmatrix} 1 & \dots & 1 & 0 & \dots & 0 \\ -1 & \dots & \cdot & \cdot & \dots & \cdot \\ 0 & & & & & \\ \vdots & & \vdots & \vdots & & \vdots \end{vmatrix}$$

Si deux colonnes identiques ou opposées apparaissent, on a  $\Delta = 0$ . Sinon, on peut supposer que

$$\Delta = \begin{vmatrix} 1 & 1 & \dots & 1 & 0 & \dots & 0 \\ -1 & 0 & \dots & 0 & \cdot & \dots & \cdot \\ 0 & -1 & \dots & \cdot & \cdot & \dots & \cdot \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \end{vmatrix}$$

On soustrait alors la première colonne à celles qui suivent tant qu'il se trouve un 1 en première ligne, ce qui amène :

$$\Delta = \begin{vmatrix} 1 & 0 & \dots & 0 & 0 & \dots & 0 \\ -1 & 1 & \dots & 1 & \cdot & \dots & \cdot \\ 0 & -1 & \dots & \cdot & \cdot & \dots & \cdot \\ \vdots & \vdots & & \vdots & \vdots & & \vdots \end{vmatrix} = \begin{vmatrix} 1 & \dots & 1 & \dots \\ -1 & \dots & \cdot & \dots \\ \vdots & \vdots & \vdots & \vdots \end{vmatrix}$$

et le déterminant obtenu est de taille  $p$ , avec exactement un 1 et un -1 par colonne. L'hypothèse de récurrence permet de conclure.  $\diamond$

Le lecteur trouvera dans l'article [D 61] *Propriétés générales des réseaux électriques linéaires à constantes localisées* une illustration de la méthode de la matrice d'incidence aux arcs.

### 3. Nombres et ensembles caractéristiques des graphes

Dans ce paragraphe, nous nous attacherons à définir quelques concepts et grandeurs intrinsèquement attachées à un graphe vu dans son ensemble. La discussion est donc essentiellement **topologique**.

#### 3.1 Degrés attachés à un graphe

**Définition 22.** On appelle **degré total** d'un graphe non orienté la somme des degrés de tous ses sommets.

On appelle **degré maximal** d'un graphe non orienté le plus grand des degrés de ses sommets.

Pour les graphes orientés, ces notions ont un sens à condition de considérer le graphe non orienté sous-jacent.

La connaissance du degré maximal est requise pour préparer une représentation du graphe par listes d'adjacence. C'est aussi la plus grande somme possible des lignes et colonnes de la matrice d'adjacence. En ce qui concerne le degré total, c'est une valeur très simple.

**Théorème 5.** Pour tout graphe (ou multigraphe) non orienté, le degré total est égal au double du nombre d'arêtes.

**Preuve.**  $\diamond$  On prend la matrice d'incidence aux arêtes ; ses coefficients valent 0 ou 1. On fait la somme de tous ses coefficients.

Si on somme ligne par ligne, on obtient le nombre d'arêtes pour chaque sommet (c'est-à-dire le degré), donc le degré total.

Si on somme colonne par colonne, on trouve 2 fois le nombre de colonnes, donc le double du nombre d'arêtes.  $\diamond$

#### 3.2 Rang et nombre cyclomatique

Nous considérons ici un  $p$ -graphe  $G$  ayant  $n$  sommets,  $m$  arêtes et  $r$  composantes connexes.

**Avertissement.** La suite de ce paragraphe fait un usage assez important de concepts d'algèbre linéaire matricielle. Le lecteur pourra se reporter utilement aux articles [AF 85] *Algèbre linéaire* et [AF 86] *Calcul matriciel*.

**Définition 23.** Pour un graphe orienté  $G$ , on appelle **rang**, noté  $\rho(G)$ , le rang de la matrice d'incidence aux arcs de  $G$ .

**Proposition 4.**

Le rang d'un graphe est égal au rang de la matrice d'adjacence étendue.

**Preuve.**  $\diamond$  A priori  $M^t M$  n'a pas un rang supérieur au rang de  $M$ , mais, de plus, si on considère un vecteur  $X$  de  $\mathbb{R}^n$  tel que :

$$M^t M X = 0$$

alors :

$${}^t X M^t M X = \|{}^t M X\|^2 = 0 \text{ (norme euclidienne)}$$

d'où  ${}^t M X = 0$ .

Ainsi, les noyaux de  $M^t M$  et  ${}^t M$  coïncident et ces matrices ont bien même rang.  $\diamond$

**Théorème 6.** Le rang d'un  $p$ -graphe non orienté  $G$  est égal au nombre de sommets privé du nombre de composantes connexes, ce qui s'écrit :

$$\rho(G) = n - r$$

**Preuve.**  $\diamond$  On note au préalable que  $n - \rho(G)$  est exactement la dimension commune des noyaux de  ${}^t M$  et  $M^t M$  (cela résulte du théorème du rang, exposé dans l'article [AF 85], *Algèbre linéaire*, § 2.2). Considérons d'abord le cas où  $G$  est connexe, et montrons que le noyau  $\text{Ker } {}^t M$  est de dimension 1. On connaît au moins un vecteur de ce noyau ; c'est le  $n$ -vecteur de composantes

$$V = (1, 1, \dots, 1)$$

puisque toute ligne de  ${}^t M$  possède un 1, un  $-1$  et  $n - 2$  zéros.

Soit un vecteur  $Z = (z_1, \dots, z_n)$  de  $\text{Ker } {}^t M$  (il faut le voir comme un vecteur colonne). On a donc pour tous sommets  $x_i$  et  $x_j$  adjacents :

$$z_i - z_j = 0.$$

Ainsi, les composantes de  $Z$  attachées à des sommets adjacents coïncident. Le long d'une chaîne, on n'a que des sommets adjacents, ce qui donne des composantes identiques pour  $Z$ . Comme le graphe est supposé connexe, le vecteur  $Z$  est proportionnel à  $V$ .

Si  $G$  a plusieurs composantes connexes, on peut regrouper les sommets par composantes connexes. La même démarche va donner plusieurs vecteurs dans le noyau de  ${}^t M$ , tous de la forme  $(0, \dots, 0, 1, \dots, 1, 0, \dots, 0)$ , et les autres vecteurs du noyau ont des composantes égales pour des sommets liés par une chaîne, ce qui conduit à la dimension  $r$ .  $\diamond$

**Définition 24.** On appelle **nombre cyclomatique** du  $p$ -graphe  $G$  l'entier :

$$v(G) = m - \rho(G) = m - n + r.$$

Les nombres  $\rho(G)$  et  $v(G)$  peuvent avoir une **signification physique** : dans un réseau électrique,  $v$  est égal au nombre de mailles pouvant servir de base (linéairement indépendantes) pour évaluer les courants des mailles, tandis que  $\rho$  est égal au nombre de paires de nœuds indépendantes requises pour évaluer les différences de potentiel.

### 3.3 Nombre de stabilité interne (externe)

**Définition 25.** Le nombre d'éléments du plus grand sous-ensemble intérieurement stable d'un graphe  $G$  est appelé **nombre de stabilité interne** et noté  $\alpha(G)$ .

**Définition 26.**

Un sous-ensemble  $A$  d'un graphe  $G$  est dit **extérieurement stable** ou **absorbant** si tout sommet n'appartenant pas à  $A$  est adjacent à au moins l'un des sommets de  $A$ .

Le **nombre de stabilité externe**, noté  $\beta(G)$ , est la taille minimale d'un ensemble absorbant.

La recherche d'un sous-ensemble stable maximal (resp. absorbant minimal) dans un graphe est un problème NP-difficile (voir paragraphe 3.9).

### 3.4 Noyau

**Définition 27.** Soit un graphe  $G = (X, \Gamma)$  : un sous-ensemble  $N \subset X$  est appelé **noyau** de  $G$  si  $N$  est stable à la fois intérieurement et extérieurement ; cela s'écrit :

- pour tout  $x \in N$ ,  $N \cap \Gamma x = \emptyset$  (les successeurs de  $N$  sont hors de  $N$ ) ;
- pour tout  $x \notin N$ ,  $N \cap \Gamma x \neq \emptyset$  (tout élément de  $X \setminus N$  a un successeur dans  $N$ ).

Dans un graphe orienté, un noyau est un ensemble stable et absorbant. Déterminer si un graphe possède un noyau est un problème NP-difficile. Pour ces deux notions, voir les livres de C. Berge [6], [7], [8], [9].

**Exemple :** dans la figure 8, le sous-ensemble  $\{A, C, D, G\}$  est un noyau : en effet, deux quelconques de ses sommets ne sont pas adjacents, et tout autre sommet  $B, E, F$  ou  $H$  est relié au moins à l'un des sommets de  $\{A, C, D, G\}$  par un arc ayant son extrémité initiale en  $B, E, F$  ou  $H$ . Ce sous-ensemble est le seul noyau du graphe.

Le noyau d'un graphe possède un certain nombre de propriétés qui sont énoncées ci-après sans démonstration.

**Proposition 5.**

a) Un graphe quelconque n'admet pas nécessairement un noyau, mais il peut également en admettre plusieurs.

b) Si  $N$  est un noyau, on a :

$$\alpha(G) \geq |N| \geq \beta(G).$$

c) Un graphe symétrique sans boucle admet toujours un noyau.

La notion de noyau est utilisée dans la théorie des jeux.

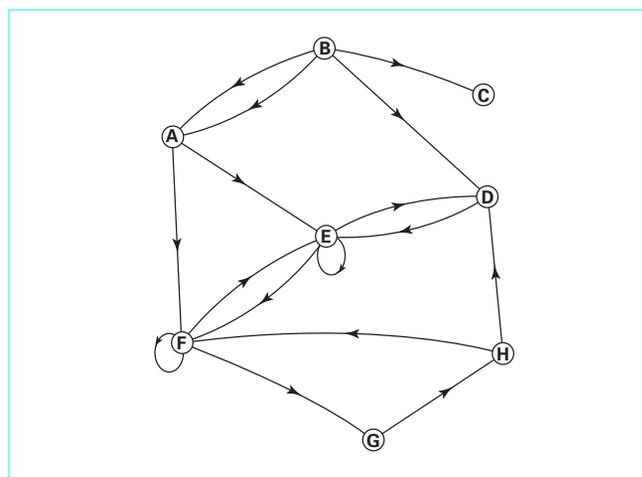


Figure 8 – Exemple de graphe possédant un noyau

### 3.5 Points et ensembles d'articulation. Nombre de connexité

Les notions définies dans ce paragraphe ne concernent que les graphes non orientés.

#### 3.5.1 Points et ensembles d'articulation

##### Définition 28.

Dans un graphe connexe, on dit qu'un sommet  $x$  est un **point d'articulation** si le sous-graphe obtenu en supprimant  $x$  n'est pas connexe.

Si ce point est relié par une seule arête à chacune des composantes du sous-graphe, on dit qu'on a un **point d'articulation simple**.

**Théorème 7.** Dans un graphe connexe, un sommet  $x$  est un point d'articulation si, et seulement si, il existe deux sommets  $y$  et  $z$  tels que toute chaîne reliant  $y$  à  $z$  passe par  $x$ .

**Preuve.**  $\diamond$  Si  $x$  est un point d'articulation, les successeurs de  $x$  ne sont pas dans la même composante connexe du sous-graphe  $G'$  de  $G$  obtenu en retirant  $x$ . On considère alors deux successeurs  $y$  et  $z$  de  $x$  n'appartenant pas à la même composante connexe de  $G'$ .

Réciproquement, si  $y$  et  $z$  sont de la sorte, la suppression de  $x$  déconnecte  $y$  de  $z$ , donc fait apparaître au moins deux composantes connexes.  $\diamond$

##### Définition 29.

Pour un sommet  $x$  d'un graphe connexe  $G = (X, A)$ , on appelle **excentricité** le nombre :

$$e(x) = \text{Max}_{y \in X} d(x, y).$$

On appelle **rayon** de  $G$  la plus petite excentricité possible, soit :

$$R(G) = \text{Min}_{x \in X} e(x).$$

On appelle **diamètre** de  $G$  la plus grande excentricité, soit :

$$\Delta(G) = \text{Max}_{x \in X} e(x) = \text{Max}_{x, y \in X} d(x, y).$$

On a évidemment, par l'inégalité triangulaire, l'encadrement :

$$R(G) \leq \Delta(G) \leq 2R(G).$$

La recherche d'un faible diamètre se trouve dans les problèmes de construction de réseaux de communications rapides, pour lesquels un degré élevé doit être combiné avec le plus petit diamètre possible.

**Exemple :** le graphe de Petersen (quatrième graphe de Micielski)  $M_4$ , représenté à la figure 9, possède 10 sommets, tous de degrés égaux à 3 et son diamètre est 2. On peut dire qu'on peut traverser ce graphe en « un ou deux coups » quels que soient les sommets choisis.

On appelle **graphes de Moore** les graphes de diamètre 2 et réguliers de degré  $d$  (donc avec  $d^2 + 1$  sommets). On peut montrer que les seules valeurs possibles de  $d$  sont 2, 3, 7, 57 ; le graphe de Petersen correspond à  $d = 3$ . La valeur  $d = 7$  est réalisable ; on ignore si  $d = 57$  l'est.

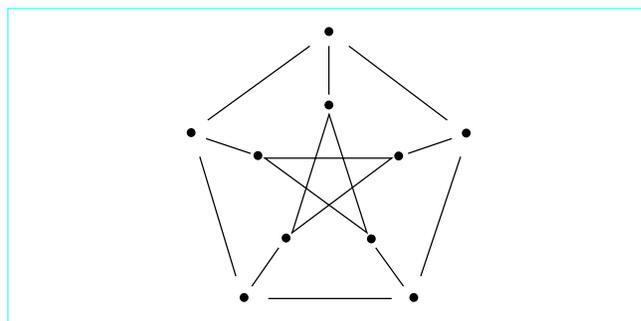


Figure 9 –  $M_4$ , un graphe facile à traverser

On dit qu'un graphe  $G$  est **inarticulé** ou **étoile** s'il est connexe et n'a pas de point d'articulation ; l'idée en est tirée des réseaux de communications : un « bon » réseau doit supporter une coupure isolée (pas de point d'articulation).

Plus généralement, la question se pose de savoir à partir de combien de coupures le réseau n'est plus connexe.

**Définition 30.** Soit  $G$  un graphe connexe et un sous-ensemble non vide  $A \subset X$ .

$A$  est un **ensemble d'articulation** si le sous-graphe engendré par le complément de  $A$  par rapport à  $X \setminus A$  n'est pas connexe.

Enfin, il existe une notion « duale » de la précédente : au lieu d'enlever des sommets, on enlève des arêtes.

**Définition 31.** Une arête  $\bar{u}$  d'un graphe est appelée **isthme** si sa suppression crée un graphe partiel  $(X, A \setminus \{\bar{u}\})$  avec au moins deux composantes connexes ayant chacune au moins une arête.

#### 3.5.2 Nombre de connexité

Dans un graphe  $G = (X, A)$  tel que  $|X| = n$ , tout ensemble de  $n - 1$  éléments constitue un **ensemble d'isolation**.

**Définition 32.** Le **nombre de connexité** d'un graphe  $G$  est le nombre minimal d'éléments d'un ensemble qui est d'articulation ou d'isolation pour  $G$ . On note  $\bar{\omega}(G)$  ce nombre.

Pour tout entier  $h$  positif, on dit que le graphe  $G$  est  **$h$ -connexe** si :

$$\bar{\omega}(G) \geq h$$

##### Proposition 6.

a) Dans un graphe  $h$ -connexe, le degré de tout sommet est supérieur ou égal à  $h$ .

b) Si  $G$  est un graphe  $h$ -connexe, le sous-graphe  $G'$  obtenu par la suppression d'un sommet est  $(h - 1)$  connexe.

c) Si  $G$  est un graphe  $h$ -connexe, le sous-graphe  $G'$  obtenu par la suppression de  $k$  arêtes ( $k < h$ ) est  $(h - k)$  connexe.

**Théorème 8 (de Whitney).** Un graphe est  $h$ -connexe si, et seulement si, on peut relier deux sommets arbitraires  $x_i$  et  $x_j$  par au moins  $h$  chaînes élémentaires distinctes, n'ayant en commun que les sommets  $x_i$  et  $x_j$ .

### 3.6 Centres d'un graphe

**Définition 33.**

On appelle **centre** tout sommet d'excentricité minimale.  
 On appelle **sommet périphérique** tout sommet d'excentricité maximale.

On démontre qu'un sommet périphérique n'est jamais un point d'articulation.

Il existe toujours au moins un centre et au moins un point périphérique dans un graphe (fini) ; toutefois, pour les graphes orientés ou non connexes, on doit prendre en compte des valeurs infinies de l'excentricité. Il peut exister plusieurs centres, plusieurs points périphériques. Bien entendu, un sommet périphérique d'excentricité infinie fournit très peu d'information (il n'est pas accessible depuis tous les autres sommets).

Le rayon du graphe n'est autre que l'excentricité de l'un de ses centres.

Dans le graphe de la figure 1 nous trouvons un point périphérique en **A** (excentricité infinie : aucun arc ne sort de ce point). Les autres excentricités sont 2 pour **B**, **C**, et 3 pour **D**. Ainsi, **B** et **C** sont des centres.

On peut alors démontrer les résultats suivants, tous relatifs à un graphe fini  $G = (X, \Gamma)$ , ayant  $n$  sommets et de degré maximal  $d > 1$ .

**Théorème 9.** Soit  $R$  le rayon de  $G$ . On a :

$$(R + 1)^d \geq n(d - 1) + 1 \Leftrightarrow R \geq \frac{\ln(nd - n + 1)}{\ln d} - 1$$

**Théorème 10.** Si  $G$  est total (c'est-à-dire tout couple de sommets est relié dans au moins une direction), son rayon est inférieur à 2. En outre, tout sommet  $x_0$  de degré  $d$  exactement est un centre.

### 3.7 Nombre chromatique

#### 3.7.1 Définition

On veut placer des pastilles de couleur sur les sommets du graphe non orienté, de sorte que deux sommets reliés par une arête ne soient pas de la même couleur. C'est évidemment possible avec un nombre de couleurs élevé ; il est alors naturel de se demander quel est le plus petit nombre de couleurs requis.

**Définition 34.** Le plus petit nombre de couleurs différentes nécessaires pour colorier les sommets de telle sorte que deux sommets adjacents ne soient pas porteurs de la même couleur s'appelle le **nombre chromatique** du graphe ; on le note  $\gamma(G)$  (parfois  $\chi(G)$ ).

Un graphe  $G$  tel que  $\gamma(G) \leq k$ , donc coloriable en  $k$  couleurs, est dit  $k$ -chromatique.

Une  $k$ -coloration des sommets est une partition de l'ensemble des sommets de  $G$  en  $k$  ensembles (intérieurement) stables.

Les bicolorations sont faciles à trouver, au contraire des  $k$ -colorations pour  $k \geq 3$ . Elles sont en rapport avec les couplages, notion qui sera étudiée au paragraphe 7.1.

**Théorème 11.** Un graphe est 2-chromatique si, et seulement si, il n'a aucun cycle de longueur impaire.

**Preuve.**  $\diamond$  On peut supposer que le graphe est connexe (autrement, on colorie les composantes connexes les unes après les autres).

■ Un cycle de longueur impaire s'oppose visiblement à toute bicoloration.

■ Réciproquement, supposons que le graphe  $G$  ne possède pas de cycle de longueur impaire.

● Prenons une **méthode de « parcours en hauteur »** du graphe (paragraphe 5.2) (dite aussi « **marche à l'aveuglette** »). On choisit un sommet quelconque  $x_1$  de  $G$ , et on le colorie en rouge. On choisit un successeur  $x_2$  quelconque de  $x_1$ , et on le colorie en bleu. Puis on cherche un successeur  $x_3$  de  $x_2$ . Si  $x_3 = x_1$ , on a un cycle de longueur 2. Sinon, on colorie  $x_3$  en rouge et on recommence. Ce processus va s'arrêter soit avec une absence de successeur, soit avec la rencontre d'un sommet déjà colorié, donc un cycle de longueur paire, pour lequel la bicoloration est adéquate. Cela fait, on prend n'importe lequel des sommets déjà visités pour lequel il reste des successeurs non coloriés, et on procède de la même manière. Il est clair que l'algorithme aboutit au résultat cherché, avec un nombre d'étapes au moins égal au nombre  $n$  de sommets du graphe et au plus égal au nombre  $m$  d'arêtes.

● On peut aussi suivre un « **parcours en largeur** » (**méthode de la « tache d'huile »**) : on choisit un sommet quelconque  $x_1$  de  $G$ , et on le colorie en rouge. Tous les successeurs de  $x_1$  seront coloriés en bleu. Pour chacun de ces sommets, on applique la même opération, mais avec les couleurs inversées. Cette méthode est, en un sens, plus naturelle que la précédente, car le parcours en largeur calcule les écarts (§ 1.4) des divers sommets à  $x_1$  et choisit la couleur suivant la parité de l'écart.  $\diamond$

Pour quelques graphes particuliers, la détermination du nombre chromatique est aisée.

**Exemple :**

- Pour le graphe **complet**,  $K_n$  on trouve  $\gamma(K_n) = n$ .
- Pour le graphe **cyclique**  $C_n$  formé de  $n$  sommets arrangés en un  $n$ -cycle, deux ou trois couleurs suffisent selon la parité de  $n$  (quand  $n$  est impair on colorie un sommet particulier en rouge et les suivants alternativement en jaune et en bleu ; la figure 10 illustre cela.

Déterminer si un graphe est  $k$ -chromatique est un problème NP-difficile pour toute valeur fixée de  $k$ , si  $k$  vaut au moins 3. Le paragraphe 3.9 traite des problèmes NP-difficiles. On pourra aussi se reporter à l'ouvrage [29] pour plus de détails sur la coloration.

On s'intéresse également aux graphes parfaits, c'est-à-dire tels que pour tout sous-graphe  $H$  de  $G$  le nombre chromatique de  $H$  est égal à la taille d'une clique maximale de  $H$  (nombre qui se note  $\omega(G)$ ). Voir à ce sujet l'ouvrage [22].

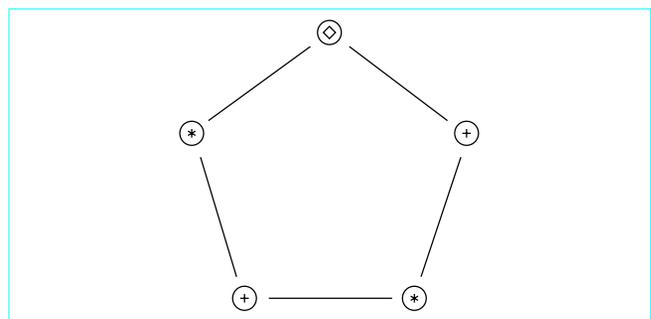


Figure 10 – Coloriage d'un cycle

Il est possible d'encadrer grossièrement l'indice chromatique.

**Théorème 12.** Soit un graphe  $G$  ayant  $n$  sommets et de degré maximal  $d$ . On a alors :

$$\begin{cases} n \leq \alpha(G)\gamma(G) \\ \gamma(G) \leq d + 1 \end{cases}$$

**Preuve.**  $\diamond$  L'ensemble des sommets coloriés dans une couleur donnée est un stable de  $G$ , donc l'union de tous ces stables donne tous les sommets. Chaque stable a au plus  $\alpha(G)$  éléments. D'autre part, si on prend  $d + 1$  couleurs, on peut colorier un sommet quelconque, puis tous ses successeurs et ainsi de suite sans jamais avoir de conflit.  $\diamond$

### 3.7.2 Exemple : la carte de géographie

Un exemple classique est celui des cartes de géographie politique. Dessinons une partie de la carte du monde comprenant  $X$  pays (sans enclave), et posons que  $(x_i, x_j) \in U$  si les pays  $x_i, x_j$  ont une frontière commune (les sommets du graphe sont les capitales des pays et les arêtes les points frontière, en supposant qu'il n'y en ait qu'un entre deux pays donnés).

Le graphe  $(X, \Gamma)$  est un graphe non orienté. De plus, il possède la propriété d'être un graphe **planaire** (§ 8, définition 50), ce que veut dire que deux arêtes quelconques du graphe ne se coupent pas.

On démontre assez facilement que le nombre chromatique  $\gamma$  d'un graphe planaire est inférieur ou égal à 5. Il en résulte que cinq couleurs au plus sont suffisantes pour colorier la carte, sans que deux pays voisins soient coloriés avec la même couleur.

De façon générale, le nombre chromatique peut être déterminé par la résolution d'un programme linéaire en nombres entiers. Un autre procédé empirique (dû à Kempe, voir [26]) consiste, à partir d'une coloration arbitraire, à chercher à éliminer petit à petit une des couleurs.

Des études plus récentes, menées avec l'aide de gros ordinateurs, semblent montrer que le nombre chromatique d'un graphe planaire est inférieur ou égal à 4, ce que les géographes constatent empiriquement.

### 3.7.3 Application aux problèmes d'affectation

Le nombre chromatique d'un graphe permet de modéliser des problèmes d'optimisation de ressources. Considérons, par exemple, un établissement d'enseignement qui propose des cours dans  $n$  matières. Les stagiaires peuvent s'inscrire à plusieurs cours simultanément, selon des modalités fixées à l'avance. On a donc un graphe non orienté à  $n$  sommets dans lequel les arêtes représentent les simultanités autorisées (il s'agit d'un graphe complet si tous les choix sont possibles). A la fin de la session, des examens sont organisés dans les diverses matières, à raison d'une heure par matière. On souhaite réduire la durée totale de l'examen grâce à des simultanités. Un coloriage du graphe donne la solution : chaque couleur représente une tranche horaire. Le nombre chromatique est la durée totale de l'examen.

On peut aussi citer l'affectation de fréquences à un certain nombre de radios : les stations de radio sont les sommets du graphe et les arêtes créées pour les stations dont les zones d'émission se recoupent. Ici, le nombre chromatique est le nombre de fréquences nécessaires.

## 3.8 Indice chromatique

La notion « duale » de la précédente intéresse la façon de colorier les arêtes.

**Définition 35.** Soit un graphe  $G$  non orienté. On appelle **indice chromatique**  $\psi(G)$  le plus petit nombre entier  $q$  tel que l'on puisse colorier chaque arête du graphe avec  $q$  couleurs distinctes sans que deux arêtes adjacentes soient de la même couleur (c'est donc impossible avec  $q - 1$  couleurs).

### Proposition 7.

L'indice chromatique d'un graphe est au moins égal au degré maximal du graphe.

**Preuve.**  $\diamond$  En chaque sommet, le degré doit être inférieur ou égal à l'indice chromatique.  $\diamond$

En fait, cette inégalité est assez bonne, comme on va le voir dans les deux théorèmes qui suivent.

**Théorème 13.** L'indice chromatique du graphe complet  $K_n$  vaut  $n - 1$  si  $n$  est pair,  $n$  si  $n$  est impair.

**Preuve.**  $\diamond$  Supposons d'abord qu'un graphe complet à  $n$  sommets soit coloriable avec  $n - 1$  couleurs. Considérons une couleur particulière qui doit figurer sur chaque sommet à cause du degré. Les arêtes de cette couleur permettent d'associer les sommets deux par deux, donc le nombre de sommets doit être pair (on dit que cela donne à  $G$  une structure de graphe biparti, § 7.1).

Dans ce cas, on peut représenter le graphe par un polygone régulier à  $n - 1$  sommets, le dernier sommet étant placé au centre. On choisit un sommet et on colorie, dans une même couleur, le rayon allant du centre à ce sommet, ainsi que toutes les arêtes perpendiculaires à ce rayon.

Puis on tourne la figure d'un  $(n - 1)$ -ième de tour, et on recommence avec une autre couleur (figure 11). Il ne peut y avoir aucun conflit à cause de la rotation. On recommence tant que nécessaire. Cela donne bien un coloriage à  $n - 1$  couleurs, et on ne peut faire mieux à cause du degré.

Dans le cas où  $n$  est impair, on peut plonger le graphe dans un graphe complet à  $n + 1$  sommets, qui est coloriable avec  $n$  couleurs d'après ce qui précède, et certainement pas moins par imparité comme on l'a noté au début de la preuve.  $\diamond$

**Théorème 14.** L'indice chromatique d'un graphe dépasse son degré maximal d'au plus une unité.

Ce théorème se démontre en construisant un algorithme qui est d'ailleurs rapide. La preuve étant assez longue, nous renvoyons le lecteur intéressé aux livres de C. Berge (cf. [Doc. AF 205]).

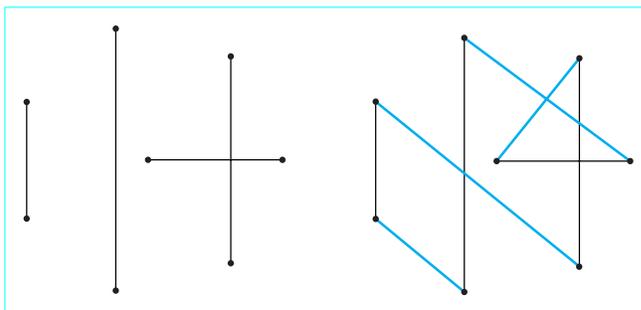


Figure 11 – Coloriage d'un graphe complet

### 3.9 Problème de l'isomorphisme et problèmes NP-difficiles

#### 3.9.1 Graphes isomorphes

Deux graphes sont **isomorphes** s'il existe une bijection des sommets de l'un sur les sommets de l'autre qui respecte les arêtes (si  $(x, y)$  est une arête de  $G$  alors  $(f(x), f(y))$  est une arête de  $G'$  et inversement). Il n'est pas toujours facile de décider que deux graphes sont isomorphes. Il faut au moins qu'ils aient même nombre de sommets, même nombre d'arêtes, même degré maximal, mêmes nombres chromatiques et de stabilité ; mais cela ne suffit pas, de loin, à assurer l'isomorphisme.

Le problème de l'isomorphisme, à savoir décider que deux graphes sont isomorphes, est actuellement considéré comme (algorithmiquement) plutôt difficile. On ne sait pas s'il s'agit d'un problème pour lequel un algorithme de complexité polynomiale peut ou ne peut pas exister.

#### 3.9.2 Problèmes P, NP et NP-difficiles

Donnons à présent quelques notions sommaires sur la complexité des algorithmes. Un algorithme peut être considéré comme une « boîte noire » acceptant, en entrée, un graphe ayant  $n$  sommets et  $m$  arêtes et donnant, en sortie, soit un nombre caractéristique de ce graphe, soit un ou quelques sommets ayant une propriété demandée. Nous avons donné, dans les paragraphes 1.4.3 et 3.7, des algorithmes pour découvrir les composantes connexes et la bicoloration d'un graphe (méthode de la « tache d'huile »).

De manière interne, l'algorithme va faire des opérations arithmétiques, principalement des tests de comparaison sur la représentation du graphe choisie : matrice ou listes d'adjacence. Le nombre d'étapes successives de l'algorithme est une fonction de  $n$  et  $m$  ; on l'appellera **complexité de l'algorithme**. On dira que la complexité est en  $\mathcal{O}(m^p)$  s'il existe une constante  $A > 0$  telle que la complexité soit inférieure à  $Am^p$  (quel que soit le graphe pris comme entrée) ; de même, on peut envisager une complexité en  $\mathcal{O}(n \cdot m^p)$  ou encore en  $\mathcal{O}(n \ln(m))$ ...

**Exemple** : la recherche des composantes connexes est pourvue d'une complexité en  $\mathcal{O}(m)$ , puisque dans le pire des cas toutes les arêtes sont explorées.

On dit qu'un algorithme est « polynomial » si sa complexité peut être majorée par une fonction polynomiale de la taille des données, c'est-à-dire par une borne du genre  $B(n + m)^p$  ; comme  $m \leq n^2$ , cela revient à une majoration du type  $C.n^q$ .

Un problème d'existence ou d'optimisation se trouve dans la classe P s'il bénéficie d'un algorithme de complexité polynomiale menant à une solution.

Un tel problème se trouve dans la classe NP (non polynomial) s'il existe un **algorithme de vérification** de complexité polynomiale capable de prouver qu'une solution (fournie a priori) convient ou non. Cette classe d'algorithmes contient la classe P. On ne sait pas si  $P \neq NP$  ; c'est même une des conjectures jugées les plus difficiles à l'heure actuelle.

Un problème est « NP-complet » si c'est un problème auquel tout autre problème de la classe NP se ramène par une transformation de nature polynomiale sur la taille des données.

De tels problèmes apparaissent régulièrement. Citons, entre autres, le problème de voyageur de commerce, celui du stable maximal, le calcul du nombre chromatique. Tous ces problèmes correspondent à une difficulté pratique à trouver un algorithme efficace pour de très grands graphes (les seuls algorithmes déterministes connus étant de complexité non polynomiale, souvent exponentielle).

Cette difficulté n'empêche pas de mettre au point des méthodes dites **heuristiques**, c'est-à-dire qui n'assurent pas à coup sûr l'obtention d'une solution du problème, mais y parviennent assez souvent ou assez près. Après avoir utilisé une méthode heuristique, on peut finir la recherche avec une méthode **exhaustive**, c'est-à-dire d'examen a priori de toutes les configurations plausibles.

## 4. Chemins et circuits

### 4.1 Réseaux

**Définition 36.** Un graphe fortement connexe, sans boucle et ayant plus d'un sommet, est appelé un **réseau**.

Il en résulte que dans un réseau, pour tout sommet  $x$ , il existe au moins deux arcs incidents à  $x$ , l'un vers l'intérieur et l'autre vers l'extérieur.

**Définition 37.**

On appelle **nœud** d'un réseau un sommet qui a plus de deux arcs incidents. Les autres sommets sont appelés **antinœuds**.

On appelle **branche** tout chemin pour lequel seuls les premier et dernier sommets sont des nœuds.

**Exemple** : sur la figure 12, les sommets **A, B, D, E, F** sont des nœuds.

Un sommet qui a deux arcs incidents (un vers l'intérieur, l'autre vers l'extérieur) est un antinœud (sommets **C, G**).

On a les branches suivantes : **(E, F)**, **(B, F)**, **(F, B)**, **(B, C, D)** et **(E, G, E)**, etc.

Dans un réseau, un chemin de longueur minimale entre un sommet  $x$  et un sommet  $y$  est appelé **piste** de  $x$  à  $y$ . Une piste est un chemin élémentaire, et un réseau possède toujours au moins une piste.

Sur la figure 12, le diamètre est égal à 4, puisque les pistes unissant **A** et **G** sont toutes de cette longueur, comme **(A, F, D, E, G)**.

**Théorème 15.** Soit  $n$  le nombre de sommets du réseau,  $m$  le nombre d'arcs,  $\delta$  la valeur du diamètre. On a alors les inégalités :

$$\delta + 1 \leq n \leq m \leq n(n - 1)$$

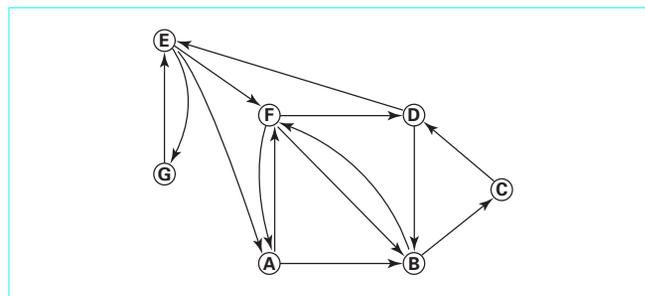


Figure 12 - Réseau

**Preuve.**  $\diamond$  A tout sommet, on peut associer un arc incident en ce sommet. Cela définit une injection de l'ensemble  $X$  dans l'ensemble  $A$  des arêtes, donc on a bien :

$$n \leq m.$$

L'ensemble  $A$  n'a pas plus d'éléments que l'ensemble des couples  $(x, y)$  de sommets distincts, ce qui amène :

$$m \leq n^2 - n.$$

Enfin, un chemin simple est de longueur  $n - 1$  au plus, ce qui concerne notamment un chemin de longueur égale au diamètre  $\delta$  du graphe.  $\diamond$

On peut appeler **densité** d'un réseau le rapport  $\frac{\ln(m)}{\ln(n)} - 1$ . Ce rapport est compris entre 0 et 1 d'après l'encadrement précédent. Un cycle est de densité 0, tandis qu'un réseau complet est de densité :

$$\begin{aligned} \frac{\ln(n) + \ln(n-1)}{\ln(n)} - 1 &= \frac{\ln(n-1)}{\ln(n)} = \frac{\ln(n) + \ln\left(1 - \frac{1}{n}\right)}{\ln(n)} \\ &= 1 - \frac{1}{n \ln(n)} + o\left(\frac{1}{n \ln(n)}\right) \end{aligned}$$

ce qui est proche de 1 quand  $n$  est grand. Certains auteurs nomment densité le rapport  $\frac{m}{n^2}$  (compris entre  $\frac{1}{n}$  et  $1 - \frac{1}{n}$ ).

## 4.2 Chemins et circuits eulériens

### 4.2.1 Problème d'Euler

Il s'agit d'une généralisation du jeu bien connu qui consiste à dessiner toutes les arêtes d'un graphe avec un crayon sans jamais le soulever, ni passer deux fois sur la même arête.

#### Définition 38.

Une **chaîne** dans un multigraphe non orienté est dite **eulérienne** si elle passe exactement une fois par chaque arête.

Un graphe est dit eulérien s'il admet un tel parcours fermé, aussi appelé **cycle eulérien**.

Voici le théorème fondateur de la théorie topologique des graphes.

**Théorème 16.** Un multigraphe non orienté possède un circuit eulérien si, et seulement si, il est connexe et que tous ses sommets ont un degré pair. Il admet une chaîne eulérienne, si, et seulement si, il est connexe et qu'il a 0 ou 2 sommets de degré impair.

**Preuve.**  $\diamond$  Si le circuit eulérien existe, le graphe est connexe ; on peut orienter le circuit d'une manière arbitraire. En chaque sommet, le nombre d'arcs incidents doit être égal au nombre d'arcs sortants par le circuit. Dans le cas d'une chaîne eulérienne, les deux extrémités font exception (on part ou on arrive une fois de plus, d'où un degré impair).

Réciproquement, supposons que tous les sommets soient de degrés pairs. Partons d'un sommet  $x_0 = x$  quelconque et suivons une arête vers un successeur  $x_1$  de  $x$  et ainsi de suite tant que le chemin ainsi construit  $(x_0, x_1, \dots, x_p)$  reste simple.

Si  $x_p = x_q$  avec  $1 \leq q < p$ , cela signifie que toutes les arêtes incidentes sur  $x_q$  ont été visitées ; il y a une « arrivée » de plus que de « départs », donc un degré impair en  $x_q$ , ce qui est exclu. On a donc un cycle :

$$x_0 = x_p.$$

Si le cycle est eulérien, c'est fini.

Sinon, on considère le graphe partiel  $G'$  engendré par les arêtes non visitées obtenu par suppression des arêtes du chemin (concrètement, cette suppression se fait au fur et à mesure de l'exploration). Les degrés des sommets restent pairs dans  $G'$  par la nature même de la suppression opérée. Comme  $G$  est connexe,  $G'$  a au moins un sommet en un  $x_k$ . On construit dans  $G'$  un cycle simple commençant en  $x_k$  de la même manière que précédemment. On « greffe » alors ce cycle sur le premier en  $x_k$ , ce qui fournit un cycle simple plus long que le premier. Il n'y a plus qu'à recommencer tant qu'il y a des arêtes non visitées.

Si on a deux sommets  $s$  et  $p$  de degrés impairs, on se ramène au cas précédent en ajoutant un sommet  $z$ , une arête  $sz$  et une arête  $pz$ .  $\diamond$

**Exemple d'Euler** : il est fourni par la ville de Königsberg (aujourd'hui Kaliningrad) ; cette ville est traversée par la rivière Pregel qui coule de part et d'autre de l'île de Kneiphof. Sept ponts traversent la rivière et ses bras. On se posa la question de savoir si un promeneur peut traverser une et une seule fois chaque pont et revenir à son point de départ.

Le 2-graphe de la figure 13 symbolise la configuration des ponts de cette ville ; le théorème d'Euler montre qu'il est impossible de trouver une chaîne eulérienne, car ce graphe possède 4 sommets de degré impair.

La preuve du théorème d'Euler fournit un algorithme de recherche de circuit eulérien, et à l'occasion un test de connexité du graphe ; mais cet algorithme n'est pas très pratique à cause de la « greffe ».

Un autre **algorithme**, dû à **Fleury**, permet de tracer des cycles eulériens. On observe les règles suivantes :

a) on part d'un sommet quelconque  $x$  ; chaque fois qu'on parcourt une arête, on l'efface ;

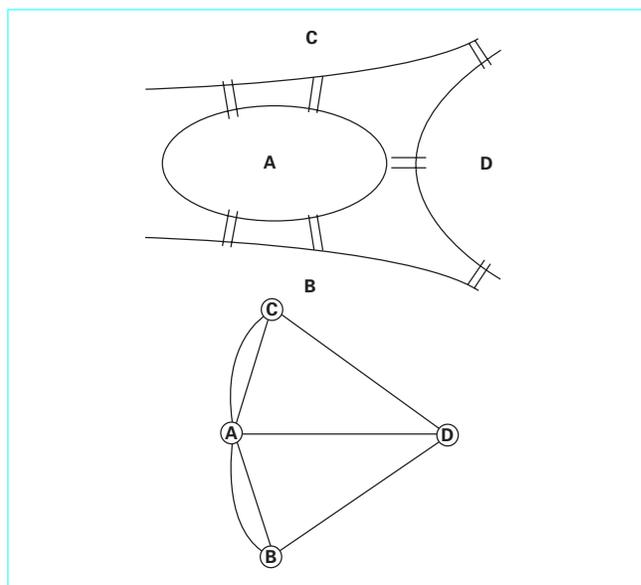


Figure 13 – Ponts de Königsberg

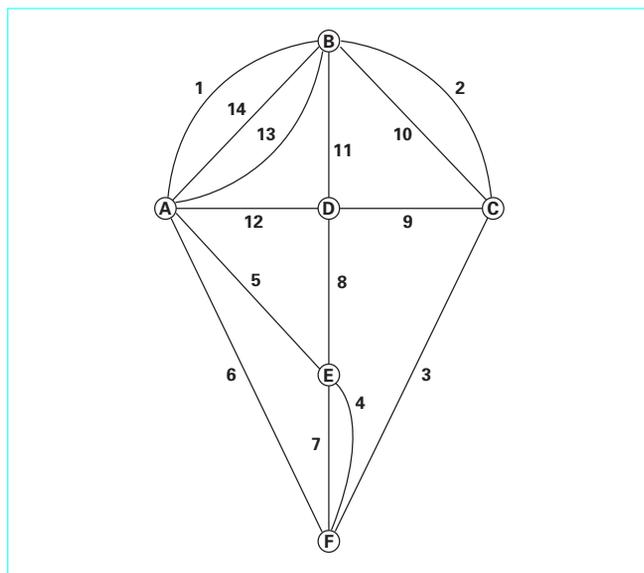


Figure 14 – Recherche d'un cycle eulérien selon Fleury

b) il ne faut jamais utiliser une arête  $\bar{u}$  qui, au moment du tracé, soit un isthme (§ 3.5.1).

**Exemple :** la figure 14 donne un graphe dont tous les sommets sont pairs. Les numéros indiquent l'ordre de parcours.

### 4.2.2 Problème d'Euler orienté

Le problème d'Euler peut aussi être considéré avec des « sens uniques ».

**Définition 39.**

Un **chemin** dans un multigraphe orienté est dit **eulérien** s'il passe exactement une fois par chaque arête.

Un graphe est dit eulérien s'il admet un **circuit eulérien**.

On adapte aisément les idées du paragraphe précédent. Nous donnons les résultats sans démonstration.

**Théorème 17.** Un graphe admet un circuit eulérien si, et seulement si, il est connexe et que pour tout sommet le degré entrant est égal au degré sortant.

Un graphe admet un chemin eulérien mais pas de circuit eulérien si, et seulement si, il est connexe et que, pour tout sommet sauf deux (**S** et **P**), le degré entrant est égal au degré sortant, ainsi que :

$$d_e(\mathbf{S}) = d_s(\mathbf{S}) - 1 ,$$

$$d_e(\mathbf{P}) = d_s(\mathbf{P}) + 1 .$$

### 4.2.3 Problème du postier chinois

Le problème dit du postier chinois est de parcourir les rues d'une ville en passant **au moins** une fois dans chaque rue, le graphe n'étant pas nécessairement eulérien ; on cherche bien sûr à minimiser le parcours total.

On rencontre ce genre de problème dans les organisations de tournées de distribution de courrier, prospectus, ramassage d'ordures, inspection de réseaux de distribution, etc.

Il s'agit d'abord de savoir si le « parcours chinois » a au moins une solution.

**Théorème 18.** Un graphe non orienté admet un cycle chinois si, et seulement si, il est connexe.

Un graphe orienté admet un circuit chinois si, et seulement si, il est fortement connexe.

**Preuve.**  $\diamond$  La première assertion est presque évidente (il suffit de se promener en aller-retour entre un sommet particulier et tous les autres).

En ce qui concerne la seconde, la condition est visiblement nécessaire. Pour montrer qu'elle est suffisante, on ajoute au graphe des arcs, le transformant en multigraphe.

Si  $x$  est un sommet tel que  $d_e(x) - d_s(x) > 0$ , il existe un sommet  $y$  tel que  $d_e(y) - d_s(y) < 0$ . Appelons **excès** d'un sommet  $z$  la différence :

$$E(z) = d_e(z) - d_s(z).$$

Comme le graphe est fortement connexe, il existe un chemin de  $x$  à  $y$  et, de plus, la somme des excès est nulle. On ajoute, le long des arcs de ce chemin,  $q$  arcs de même sens, avec :

$$q = \text{Min}(E(x), -E(y)).$$

Les sommets visités gardent le même excès, tandis que l'une des deux extrémités récupère un excès nul, l'autre s'étant rapprochée de 0. On recommence autant de fois que nécessaire jusqu'à obtenir un multigraphe qui va posséder un circuit eulérien. Il n'y a plus qu'à interpréter le passage le long des arêtes de mêmes extrémités comme plusieurs passages successifs le long de l'arête initiale.  $\diamond$

La minimisation de la longueur du trajet ou du coût par rapport à une distribution de coûts est une affaire plus délicate qui relève plutôt de l'optimisation des flots (paragraphe 6.1).

## 4.3 Chemins et circuits hamiltoniens

### 4.3.1 Généralités

**Définition 40.** Dans un graphe  $(X, \Gamma)$ , on dit qu'un chemin  $\mu = [x_1, x_2, \dots, x_n]$  est **hamiltonien** s'il passe une fois, et une seule, par chaque sommet du graphe. La définition est identique pour un circuit mais, dans ce cas,  $x_0$  est confondu avec  $x_n$ .

Le graphe lui-même est dit hamiltonien s'il possède un circuit hamiltonien comme sous-graphe.

**Théorème 19 (de König).** Si le graphe est total (c'est-à-dire si tout couple de sommets est joint dans au moins une direction), il existe un chemin hamiltonien.

Il est facile, dans ce cas, de trouver le chemin de proche en proche.

Dans le cas général, il existe diverses méthodes.

Lorsque l'ordre du graphe est élevé et qu'il n'est pas fortement connexe, on a intérêt à le décomposer en sous-graphes fortement connexes pour rechercher les chemins hamiltoniens ; l'algorithme de Foulkes, décrit au paragraphe 2.2.3, convient à cet usage.

Pour rechercher les chemins hamiltoniens existant éventuellement, on opère donc comme suit. Si  $x_i$  et  $x_j$  sont deux sommets du graphe appartenant à deux composantes fortement connexes différentes et s'il existe un chemin de  $x_i$  vers  $x_j$ , alors il n'existe pas de chemin de  $x_j$  vers  $x_i$  par définition des composantes fortement connexes. Soit  $C$  la composante de  $x_i$  et  $C'$  celle de  $x_j$ ; il en résulte qu'il existe un chemin allant de tout sommet de  $C$  à tout sommet de  $C'$ , mais qu'il n'existe aucun chemin allant d'un sommet de  $C'$  vers un sommet de  $C$ . On écrira :

$$C < C' \text{ si } C \neq C' ;$$

une relation d'ordre strict transitive et antisymétrique est donc définie.

L'existence d'un **ordre total** entre toutes les classes d'équivalence  $C_1, C_2, \dots, C_r$ , ainsi défini est une condition nécessaire mais non suffisante pour qu'il existe un ou plusieurs chemins hamiltoniens dont l'origine est en  $C_1$  et la terminaison en  $C_r$ .

Déterminer si un graphe est hamiltonien est en fait un problème NP-complet, pour lequel on ne doit pas s'attendre à trouver facilement une réponse.

On trouve des applications très diverses des chemins hamiltoniens. Citons le problème du cavalier (déplacer un cavalier sur un échiquier de sorte que le cavalier passe une fois, et une seule, sur chacune des cases), la recherche opérationnelle, les problèmes d'ordonnement, la construction de codes correcteurs d'erreurs, etc.

### 4.3.2 Méthode de Kaufmann

Kaufmann a proposé la méthode suivante qui semble relativement simple : elle repose sur l'emploi d'une multiplication matricielle particulière que l'auteur désigne sous le nom de **multiplication latine**. Nous allons exposer cette méthode sur un exemple (figure 15).

La **matrice latine**  $[M]^{(1)}$  est définie de la façon suivante : si le sommet  $i$  est relié au sommet  $j$ , on porte  $ij$  dans la case  $(i, j)$  ; s'il n'est pas relié, on porte 0. De plus, on porte 0 pour toute case de la diagonale. De cette matrice, on va déduire une autre matrice latine en enlevant la première lettre de chaque case ou lettre initiale :  $[M]^{(1)}$  devient  $[\tilde{M}]^{(1)}$ . Dans notre exemple :

$$[M]^{(1)} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 0 & AB & AC & 0 & AE \\ 0 & 0 & BC & 0 & BE \\ 0 & 0 & 0 & CD & 0 \\ 0 & 0 & DC & 0 & DE \\ EA & 0 & 0 & ED & 0 \end{pmatrix} \end{matrix}$$

$$[\tilde{M}]^{(1)} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 0 & B & C & 0 & E \\ 0 & 0 & C & 0 & E \\ 0 & 0 & 0 & D & 0 \\ 0 & 0 & C & 0 & E \\ A & 0 & 0 & D & 0 \end{pmatrix} \end{matrix}$$

Pour obtenir  $[M]^{(2)}$ , on va multiplier d'une certaine manière  $[M]^{(1)}$  par  $[\tilde{M}]^{(1)}$ . On va opérer, comme pour le calcul matriciel classique, « ligne par colonne ». On marquera dans la case  $(i, k)$  de  $[M]^{(2)}$ , c'est-à-dire dans la case produit, non pas un produit classique, ce qui n'aurait ici aucun sens, mais les signes suivants :

— 0 si, pour tous les  $j$ , l'une au moins des cases  $(i, j)$  de  $[M]^{(1)}$  et  $(j, k)$  de  $[\tilde{M}]^{(1)}$  contient 0 ;

— 0 si, pour chaque couple, formé d'une case  $(i, j)$  de  $[M]^{(1)}$  et d'une case  $(j, k)$  de  $[\tilde{M}]^{(1)}$ , examiné dans le produit ligne par

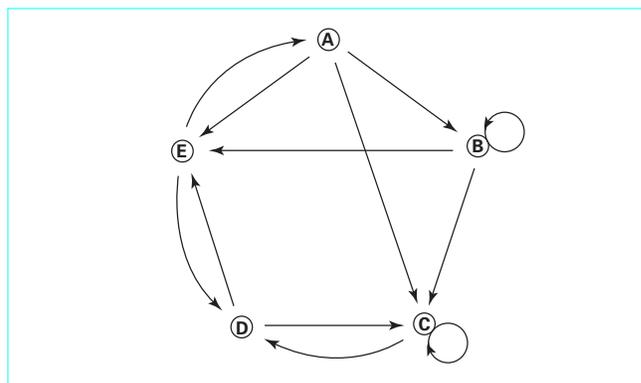


Figure 15 – Recherche de chemin hamiltonien selon Kaufmann

colonne, on ne peut pas former une séquence ne contenant pas de lettre répétée ;

— autrement, les séquences obtenues en mettant à la suite d'une séquence de la case  $(i, j)$  de  $[M]^{(1)}$  une séquence  $(j, k)$  de  $[\tilde{M}]^{(1)}$ , à condition que cela ne crée pas de répétition de lettres. Dans l'exemple proposé, les séquences  $[\tilde{M}]^{(1)}$  se réduisent à une lettre.

Cela donne dans l'exemple :

$$[M]^{(1)} \mathcal{L} [\tilde{M}]^{(1)} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 0 & 0 & ABC & AED & ABE \\ BEA & 0 & 0 & BED & 0 \\ 0 & 0 & 0 & 0 & CDE \\ DEA & 0 & 0 & 0 & 0 \\ 0 & EAB & EDC & 0 & 0 \end{pmatrix} \end{matrix} = [M]^{(2)}$$

La matrice  $[M]^{(1)}$  donne la liste des chemins élémentaires de longueur 1,  $[M]^{(2)}$  celle des chemins élémentaires de longueur 2. Le même processus répété donne les matrices suivantes par multiplication de  $[\tilde{M}]^{(1)}$  :

$$[M]^{(2)} \mathcal{L} [\tilde{M}]^{(1)} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 0 & 0 & AEDC & ABED & ACDE \\ 0 & 0 & BEDC & 0 & BCDE \\ CDEA & 0 & 0 & 0 & 0 \\ 0 & DEAB & DEAC & 0 & 0 \\ 0 & 0 & EABC & EACD & 0 \end{pmatrix} = [M]^{(3)}$$

$$[M]^{(3)} \mathcal{L} [\tilde{M}]^{(1)} = \begin{matrix} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 0 & 0 & ABEDC & 0 & ABCDE \\ BCDEA & 0 & 0 & BEACD & 0 \\ 0 & CDEAB & 0 & 0 & 0 \\ 0 & 0 & DEABC & 0 & 0 \\ 0 & 0 & 0 & EABCD & 0 \end{pmatrix} = [M]^{(4)}$$

$[M]^{(4)}$  est la dernière itération donnant tous les chemins hamiltoniens ; il y en a 7, les sept termes non nuls de la matrice.

### 4.3.3 Recherche d'un circuit hamiltonien

Nous donnons à présent quelques idées sur la recherche générale des circuits hamiltoniens.

Pour rechercher les circuits hamiltoniens, il suffit d'abord de déterminer les chemins hamiltoniens du graphe donné et de fermer les chemins obtenus, si cela est possible, par un arc appartenant au graphe.

Les notions de chemins et circuits hamiltoniens s'adaptent facilement aux notions de chaînes et de cycles hamiltoniens. En considérant un graphe non orienté comme un graphe symétrique, on se retrouve placé dans le cas de chemins et circuits.

**Lemme 1 (de Bondy-Chvátal).** Soit  $u$  et  $v$  deux sommets non adjacents dans un graphe non orienté  $G$  ayant  $n$  sommets, tels que

$$d(u) + d(v) \geq n.$$

Soit  $G'$  le graphe obtenu en ajoutant une arête entre  $u$  et  $v$ . Si  $G'$  est hamiltonien, alors  $G$  l'est aussi.

**Preuve.**  $\diamond$  Il existe au moins un chemin hamiltonien

$$C = (u = w_1, w_2, \dots, w_n = v)$$

dans  $G$  de  $u$  à  $v$  puisque  $G'$  est hamiltonien.

Soit  $S$  l'ensemble des sommets adjacents à  $u$  dans  $G$ . Soit  $T$  l'ensemble des successeurs (selon  $C$ ) des sommets adjacents à  $v$  dans  $G$ . On a

$$|S| + |T| = d(u) + d(v) \geq n,$$

tandis que  $u$  ne figure ni dans  $S$  ni dans  $T$ ; donc  $S$  et  $T$  ont au moins un sommet commun  $w_{i+1}$ , de sorte que  $w_{i+1}$  est un successeur de  $u$  et que  $w_i$  est un successeur de  $v$ . On trouve alors le cycle hamiltonien :

$$u = w_1, \dots, w_i, v = w_n, w_{n-1}, \dots, w_{i+1}, u. \quad \diamond$$

**Théorème 20 (de Dirac).** Soit un graphe non orienté, connexe et sans boucle, ayant  $n$  sommets. Si le degré de chaque sommet est au moins égal à  $n/2$ , alors il existe au moins un circuit hamiltonien dans  $G$ .

**Preuve.**  $\diamond$  On applique de façon répétée le lemme précédent, ce qui est possible par l'hypothèse sur le degré. On finit par obtenir un graphe complet qui est évidemment hamiltonien, donc le graphe initial l'est aussi.  $\diamond$

## 4.4 Recherche du plus court chemin

Étant donné un graphe  $(X, \Gamma)$  et deux sommets  $S, P$ , le problème est de trouver le plus court chemin allant de  $S$  (la « source ») à  $P$  (le « puits »).

Deux cas peuvent se présenter, étudiés dans les deux paragraphes suivants.

### 4.4.1 Cas où le minimum recherché correspond à un nombre minimum d'arcs

Pour rechercher le ou les chemins existants, on peut appliquer un algorithme de recherche en largeur (§ 5.2). On numérote les

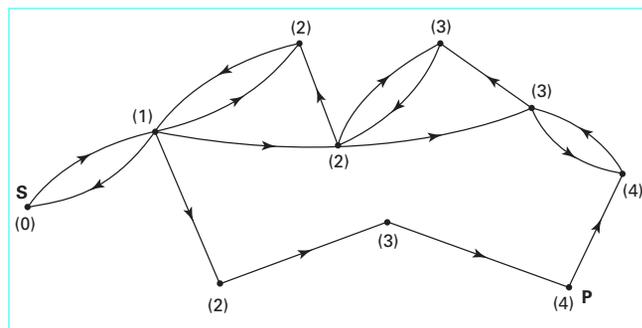


Figure 16 – Calcul progressif des écarts

sommets adjacents, en partant de l'origine, jusqu'à atteindre le sommet final. Les sommets déjà numérotés ne doivent pas être modifiés. Les numéros ainsi déposés sont égaux aux écarts des divers sommets vers  $S$ .

Sur la figure 16, pour aller de  $S$  en  $P$  :

$$d(S, P) = 4 \text{ et } d(P, S) = 7.$$

### 4.4.2 Cas où le minimum recherché correspond à une longueur minimale

Nous considérons désormais un graphe « pondéré » par une valeur numérique  $\ell(u) \geq 0$  attachée à chaque arc  $u$ . Ce nombre peut, selon les problèmes, s'appeler longueur, coût, distance, durée, etc.

**Exemple :** une carte routière avec les distances entre villes est un graphe pondéré (non nécessairement planaire : une autoroute coupe une route normale sans point commun).

Étant donné un chemin  $\mu$ , on appelle désormais longueur ou coût  $\ell(\mu)$  de ce chemin la somme des longueurs des arcs parcourus :

$$\ell(\mu) = \sum_{u \in \mu} \ell(u).$$

On se propose de trouver un chemin  $\mu$  allant d'un sommet  $S$  (la source) à un autre sommet  $P$  (le puits) et de longueur minimale (ou coût minimal).

Il existe plusieurs algorithmes pour résoudre ce problème ; trois d'entre eux sont particulièrement utilisés : ce sont les algorithmes de Dijkstra, Bellman-Kalaba et Floyd.

Les sommets du graphe étant notés  $(x_1, \dots, x_n)$ , la source sera notée  $x_1$  et le puits  $x_n$ . Nous posons :

$$\left. \begin{aligned} C_{i,j} &= \ell(x_i x_j) && \text{si } x_i \text{ et } x_j \text{ sont adjacents ;} \\ C_{i,j} &= +\infty && \text{si } x_i \text{ et } x_j \text{ ne sont pas adjacents} \\ \text{et } C_{i,i} &= 0. \end{aligned} \right\}$$

On cherche donc un chemin

$$\mu = [x_1, x_{i_1}, x_{i_2}, \dots, x_{i_k}, x_n]$$

tel que l'expression  $C_{1, i_1} + C_{i_1, i_2} + \dots + C_{i_k, n}$  soit minimale (et finie).

### 4.4.3 Algorithme de Dijkstra

Nous considérons ici le problème du chemin optimal quand les coûts sont positifs. On peut, dans ce cas, parler de longueurs. Une

stratégie simple pour trouver un chemin optimal est la méthode dite « gloutonne », dans laquelle on s'efforce, à chaque décision, de réaliser un optimum local (on a choisi le terme « glouton », parce qu'on tente d'obtenir la plus grande satisfaction immédiate).

**Description de l'algorithme**

● **Variables :**

- un tableau de valeurs  $V$  qui contient *in fine* les distances des divers sommets du graphe à la source ;  $V_j$  contient, à chaque étape, la plus courte distance découverte entre  $S$  et  $x_j$  ;
- un ensemble  $M$ , initialement vide, qui donne, à tout instant, l'ensemble des numéros des sommets analysés ;
- un tableau  $P$  de numéros de sommets précédents, qui permet de tracer le chemin optimal (si on ne se contente pas de la distance minimale) ;  $P_j$  donne le numéro du sommet qui précède  $x_j$  dans le chemin optimal entre  $S$  et  $x_j$ .

● **Initialisation :**

$$\begin{aligned} V_1 &= 0 \\ V_j &= C_{1j} \text{ si } j > 1, \\ M &= \{1\}, \\ P_j &= 1 \text{ si } j > 1. \end{aligned}$$

● **Itération :** tant que  $n$  n'est pas dans  $M$  :

- choisir un sommet  $x_j$  tel que  $j$  ne soit pas dans  $M$  et que  $V_j$  soit le plus petit possible ;
- ajouter  $j$  à  $M$  ;
- pour tout sommet  $x_i$  tel que  $i$  ne soit pas dans  $M$ , si  $V_j + C_{ji} < V_i$  remplacer  $V_i$  par  $V_j + C_{ji}$  et poser  $P_i = j$ .

● **Achèvement :**  $V_n$  donne la longueur (ou coût) du chemin minimal entre  $S = x_1$  et  $P = x_n$ .

**Exemple de mise en œuvre**

Voici un exemple simple inspiré de [1]. Dans la figure 17, la source est numérotée **A** et le puits **E**. Les distances (positives) sont indiquées près des arcs.

Étape	$M$	$j$	$V_2$	$V_3$	$V_4$	$V_5$
0	{A}		1	$\infty$	3	10
1	{A, B}	2	1	6	3	10
2	{A, B, D}	3	1	5	3	9
3	{A, B, D, C}	4	1	5	3	6
4	{A, B, D, C, E}	5	1	5	3	6

**Justification de l'algorithme**

Il est clair que l'algorithme se termine, puisque, à chaque étape, au moins un sommet  $x_j$  sera trouvé et  $j$  ajouté à  $M$ . Le problème est de savoir si les valeurs  $V_j$  trouvées à la fin sont bien minimales.

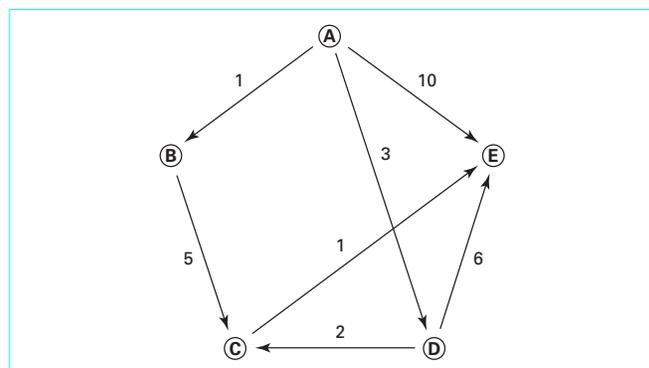


Figure 17 – Graphe pondéré pour Dijkstra

Il sera commode de noter  $S$  l'ensemble des sommets dont les numéros sont dans  $M$ . Par « chemin minimal depuis  $x_1$  tracé dans  $S$  », nous entendons un chemin dont tous les sommets sont dans  $S$  sauf le dernier.

On montre, par récurrence, qu'à chaque étape :

(a) la valeur  $V_j$  d'un sommet  $x_j \in S$  est bien la distance minimale de  $x_j$  à la source  $x_1$  ;

(b) la valeur  $V_k$  d'un sommet  $x_k \notin S$  est la longueur minimale d'un chemin à la source à  $x_j$  entièrement tracé dans  $S$ .

● A l'initialisation, c'est bien le cas, puisque  $S = \{x_1\}$  et que les  $V_k$  sont infinis s'il n'y a pas d'arc entre  $x_1$  et  $x_k$ , égaux à la distance entre  $x_1$  et  $x_k$  sinon.

● Quand on itère et qu'on ajoute  $x_j$  à  $S$ , on se demande si (a) est vérifié. Si ce n'était pas le cas, il existerait un chemin  $\mu'$  entre  $x_1$  et  $x_j$  visitant au moins un sommet  $y$  hors de  $S$  et de longueur inférieure au chemin  $\mu$  tracé dans  $S$ . On peut supposer que tous les sommets visités entre  $x_1$  et  $y$  sont dans  $S$ . Soit  $\mu'_0$  le sous-chemin de  $\mu'$  compris entre  $x_1$  et  $y$ . Comme les longueurs des arcs sont supposées positives, on a :

$$\ell(\mu'_0) \leq \ell(\mu') < \ell(\mu) = V_j.$$

Cela contredit le choix de  $x_j$ , puisque  $y$  fait « mieux que lui ».

● En ce qui concerne (b), considérons un sommet  $z = x_i$  hors de  $S$ , alors qu'on vient d'ajouter  $x_j$  à  $S$ . Si le chemin minimal tracé dans  $S$  de  $x_1$  à  $z$  ne passe pas par  $x_j$ , alors  $V_i$  est bien sa longueur en appliquant l'hypothèse (b) de l'étape précédente. Si ce chemin passe par  $x_j$ , sa longueur est  $V_j + C_{ji}$ . Or, la nouvelle valeur de  $V_i$  est précisément  $\text{Min}(V_j + C_{ji}, V_i)$ , et c'est bien la longueur attendue.

Terminons en signalant que la complexité de l'algorithme de Dijkstra dépend de son implémentation ; elle est en général du type  $\mathcal{O}(n^2)$ . S'il est mis en œuvre sur un graphe peu dense, avec une structure de tas pour représenter l'arborescence en cours de construction, sa complexité devient un  $\mathcal{O}(m \ln(n))$ .

**4.4.4 Algorithme de Bellman-Kalaba**

**Principe**

C'est une variante de l'algorithme précédent qui utilise le principe d'optimalité et fonctionne même si les coûts des arcs sont négatifs.

L'équation de référence de Bellman s'écrit :

$$\begin{cases} V_i = \text{Min}_{j \neq i} (V_j + C_{j,i}) \text{ avec } i = 2, \dots, n \\ V_1 = 0 \end{cases}$$

où  $V_j$  ( $i = 1, \dots, n - 1$ ) représente la valeur du chemin de longueur minimale qui joint le sommet  $x_i$  au sommet initial  $x_1$ .

**Description de l'algorithme**

● **Variable :** un tableau de valeurs  $V$  (dans la suite, on considère une suite de tels tableaux en indexant par le numéro de l'étape, mais, d'un point de vue informatique, un seul tableau suffit) ;  $V_j^{(k)}$  contient, à l'issue de la  $k$ -ième étape, la longueur du chemin à  $k$  arcs le plus avantageux entre  $x_1$  et  $x_j$ .

● **Initialisation :**

$$\begin{cases} V_i^{(0)} = C_{1,i} \text{ pour } i = 2, \dots, n \\ V_1^{(0)} = 0 \end{cases}$$

● **Itération :**

— calculer :

$$\begin{cases} V_i^{(1)} = \text{Min}_{j \neq i} (V_j^{(0)} + C_{j,i}) \text{ avec } i = 2, \dots, n \text{ et } j = 1, \dots, n \\ V_1^{(1)} = 0 \end{cases}$$

— et, plus généralement :

$$\begin{cases} V_i^{(k)} = \text{Min}_{j \neq i} (V_j^{(k-1)} + C_{j,i}) \\ V_1^{(k)} = 0 \end{cases}$$

● **Achèvement** : on s'arrête lorsque

$$V_i^{(k)} = V_i^{(k-1)},$$

ce qui se produit au plus tard pour  $k = n$  ;  $V_n^{(n)}$  donne la longueur (ou coût) du chemin minimal entre  $s = x_1$  et  $p = x_n$ .

■ **Autres considérations**

Il faut d'abord vérifier que l'algorithme se termine. Dans le cas où les coûts des arcs peuvent être négatifs, il y a une situation gênante, celle où un circuit de coût total négatif existe. En parcourant ce circuit de manière répétée, on peut avoir un coût aussi petit qu'on le veut, donc le problème de minimisation posé n'a pas de solution.

Autrement, la formule d'évolution  $V_i^{(k)}$  ne peut produire d'effet que si le coût de l'arc de  $x_j$  vers  $x_i$  n'est pas  $+\infty$ , c'est-à-dire si  $x_j$  est un successeur de  $x_i$  ; en ce cas, elle exprime que tout chemin optimal entre  $x_1$  et  $x_i$  doit inclure un sous-chemin optimal entre  $x_1$  et  $x_j$ . Il ne reste plus qu'à balayer l'ensemble des prédécesseurs de  $i$  à la recherche du meilleur coût !

Le cas d'exception (circuit de coût total négatif) apparaîtra dans l'algorithme au plus tard avec  $k = n$  : si  $V^{(n)} \neq V^{(n-1)}$ , un tel circuit existe.

L'algorithme de Bellman-Kalaba peut être appliqué à la recherche d'un chemin à **valeur maximale**, mais avec une restriction évidente : le graphe ne doit contenir ni circuit, ni boucle ; dans ce cas, en effet, la longueur d'un chemin pourrait ne pas être finie. Il faut simplement remplacer l'abréviation  $\text{Min}_{j \neq i}$  par  $\text{Max}_{j \neq i}$  et on écrira que  $C_{ij} = -\infty$  si  $x_j$  n'est pas relié à  $x_j$  par un arc.

**4.4.5 L'algorithme de Floyd**

Cet algorithme permet de calculer les valeurs de toutes les distances minimales sans donner les chemins optimaux. L'idée est de faire évoluer la matrice  $C$  vers la matrice des distances optimales en tenant progressivement compte des divers chemins possibles.

- **Variable** : une matrice  $D$  indexée comme  $C$ , donc de taille  $n \times n$ .
- **Initialisation** :  $D = C$ .
- **Itération** :
  - pour  $k$  variant de 1 à  $n$ ,
  - pour  $i$  et  $j$  variant de 1 à  $n$ , si  $D_{ik} + D_{kj} < D_{ij}$  remplacer  $D_{ij}$  par  $D_{ik} + D_{kj}$  (cela n'arrive pas si  $i = k$  ou  $j = k$ ).
- **Achèvement** :  $D$  contient les distances mutuelles des divers sommets du graphe.

**4.5 Problème du voyageur de commerce (PVC)**

**4.5.1 Position du problème**

Ce problème pratique consiste à trouver l'itinéraire le moins coûteux pour un voyageur de commerce qui, partant d'une ville donnée, doit visiter chacune des villes d'une certaine région et ensuite retourner à son point de départ. Il s'agit donc de la recherche d'un circuit hamiltonien tel que la somme des coûts associés aux arêtes parcourues soit minimale. Il n'y a pas d'algorithme

déterministe efficace répondant à cette question (c'est un problème NP-complet), mais on peut en trouver des solutions approchées (méthode du recuit simulé, § 4.5.3).

On rencontre des variantes du PVC dans les problèmes d'organisation de déplacements d'un outil sur une série de sites prédéfinis. Ce dernier problème est dit **euclidien** parce que les distances sont faciles à calculer et vérifient les inégalités triangulaires.

On peut voir le PVC comme un problème d'optimisation et l'aborder avec des méthodes de programmation linéaire. En effet, notons  $H$  l'ensemble des circuits hamiltoniens dans le graphe  $G$  (en espérant que  $H$  n'est pas vide !), et pour un circuit hamiltonien

$$\mu = (x_{i_1}, x_{i_2}, \dots, x_{i_n}, x_{i_1}),$$

posons

$$\ell(\mu) = C_{i_1 i_2} + C_{i_2 i_3} + \dots + C_{i_n i_1}.$$

Dès lors, il s'agit de minimiser la fonction  $\ell(\mu)$  en faisant varier les  $i_1, \dots, i_n$  de sorte que  $x_{i_1}, x_{i_2}, \dots, x_{i_n}$  soient distincts. Rappelons que les non-arcs sont pris en compte par des coûts  $C_{ij}$  infinis ; en pratique, on considère donc que le graphe est complet mais que certains arcs peuvent être très coûteux.

**4.5.2 Heuristiques gloutonnes**

Les méthodes étudiées ici ne visent pas à fournir d'emblée le circuit optimal, mais à donner un circuit déjà avantageux avant d'affiner ce résultat par des méthodes plus sophistiquées.

■ Une première idée consiste à construire de proche en proche un circuit en choisissant à chaque étape le voisin libre le plus proche du dernier sommet occupé. Cette méthode est assez rapide, mais elle peut conduire à un circuit de coût très supérieur au coût minimal si le « retour » vers le sommet initial est de grande longueur.

**Exemple** : la figure 18 présente une telle situation : la méthode du plus proche voisin va choisir le chemin (A, B, C, D, A) de coût  $10 + 20 + 40 + 30 = 100$  alors que le circuit (A, C, D, B, A) a un coût de  $11 + 40 + 25 + 10 = 86$ .

■ Une autre idée consiste à avoir un petit circuit initial (une boucle de coût nul sur un sommet, faute de mieux) et de tenter d'y greffer un « détour » vers un sommet libre. Il faut choisir ce sommet libre ; cela peut se faire en choisissant le sommet libre le plus lointain possible ou, au contraire, le plus proche. Enfin il faut choisir l'arc à remplacer par le détour, en minimisant l'augmentation du coût. On peut aussi, mais c'est plus long, fusionner les deux démarches, cherchant à la fois le sommet libre et le lieu de son insertion pour avoir une augmentation minimale du coût.

**Exemple** : pour la figure 18, si on commence avec un cycle vide en A, D est le plus lointain ; le cycle (A, D, A) admet comme plus lointain C à cause de

$$d_{AC} = 11 > d_{AB} = 10.$$

On crée le cycle (A, C, D, A) de coût 81. Il reste trois manières de greffer B sur ce cycle, qui ont des coûts de 100, 86 et 86.

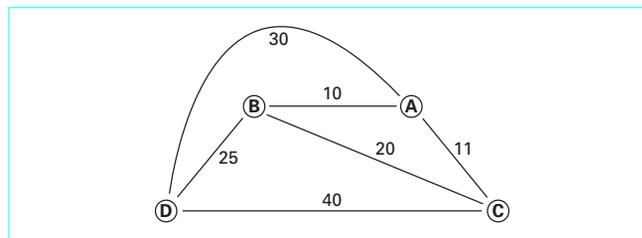


Figure 18 – Problème du voyageur de commerce

■ Une amélioration des résultats fournis par ces méthodes peut être tentée par des techniques « locales », c'est-à-dire qui consistent à déplacer quelques arêtes. La plus connue revient à sélectionner deux arêtes non adjacentes  $xy$  et  $uv$  du circuit hamiltonien déjà obtenu et à « croiser » les sommets, remplaçant ces deux arêtes par  $xu$  et  $yv$  ou  $xv$  et  $yu$ . Le croisement est appliqué s'il conduit à une amélioration de la longueur du circuit. On essaie tous les couples d'arêtes possibles et on retient celui qui produit la meilleure amélioration.

### 4.5.3 Méthode du recuit simulé

Malgré toutes les idées antérieures, il se peut que le graphe possède deux circuits hamiltoniens très différents, mais ayant des coûts voisins et presque minimaux. Dans ces conditions, les méthodes heuristiques et les améliorations locales risquent de ne donner qu'un optimum local et non l'optimum global. A partir d'un optimum local, pour aller vers l'optimum global il va falloir accepter une dégradation temporaire du coût.

Le nom de la méthode qui va être exposée provient d'une *analogie thermodynamique*. Quand on refroidit trop vite un métal, la cristallisation comporte de nombreux défauts, de sorte que les énergies de liaisons chimiques ne sont pas à leur minimum. Si on réchauffe le métal et qu'on le refroidit, les liaisons atomiques se réorganisent et il devient possible d'atteindre le minimum par une structure beaucoup plus ordonnée.

Dans le cas qui nous intéresse, on va perturber suffisamment un circuit hamiltonien initial dans le but d'échapper au minimum local. La perturbation contiendra un certain degré de hasard et sera répétée si besoin. Un paramètre permettra de régler l'évolution du système, nommé **température** en raison de l'analogie thermodynamique.

De manière plus détaillée, on initialise le paramètre  $T$  à une valeur positive déterminée empiriquement, et le circuit de manière quelconque, son coût  $\ell(\mu)$  étant connu. On tire au sort une transformation de ce circuit, par exemple une transformation de croisement d'arêtes expliquée paragraphe 4.5.2.

Il en résulte un circuit  $\mu'$  de coût  $\ell(\mu')$ .

Si  $\varepsilon = \ell(\mu') - \ell(\mu) < 0$ , on effectue la transformation qui va remplacer  $\mu$  par  $\mu'$ . Sinon, on tire au hasard un nombre compris entre 0 et 1. Si ce nombre dépasse  $e^{-\varepsilon/T}$ , on réalise quand même la transformation. Puis on diminue la « température » en la multipliant par un facteur proche de 1 et inférieur à 1. Il ne reste qu'à surveiller le moment auquel le coût du circuit sera redescendu en-dessous du meilleur coût déjà rencontré pour noter le circuit correspondant. On répète le processus autant de fois que nécessaire.

Les résultats obtenus pour de grands graphes sont généralement satisfaisants, à condition de choisir soigneusement les valeurs d'initialisation des divers paramètres et de ne pas reculer devant des milliers d'itérations qui convergent assez lentement.

### 4.5.4 Méthode des zones taboues

On peut simplifier l'approche précédente en notant que les coûts des divers circuits forment un ensemble discret. Par conséquent, on peut réaliser le travail de « sortie du voisinage » d'un minimum relatif en s'interdisant quelque temps de revenir sur les circuits déjà visités (d'où le nom de la méthode).

On part d'un circuit aussi bon que possible (grâce à une heuristique bien choisie). On applique sur ce circuit une transformation de croisement d'arêtes en cherchant le croisement optimal ; il peut en résulter un coût supérieur, mais on l'accepte. Puis on recommence la transformation sur le circuit trouvé, en s'interdisant de décroiser les mêmes arêtes (ce qui ferait revenir au point de départ), et ainsi de suite, en prohibant, pendant un nombre fixé d'itérations, de

revenir à des couples d'arêtes déjà essayés. De cette manière, on va essayer une quantité de circuits pour la plupart non optimaux, mais parmi lesquels certains seront des optimaux locaux et sans doute d'autres égaux à l'optimum absolu. Enfin, on gardera trace des « records » constatés, et on décidera de stopper les itérations après un temps fixé à l'avance.

## 4.6 Algorithmes distribués

Dans la réalité, les problèmes de circulation dans un réseau ne sont pas toujours gérés de manière centralisée, c'est-à-dire que l'information sur les longueurs des arcs n'est pas immédiatement disponible. Un bon exemple d'une telle situation est le réseau Internet, dans lequel les messages circulent d'un centre de services à l'autre sans que les longueurs des arêtes du graphe soient toutes connues ; en effet, le graphe est beaucoup trop grand pour que ce soit possible et efficace. De plus, les longueurs sont en fait des durées de transfert qui peuvent varier dans le temps et être infinies en cas de déconnexion.

La circulation des messages se fait, dans ces conditions, avec une version *distribuée* de l'algorithme de Bellman-Kalaba, dans laquelle les longueurs des arcs sont remplacées par des estimations obtenues par une interrogation des stations voisines. L'ouvrage [56] développe abondamment cette question.

On rencontre des problèmes similaires dans les systèmes de gestion du trafic automobile en temps réel (en tenant compte des embouteillages et interruptions) ainsi que les problèmes d'interrogation de bases de données réparties.

## 5. Arbres et arborescences

### 5.1 Définition des arbres

**Définition 41.** Un graphe non orienté, connexe, n'ayant aucun circuit (ou cycle) est appelé un **arbre**.

Un graphe non orienté n'ayant aucun circuit est appelé une **forêt**.

On dit qu'un sommet  $x$  d'un arbre est **pendant** s'il n'existe qu'une seule arête incidente à ce sommet. On dit qu'une arête est **terminale** si l'une de ses extrémités est pendante.

Il est évident qu'une forêt a pour composantes connexes des arbres (d'où la terminologie).

**Théorème 21.** Un arbre admet au moins deux sommets pendants.

**Preuve.**  $\diamond$  Considérons un arbre  $H$  n'ayant que 0 ou 1 sommet pendant, et imaginons un voyageur partant d'un sommet quelconque, se déplaçant le long des arêtes de  $H$  sans jamais suivre deux fois la même arête. D'une part, ce voyageur ne pourra pas passer deux fois par le même sommet, car  $H$  ne contient pas de cycle. D'autre part, si le voyageur parvient à un sommet  $x$ , il peut toujours en repartir car  $x$  n'est pas pendant. Dans ces conditions, le voyageur poursuit indéfiniment son chemin dans  $H$ , ce qui est absurde,  $H$  étant fini.  $\diamond$

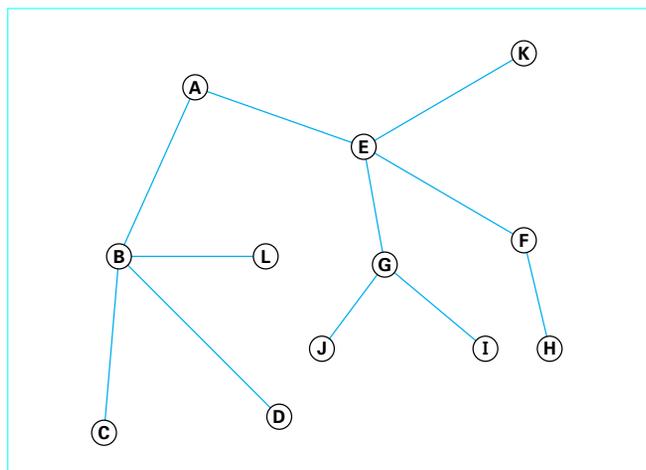


Figure 19 - Arbre

Dans la figure 19, les sommets pendants sont C, D, H, I, J, K, L. Ce sont les sommets de degré 1. On remarque la présence d'un sommet de degré 3 (G) et de sommets de degré 4 (B, E).

**Théorème 22.** Soit  $H$  un graphe ayant  $n$  sommets.

Les propositions suivantes sont équivalentes :

- a)  $H$  est connexe et sans cycle (donc est un arbre) ;
- b)  $H$  est sans cycle, et admet  $n - 1$  arêtes ;
- c)  $H$  est connexe, et admet  $n - 1$  arêtes ;
- d)  $H$  est sans cycle, et, en ajoutant une arête entre deux sommets non adjacents, on crée un cycle et un seul ;
- e)  $H$  est connexe, et, en supprimant une arête quelconque, il n'est plus connexe ;
- f) tout couple de sommets est relié par une chaîne et une seule.

**Preuve.**  $\diamond$

(a  $\Rightarrow$  b, c) Par récurrence. Soit  $m$  le nombre d'arêtes. On cherche un sommet pendant et on le détruit ainsi que l'arête terminale. On obtient aussitôt un graphe à  $n - 1$  sommets et  $m - 1$  arêtes, qui est toujours un arbre. Par hypothèse de récurrence, il vient

$$m - 1 = n - 2,$$

d'où le résultat.

(b  $\Rightarrow$  a) Par l'absurde. Supposons qu'il existe un graphe acyclique, non connexe et ayant  $n - 1$  arêtes. Joignons deux sommets appartenant à des composantes connexes distinctes par une nouvelle arête. Le graphe obtenu reste acyclique et possède  $n$  arêtes. S'il est connexe, c'est un arbre et il y a contradiction. Sinon, on recommence le procédé jusqu'à conclure de même.

(c  $\Rightarrow$  a) Par l'absurde. Supposons qu'il existe un graphe à  $n$  sommets,  $n - 1$  arêtes, connexe et avec au moins un cycle. Supprimons une arête d'un cycle : le graphe demeure connexe à cause du reste du cycle, mais il possède un cycle de moins. On recommence avec les autres cycles, jusqu'à aboutir à un graphe acyclique à  $n$  sommets et  $n - 1 - i$  arêtes,  $i$  étant le nombre de cycles supprimés. C'est donc un arbre à  $n$  sommets et  $n - i - 1$  arêtes, ce qui est exclu.

(a  $\Rightarrow$  d) En ajoutant une arête on n'a plus un arbre. La connexité demeure, donc c'est qu'un cycle est apparu. Si cette arête est commune à deux cycles, il existait déjà un cycle auparavant, exclu (faire un dessin).

Les autres implications sont aussi faciles et nous les omettons.  $\diamond$   
 Bien plus difficile est le théorème suivant. Une preuve peut être trouvée dans [13].

**Théorème 23 (de Cayley).** Le nombre d'arbres différents (non isomorphes) ayant  $n$  sommets vaut  $n^{n-2}$ .

## 5.2 Arbres couvrants sur un graphe

### 5.2.1 Principe

**Définition 42.** Soit un graphe  $G = (X, A)$ . Un sous-graphe de la forme  $(X, A')$  (ayant les mêmes sommets que  $G$ , mais moins d'arêtes) qui est un arbre est appelé un **arbre couvrant** de  $G$ .

**Théorème 24.** Un graphe  $G$  admet un arbre couvrant si, et seulement si, il est connexe.

**Preuve.**  $\diamond$  Pour construire un arbre couvrant on cherche une arête dont la suppression ne **déconnecte** pas le graphe. Si une telle arête n'existe pas, le graphe est un arbre. Si elle existe, on la supprime et on cherche une nouvelle arête sur laquelle on recommence l'opération.  $\diamond$

La figure 20 représente (en traits forts) un arbre couvrant pour le graphe de Petersen (ou de Micielski) (cf. figure 9, § 3.5.1). On remarque un phénomène très particulier : la « profondeur » de cet arbre est 2 sur toutes les branches.

On pourra consulter l'article [D 61] (§ 5.1), *Réseaux électriques linéaires*, pour une illustration pratique de la notion d'arbre couvrant d'un graphe.

### 5.2.2 Parcours d'un graphe et des arbres couvrants

Parcourir un graphe non orienté, c'est souvent déterminer un ordre de traitement de tous les sommets du graphe, et notamment disposer ces sommets en une liste ordonnée. Deux procédés permettent d'y parvenir. Dans chacun des cas, on part de l'un des sommets de l'arbre couvrant, appelé **racine**. De fait, l'arbre couvrant peut être construit de cette manière : on crée les branches les unes après les autres, tant qu'il est possible d'avancer dans le graphe sans rencontrer de sommet déjà pris.

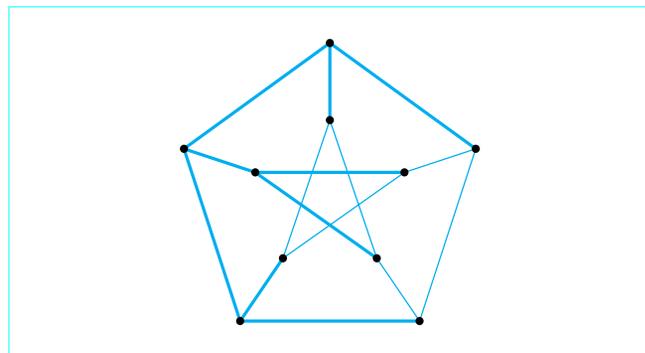


Figure 20 - Arbre couvrant pour  $M_4$

■ **Parcours en hauteur**

Il s'agit d'une exploration branche par branche. On choisit un successeur de la racine, puis un successeur de celui-ci et ainsi de suite. Lorsqu'il n'est plus possible de continuer ainsi (« marche à l'aveuglette »), on remonte d'un cran à la recherche d'un successeur non encore visité, et ainsi de suite.

L'arbre de la figure 19 peut être ainsi parcouru dans le sens **A, B, C, D, L, E, G, J, I, F, H, K**.

Le parcours en hauteur (parfois appelé « en profondeur ») est particulièrement naturel lors de l'usage d'un langage de programmation de nature récursive. En effet, si  $\text{action}(G, s)$  est une procédure ou fonction pouvant être appliquée sur un sommet  $s$  d'un graphe  $G$ , le schéma logique peut être le suivant.

- Objet préalable : un graphe non orienté  $G$ .
- Variable permanente (globale) : une liste  $M$  de « marques » booléennes affectées aux sommets de  $G$  et initialisées à la valeur « faux ».
- Procédure récursive  $\text{parcours}(a)$  :
  - le paramètre  $a$  est un sommet de  $G$
  - changer  $M(a)$  en « vrai »
  - et pour tout successeur  $x$  non marqué de  $a$ , appliquer  $\text{action}(G, x)$  puis  $\text{parcours}(x)$ .

■ **Parcours en largeur**

Il s'agit d'une exploration simultanée de toutes les branches. On repère tous les successeurs de la racine, puis les successeurs de ceux-ci et ainsi de suite en prenant garde de ne pas revenir en arrière. C'est une progression en « tache d'huile ».

L'arbre de la figure 19 peut être ainsi parcouru dans le sens **A, B, E, C, D, L, G, F, K, J, I, H**.

Nous avons déjà fait usage de cette méthode (§ 1.4.3 : § 3.7.1 ; § 4.4.1), pour la recherche des composantes connexes, de la bicoloration, des écarts à un sommet fixé.

**5.2.3 Problème des arbres couvrants minimaux**

Considérons le problème qui consiste à relier  $n$  villes par un réseau câblé de la manière la plus économique possible. On suppose connue la longueur  $\ell(u_{ij})$  de câble nécessaire pour relier les villes  $x_i, x_j$ . Le réseau doit évidemment être connexe et il ne doit pas admettre de cycles pour être de coût minimal ; c'est donc un arbre et ce doit être l'arbre le plus économique possible dans le graphe complet des  $n$  villes.

On voit, à partir de cet exemple, qu'un arbre couvrant minimal n'est autre qu'un graphe partiel connexe de longueur minimale.

Le problème se pose donc dans les termes suivants.

**Définition 43.** Soit un graphe non orienté  $G$ , connexe, pondéré par une fonction positive  $\ell$  attachée aux arêtes. Soit un arbre couvrant  $H = (X, B)$ , défini comme graphe partiel de  $G$  avec un ensemble d'arêtes  $B$  ; son poids (ou coût) total est

$$\ell(H) = \sum_{u \in B} \ell(u)$$

On dit que  $H$  est un **arbre de recouvrement de poids minimal** (ARPM) si  $\ell(H)$  est minimal parmi les poids de tous les arbres couvrants possibles.

On peut montrer que si toutes les arêtes sont de valeurs différentes,  $H$  est unique.

Plusieurs algorithmes ont été proposés pour résoudre ce problème ; en voici deux qui sont simples.

**5.2.4 Algorithme de Prim**

Il consiste à bâtir progressivement un arbre en partant de l'arête de poids minimal, et en y greffant, à chaque étape, l'arête de poids minimal parmi celles qui permettent de maintenir un graphe partiel qui soit un arbre. Si le graphe est connexe, le processus s'arrête avec un arbre couvrant. Sinon, il aboutit à un arbre couvrant pour une composante connexe ; on poursuit avec les autres composantes connexes s'il y en a, obtenant une forêt couvrante. L'arbre obtenu dépend, en général, de la numérotation des sommets et des arêtes du graphe, sauf si les arêtes ont des valeurs différentes les unes des autres.

L'algorithme de Prim est le type même d'un algorithme « glouton » (on devrait plutôt dire « avaro »). Le problème posé est de savoir si la décision « localement optimale » est aussi « globalement optimale ». Pour le justifier, considérons un problème un peu plus général de construction progressive d'une forêt couvrante.

**Lemme 2.** Soit un réseau pondéré

$$G = (X, A, \ell) \text{ et } F = (X_1, A_1), \dots, (X_k, A_k)$$

une forêt sur  $G$ , chacun des  $(X_i, A_i)$  étant un arbre.

Soit  $S$  l'ensemble des arêtes  $u = (a, b)$  telles que  $a \in X_1, b \notin X_1$  et  $S_m$  l'ensemble des arêtes de poids minimal parmi les arêtes de  $S$ . Alors :

- (i) tout ARPM incluant  $F$  doit aussi contenir au moins une arête de  $S_m$  ;
- (ii) si  $u = (a, b) \in S_m$  et s'il existe un ARPM incluant  $F$ , alors il existe un ARPM incluant  $F$  et  $u$ .

**Preuve.**  $\diamond$  Soit  $H$  un ARPM incluant  $F$  et  $u = (a, b) \in S_m$ .

(i) Supposons que  $H$  ne contienne pas  $u$ . Dès lors, l'adjonction de  $u$  à  $H$  va créer un cycle  $\mu$  et, sur ce cycle, il existe une arête :

$$u' = (c, d)$$

telle que  $c \in X_1$  et  $d \notin X_1$ .

Si  $H$  ne contient pas d'arête de  $S_m$ ,  $u'$  est de poids supérieur à  $u$ . On crée un nouvel arbre couvrant en supprimant  $u'$  de  $H$  et en ajoutant  $u$  à sa place. Le poids de cet arbre est inférieur à celui de  $H$ , ce qui est exclu.

(ii) La construction précédente permet, en fait, de remplacer toute arête de  $S_m$  apparaissant dans  $H$  par  $u$ .  $\diamond$

**Proposition 8.**

Tout arbre obtenu à l'issue de l'application de l'algorithme de Prim est un ARPM.

**Preuve.**  $\diamond$  Appliquons, d'abord, le lemme 2 à la forêt  $F$  formée par les sommets du graphe, isolés, et soit  $X_1$  une extrémité d'une arête  $u$  de poids minimal. Il existe évidemment un ARPM incluant  $F$ , n'importe lequel convient. L'énoncé (ii) montre qu'il existe aussi un ARPM incluant  $u$ .

Supposons que, à une certaine étape de l'algorithme, on ait déjà créé un arbre  $Z = (X_1, A_1)$  et que l'on sache l'existence d'un ARPM  $H$  incluant  $Z$ . Créons la forêt

$$F = Z, (X_2, \emptyset), \dots, (X_k, \emptyset),$$

les  $X_i$  étant les sommets de  $X \setminus X_1$  (pris isolément).

Soit  $u = (a, b) \in S_m$  (arête de poids minimal ayant une seule extrémité dans  $X_1$ ). Si  $H$  n'inclut pas  $u$ , on peut le modifier (d'après le lemme 2) pour obtenir un ARPM  $H'$  incluant  $Z$  et  $u$ . On obtient un arbre  $Z'$  ayant un sommet et une arête de plus, inclus dans un ARPM  $H'$ , et il suffit de recommencer.  $\diamond$

**Exemple :**

Dans le graphe représenté à la figure 21, l'algorithme de Prim peut conduire à différents arbres, selon les choix effectués quand il y a plusieurs arêtes de même poids qui peuvent être sélectionnées.

On a mis en évidence, à la figure 22, le début de l'algorithme, les arêtes retenues apparaissant en traits forts. L'algorithme se déroule sans aucun choix à faire jusqu'à atteindre le sommet « Grenoble ». On note au passage que l'arête « Annemasse-Annecy », de poids 38, ne peut être sélectionnée car elle crée un cycle avec les deux précédentes.

À partir de « Grenoble », on peut hésiter entre deux arêtes de même poids 70. Ici nous avons choisi l'arête se dirigeant vers « Valence », mais on aurait aussi bien pu ajouter à l'arbre en cours de construction l'arête « Genève-Pont-d'Ain ». La fin de l'algorithme apparaît à la figure 23.

**5.2.5 Algorithme de Kruskal**

C'est un autre exemple du « triomphe de l'avarice ». On choisit, à chaque étape, l'arête ayant la valeur la plus faible, ne formant pas de cycle avec les arêtes déjà choisies, jusqu'à obtenir un ensemble  $(u_1, u_2, \dots, u_{n-1})$  de  $n-1$  arêtes formant l'arbre désiré (dans le cas connexe).

Dans le cas non connexe, l'algorithme aboutit non pas à un arbre couvrant, mais à une forêt couvrante ; le nombre d'arêtes sélectionnées est alors inférieur.

On démontre que cet algorithme fournit effectivement un ARPM avec une méthode analogue à celle que l'on a suivie pour l'algorithme de Prim.

On peut également procéder en sens inverse, en éliminant de proche en proche les arêtes de plus forte valeur qui ne déconnectent pas le graphe quand on les supprime.

L'arbre obtenu n'est pas nécessairement le même que ceux que l'algorithme de Prim construit, sauf évidemment lorsqu'il y a unicité.

**Exemple :**

Reprenons le graphe représenté à la figure 21. L'algorithme de Kruskal apparaît aux figures 24 et 25 (l'arbre couvrant apparaît toujours en traits forts).

Il est intéressant de noter que le graphe partiel cesse assez vite d'être un arbre, dès que l'arête de poids 49 (« Lyon-Pont-d'Ain ») est retenue. La connexité du graphe partiel n'est récupérée qu'à la fin avec les arêtes de poids 70. On comprend bien alors qu'il n'y a que trois ARPM possibles sur ce réseau.

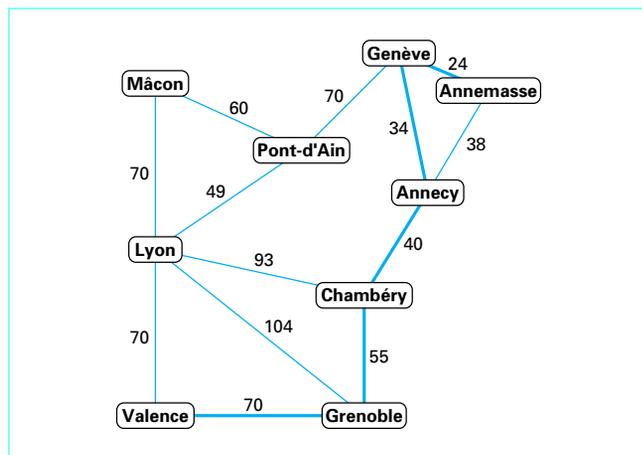


Figure 22 – Algorithme de Prim : début

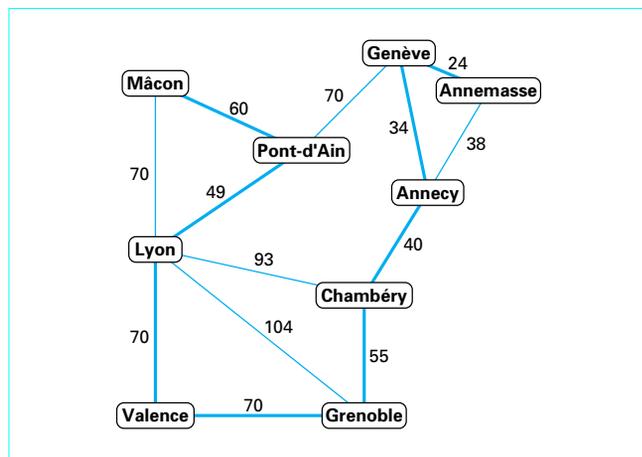


Figure 23 – Algorithme de Prim : l'arbre couvrant est obtenu

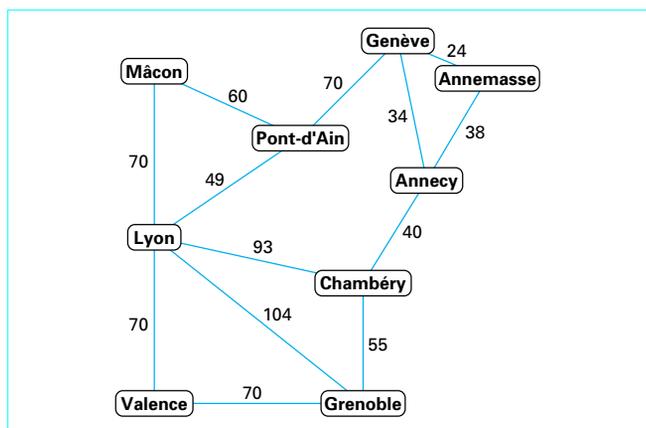


Figure 21 – Réseau autoroutier

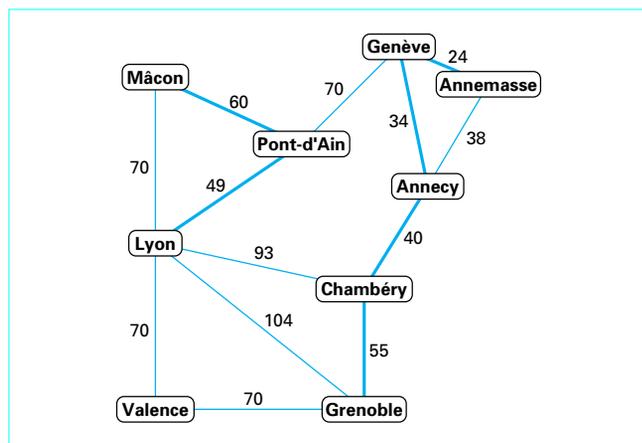


Figure 24 – Démarrage de l'algorithme de Kruskal

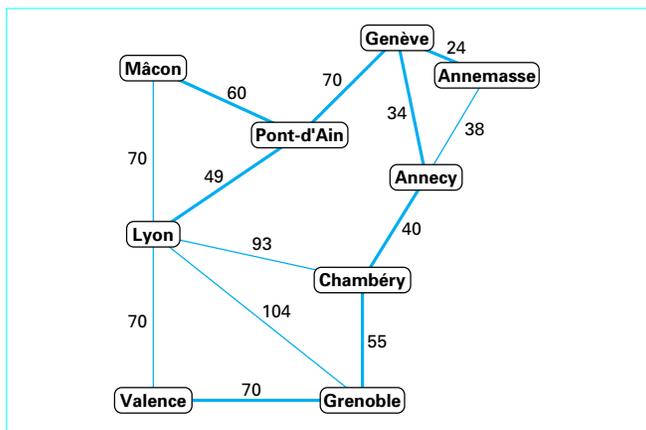


Figure 25 – Arbre couvrant obtenu à l'issue de l'algorithme

### 5.2.6 Problèmes d'optimisation mixte

Dans ce qui précède on a ignoré le fait que certaines arêtes de l'arbre sont plus « centrales » que d'autres. Ainsi, le débit supporté par ces arêtes devra être supérieur à celui attendu pour les arêtes terminales. En conséquence, le coût des arêtes centrales devrait être supérieur au coût des arêtes terminales. Dans ces conditions, le coût total va dépendre de la forme de l'arbre.

Ce genre de problème d'optimisation est beaucoup plus compliqué que le précédent. On peut empiriquement envisager une application itérative de la méthode de Kruskal : on calcule un premier arbre couvrant minimal pour un coût donné des arêtes, puis on ajuste les coûts pour tenir compte de la centralité de certaines arêtes, et on recommence.

### 5.2.7 Problèmes euclidiens

La recherche d'arbres couvrants de poids minimal s'applique facilement à des problèmes de minimisation de distances entre points, lorsque toutes les connexions sont possibles (graphe complet). Toutefois, on peut souhaiter, dans ce cadre, ajouter de nouveaux sommets (des « carrefours ») afin de diminuer la longueur totale du réseau. Ce problème est de nature géométrique et non plus topologique.

On appelle **arbre de Steiner** tout arbre de longueur minimale ayant parmi ses sommets l'ensemble des points imposés. On peut démontrer que tout sommet ajouté est nécessairement de degré 3 et que les trois segments incidents forment des angles de 120 degrés.

### 5.2.8 Système fondamental de cycles indépendants d'un graphe connexe

Dans le cas d'un graphe  $G$  connexe à  $n$  sommets et  $m$  arêtes, le rang  $\rho(G)$  vaut  $n - 1$  (§ 3.2) et le nombre cyclomatique  $\nu(G)$  vaut  $m - n + 1$ , exactement égal au nombre d'arêtes restantes lorsqu'on a déterminé un arbre couvrant de  $G$ .

Ainsi, pour trouver  $m - n + 1$  cycles indépendants dans  $G$ , il suffit de construire un arbre couvrant et de faire passer un cycle et un seul par chacune des arêtes qui n'appartiennent pas à l'arbre.

Si le graphe n'est pas connexe, on opère ainsi pour chacune de ses composantes connexes.

Cette méthode est couramment utilisée dans la théorie des réseaux électriques, où elle permet de construire une base de l'espace des courants de mailles.

### 5.2.9 Nombre d'arbres couvrants

Il est clair, d'après ce qui précède, que les arbres couvrants d'un graphe connexe peuvent, dans certains cas, être fort nombreux. On peut se demander quel est leur nombre, afin d'estimer le temps d'une recherche exhaustive.

Un exemple montrera que ce dénombrement est significatif. Considérons un graphe non orienté  $G$ , connexe, ayant  $n$  sommets et  $n$  arêtes ; il peut être vu comme un arbre avec une arête ajoutée entre deux de ses sommets.

Cette arête délimite un cycle  $(x_1, x_2, \dots, x_p)$  auquel les autres sommets sont reliés par des chaînes uniques.

On obtient une structure dite de « forêt enracinée » ou encore « arbres au bord d'un étang ». La figure 26 montre un tel graphe. Les arbres couvrants s'obtiennent en retirant une arête qui ne déconnecte pas le graphe, donc l'une des arêtes du cycle. Ainsi, il y a  $p$  arbres couvrants ( $p$  est la taille du cycle).

Il existe un lien intéressant entre les arbres couvrants et les matrices d'adjacence étendues (§ 2.3).

#### Proposition 9.

Soit  $M$  la matrice d'incidence d'un graphe  $G$  connexe. Alors  $n - 1$  arêtes de numéros  $j_1, \dots, j_{n-1}$  forment un arbre couvrant si, et seulement si, une sous-matrice carrée de  $M$  ayant ces numéros de colonne est inversible ; si c'est le cas, toutes ces sous-matrices sont inversibles.

**Preuve.**  $\diamond$  On traite le problème par transposition. Rappelons que  ${}^tM$  est singulière, car  ${}^tMV = 0$  si le vecteur  $V$  a toutes ses composantes égales à 1 ; ainsi, chaque colonne de  ${}^tM$  est l'opposée de la somme des autres colonnes et les sous-matrices de  ${}^tM$ , ayant des numéros de lignes  $j_1, \dots, j_{n-1}$ , ont des déterminants égaux au signe près. Nous savons déjà que ces déterminants valent 0, 1 ou  $-1$ . D'autre part, un choix de  $n - 1$  arêtes de  $G$  détermine un graphe partiel qui est un arbre (couvrant) si, et seulement si, il est connexe. Il suffit alors de se reporter à la preuve du théorème 6 pour conclure.  $\diamond$

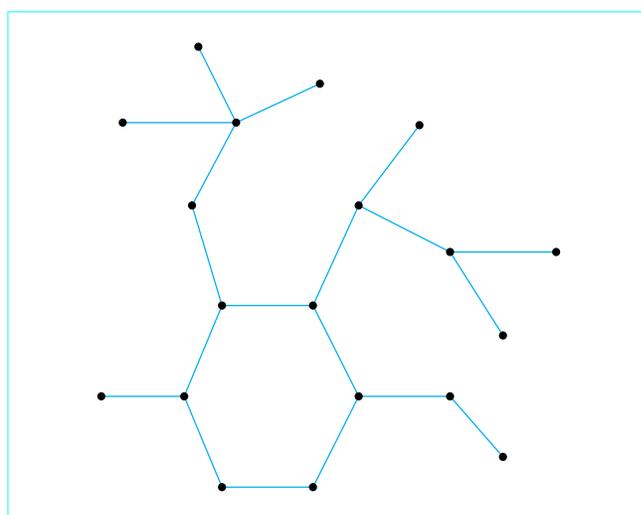


Figure 26 – Forêt enracinée

Le théorème suivant répond à la question posée. La preuve sera omise (outre la proposition 9, elle nécessite des concepts avancés d'algèbre linéaire comme la formule de Binet-Cauchy).

**Théorème 25.** Soit  $G$  un graphe non orienté à  $n$  sommets et  $B$  sa matrice d'adjacence étendue. Le nombre d'arbres couvrants de  $G$  est égal à la valeur absolue de n'importe quel cofacteur (ou mineur de taille  $n - 1$ ) de la matrice  $B$ .

### 5.3 Notion d'arborescence

La notion d'arborescence est l'adaptation de la structure d'arbre aux 1-graphes orientés. La figure 27 en est un exemple.

**Définition 44.**

Un graphe  $G$  est une **arborescence** s'il existe un sommet  $R$  appelé **racine** de  $G$  tel que, pour tout sommet  $S$  de  $G$ , il existe un chemin et un seul de  $R$  vers  $S$ .

Il résulte de cette définition que le graphe est acyclique et que c'est un arbre si on oublie son orientation ; on peut envisager d'autres définitions équivalentes.

La notion d'**arborescence couvrante** se définit comme celle d'arbre couvrant (§ 5.2.1), mais elle est un peu plus délicate, car il faut trouver une racine.

#### 5.3.1 Combinatoire des arborescences

Soit  $G$  un graphe orienté sans boucle,  $M$  sa matrice d'incidence et  $M^t M$  sa matrice d'adjacence étendue (voir le théorème 3). On cherche des renseignements sur le nombre d'arborescences couvrantes de sommet  $x$  fixé.

Soit  $\Delta_k$  le mineur obtenu en supprimant de  $M^t M$  la  $k$ -ième ligne et la  $k$ -ième colonne.

**Théorème 26 (de Tutte et Bott).** Soit  $G$  un graphe orienté sans boucle. Le nombre d'arborescences couvrantes de racine  $x_1$  est exactement  $\Delta_1$  ( $\Delta_1$  étant défini ci-dessus).

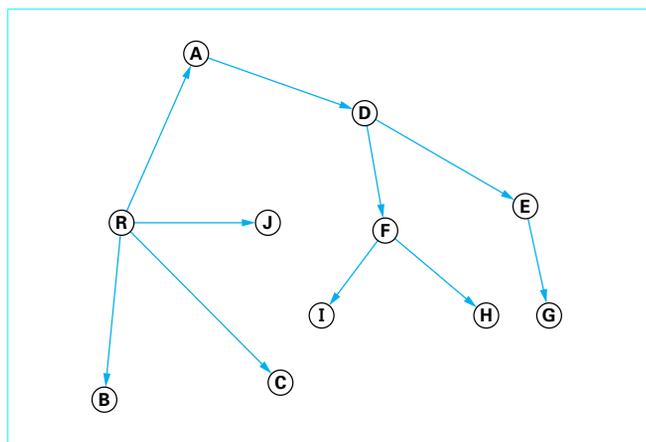


Figure 27 - Arborescence

Ce théorème peut aboutir à un algorithme de calcul pour les très grandes matrices, telles que celles que l'on rencontre en économétrie, comme la matrice de Léontief en particulier.

**Théorème 27 (de Bott et Mayberry).** Considérons un réseau  $R = (X, A, C)$ .

Soit  $C = (c_{ij})$  la matrice des capacités des arcs, telle que

$$c_{ij} = c(x_i, x_j).$$

Soit encore  $D = (d_{ij})$  avec :

$$d_{ij} = \begin{cases} 0 & \text{si } i \neq j \\ \sum_{k \neq i} c_{ki} & \text{si } i = j \end{cases}$$

et  $\Delta_1$  le mineur de  $D - C$  obtenu en supprimant la première ligne et la première colonne.

Soit enfin, pour une arborescence couvrante  $H$  de racine  $x_1$  :

$$c(H) = \prod_{x_i \in X, x_j \in \Gamma x_i} c_{ij}.$$

Alors  $\Delta_1$  est égal à la somme des  $c(H)$ , quand on prend pour  $H$  toutes les arborescences couvrantes de racine  $x_1$ .

#### 5.3.2 Arborescences couvrantes de poids minimal

Le problème de l'optimisation du choix d'une arborescence couvrante est plus difficile que pour les arbres couvrants, parce qu'il faut, en fait, vérifier que le sommet choisi au départ est bien racine de l'arbre. Un algorithme d'Edmonds répond à cette question. Le lecteur pourra se reporter à l'ouvrage [46] pour plus de détails.

## 6. Réseaux de transport

Dans un graphe orienté  $G$ , un **flot** est l'affectation d'une valeur réelle à chaque arc de  $G$ , représentant une quantité de matière transportée sur cet arc, de telle sorte que, en chaque sommet, la somme des flots entrants soit égale à la somme des flots sortants.

On se donne habituellement une capacité maximale sur chaque arc, qui sera une borne supérieure sur le flot autorisé sur cet arc ; le problème du flot maximal est alors de trouver un flot dont la valeur, en un certain lieu, est maximale.

On peut, de plus, se donner un coût de transport d'une unité de flot sur chaque arc, et chercher un flot maximal de coût minimal.

Ces problèmes admettent des algorithmes relativement efficaces et de nombreuses applications.

Au sujet des lois de Kirchhoff et des applications aux réseaux électriques, le lecteur pourra consulter avec profit les articles [D 61-64] (*Réseaux électriques, lois de Kirchhoff, nœuds et mailles*), [D 4 065] (*L'art de construire et de faire fonctionner un réseau électrique*), ainsi que [D 4 070] (*Réseaux de transport et d'interconnexion de l'énergie électrique*).

L'aspect informatique (création et fonctionnement d'un réseau d'ordinateurs) est abordé dans l'article [H 1 418] (*Réseaux locaux*).

## 6.1 Flots dans un réseau de transport

### 6.1.1 Vocabulaire

#### Définition 45.

On appelle **réseau de transport** un graphe orienté valué (cf. définition 6, § 1.1.2)  $G = (X, A, C)$ , sans boucle, et dans lequel il existe :

- a) un sommet  $x_1$  sans prédécesseur (c'est-à-dire  $\Gamma^{-1}x_1 = \emptyset$ ), nommé **entrée** ou **source** du réseau,
  - b) un sommet  $x_n$  sans successeur (c'est-à-dire  $\Gamma x_n = \emptyset$ ), nommé **sortie** ou **puits** du réseau,
- et tels qu'au moins un chemin unisse  $x_1$  à  $x_n$  dans  $G$ .

La fonction de pondération  $C$  est supposée positive, et on nomme **capacité** de l'arc  $u$  le nombre  $C(u)$ .

Si on désigne par  $A_x^-$  l'ensemble des arcs incidents au sommet  $x$  et orientés vers  $x$  (entrants) et par  $A_x^+$  l'ensemble des arcs incidents en  $x$  et orientés vers l'extérieur (sortants), on dit qu'une fonction  $\varphi(u)$  définie sur  $A$  et à valeurs réelles est un **flot** pour le réseau de transport si :

- (i) il est positif :

$$\varphi(u) \geq 0,$$

pour tout  $u \in A$  ;

- (ii) il vérifie la **loi des nœuds de Kirchhoff** :

$$\sum_{u \in A_x^-} \varphi(u) - \sum_{u \in A_x^+} \varphi(u) = 0,$$

pour tout sommet  $x \neq x_1, x \neq x_n$  ;

- (iii) il ne dépasse pas la capacité des arcs :

$$\varphi(u) \leq C(u)$$

pour tout  $u \in A$ .

Le nombre  $\varphi(u)$  peut alors être assimilé à une certaine quantité de matière circulant dans l'arc

$$u = (xy)$$

de  $x$  vers  $y$ , et n'excédant pas la capacité de l'arc. Si  $x$  n'est ni  $x_1$  ni  $x_n$ , le second axiome des flots exprime que la **quantité entrante** en  $x$  doit être égale à la **quantité sortante** que nous désignons par  $\varphi_x$  :

$$\varphi_x = \sum_{u \in A_x^-} \varphi(u) = \sum_{u \in A_x^+} \varphi(u).$$

#### Proposition 10.

Si  $\varphi$  est un flot sur le réseau de transport  $G$ , alors on a :

$$\varphi_{x_n} = \varphi_{x_1}.$$

Ce nombre représente la quantité de matière transportée à l'entrée comme à la sortie du réseau de transport ; on l'appelle la **valeur du flot**.

**Preuve.**  $\diamond$  On peut sommer la relation de l'axiome (ii) pour tous les sommets, en prenant garde aux rôles particuliers de  $x_1$  et  $x_n$  :

$$S = \sum_{\substack{x \in X \\ x \neq x_1, x \neq x_n}} \left( \sum_{u \in A_x^-} \varphi(u) - \sum_{u \in A_x^+} \varphi(u) \right) \tag{1}$$

$$+ \left( \sum_{u \in A_{x_1}^-} \varphi(u) - \sum_{u \in A_{x_1}^+} \varphi(u) \right) + \left( \sum_{u \in A_{x_n}^-} \varphi(u) - \sum_{u \in A_{x_n}^+} \varphi(u) \right)$$

Comme :

$$\sum_{u \in A_{x_1}^-} \varphi(u) = 0 \text{ et } \sum_{u \in A_{x_n}^+} \varphi(u) = 0,$$

on a :

$$S = - \sum_{u \in A_{x_1}^+} \varphi(u) + \sum_{u \in A_{x_n}^-} \varphi(u) = \varphi_{x_n} - \varphi_{x_1}.$$

D'autre part, la quantité  $S$  en (1) est nulle parce que, pour chaque arête  $u$ , on compte  $\varphi(u)$  une fois positivement à l'extrémité et une fois négativement à l'origine.  $\diamond$

Le second axiome des flots peut s'interpréter commodément à l'aide de la matrice d'incidence du graphe (§ 2.3). En effet, si on note  $M$  cette matrice (à  $n$  lignes et  $m$  colonnes) et  $\Phi$  le vecteur-colonne de composantes  $\varphi(u_j)$ , on a :

$$\sum_{j=1}^m m_{ij} \varphi(u_j) = 0 = M\Phi,$$

c'est-à-dire que  $\Phi$  appartient au noyau de  $M$ .

Il ne s'agit toutefois pas d'un vecteur quelconque de ce noyau puisque que la positivité est exigée, voire le caractère entier si les capacités sont entières.

### 6.1.2 Problèmes posés par les réseaux de transport

La question qui se pose, pour un réseau de transport donné, est d'abord de déterminer un flot sur le réseau ayant la valeur maximale possible, ainsi que de déterminer ce flot le long de chaque arc.

Dans de nombreux cas, les valeurs des flots et des **capacités** sont **entières** ; cette situation comporte l'avantage de permettre un algorithme fini.

Dans le cas de **capacités non entières**, on peut approcher les valeurs réelles par des valeurs rationnelles, réduire ces nombres au même dénominateur  $d$  et prendre comme unité de référence  $1/d$  ; on se ramène ainsi à des capacités entières. L'inconvénient de cette manière de procéder est qu'on obtient généralement des capacités très élevées, de sorte que l'algorithme de Ford-Fulkerson (§ 6.2) aboutit beaucoup plus lentement.

Il arrive aussi que l'on doive considérer des réseaux avec des **capacités localisées** non seulement sur les arêtes, mais encore sur les sommets du graphe ; c'est notamment le cas avec les réseaux téléphoniques pour lesquels la limite de capacité est autant due aux lignes qu'aux centraux. On peut ramener aisément ce problème au précédent : il suffit de doubler chaque sommet en une entrée et une sortie, liées par un arc ayant pour capacité celle qu'on attribuait auparavant au sommet.

On peut de même ramener l'étude des flots à plusieurs sources (resp. plusieurs puits) au cas fondamental à une seule source et un seul puits, en créant une « supersource » (resp. un « superpuits ») reliée aux sources (puits) initiales par autant d'arcs de capacités finies ou infinies selon la situation étudiée.

## 6.2 Méthode de recherche du flot maximal : algorithme de Ford et Fulkerson

Considérons un réseau de transport ayant un flot  $\varphi$  et cherchons à améliorer la valeur  $\varphi_{x_n}$ . Comme les capacités des arcs sont toutes

positives ou nulles, on peut, faute de mieux, commencer avec le flot nul. Le lecteur est invité à s'appuyer sur l'exemple traité au paragraphe 6.2.4.

### 6.2.1 Recherche d'un flot complet

**Définition 46.** Pour un flot  $\varphi$  sur un réseau de transport  $G = (X, A, C)$  on dit qu'un arc  $u$  est **saturé** si on a

$$\varphi(u) = C(u).$$

Le flot est dit **complet** si tout chemin allant de  $x_1$  à  $x_n$  contient au moins un arc saturé.

Considérons le graphe partiel (définition 5) engendré par les arcs non saturés par le flot. Si le flot n'est pas complet, il existe, dans ce graphe partiel, un chemin  $\mu$  allant de l'entrée à la sortie.

Soit alors :

$$\varphi'(u) = \begin{cases} \varphi(u) + 1 & \text{si } x \in \mu \\ \varphi(u) & \text{si } x \notin \mu \end{cases}$$

$\varphi'$  est encore un flot sur le réseau et on a évidemment

$$\varphi'_{x_n} = \varphi_{x_n} + 1 > \varphi_{x_n}.$$

On peut donc facilement augmenter (de 1 en 1) la valeur d'un flot incomplet jusqu'à ce qu'il soit complet. En tenant compte des différences entre les capacités et la valeur du flot sur les arcs de  $\mu$ , on peut connaître d'avance l'augmentation possible du flot. Ce n'est pas pour autant, en général, un flot maximal.

### 6.2.2 Amélioration du flot

Soit  $\varphi$  un flot complet. On va utiliser une procédure itérative pour identifier, et marquer tous les sommets du graphe où il est possible de faire parvenir une unité de flot supplémentaire.

#### ■ Processus d'étiquetage

On définit alors un processus d'étiquetage de certains sommets du graphe. En  $x_1$  on place une étiquette  $\boxed{+}$ . Soit  $x_j$  un sommet déjà marqué :

a) on marque avec une étiquette  $\boxed{+x}$  tout successeur  $y$  non marqué de  $x$  pour lequel le flot n'est pas à son maximum, c'est-à-dire tel que

$$(x, y) \in A$$

et  $\varphi(x, y) < C(x, y)$  ;

b) on marque avec une étiquette  $\boxed{-x}$  tout prédécesseur  $y$  non marqué de  $x$  pour lequel le flot n'est pas nul, c'est-à-dire tel que

$$(y, x) \in A$$

et  $\varphi(y, x) > 0$ .

La signification de l'étiquette est de donner le nom d'un prédécesseur ou successeur d'un moment donné et d'indiquer si le flot peut être augmenté dans le sens du parcours (étiquette  $\boxed{+x}$ ) ou diminué s'il est dans le sens contraire.

#### ■ Modification du flot

Si on parvient jusqu'au marquage du sommet  $x_n$  avec cette procédure, c'est qu'il existe, dans le réseau étendu, un chemin  $\mu$  de  $x_1$  vers  $x_n$  dont tous les sommets sont marqués avec l'indice du sommet précédent au signe près.

Soit alors :

$$\begin{cases} \varphi'(u) = \varphi(u) & \text{si } u \notin \mu ; \\ \varphi'(u) = \varphi(u) + 1 & \text{si } u \in \mu \text{ et si } u \text{ est orienté dans le sens de } \mu ; \\ \varphi'(u) = \varphi(u) - 1 & \text{si } u \in \mu \text{ et si } u \text{ est orienté dans le sens contraire de } \mu. \end{cases}$$

On vérifie sans peine que  $\varphi'$  est encore un flot.

Comme  $\varphi'_{x_n} = \varphi_{x_n} + 1$ , la valeur du flot  $\varphi'$  est meilleure que celle de  $\varphi$ .

#### ■ Interprétation

Il est possible de donner un cadre théorique à la méthode précédente.

**Définition 47.** Soit un réseau de transport  $(X, A, C)$ , avec un flot complet  $\varphi$  sur celui-ci. On appelle **réseau résiduel** le réseau  $(X, A', C')$  tel que :

- (i) si  $u \in A$  et  $\varphi(u) < C(u)$  alors :  
 $u \in A'$  et  $C'(u) = C(u) - \varphi(u)$  ;
  - (ii) si  $u \in A$  et  $\varphi(u) = C(u)$  alors :  
 $u \notin A'$  ;
  - (iii) si  $u = (x, y) \in A$  et  $\varphi(u) > 0$ , alors :  
 $u^{-1} = (y, x) \in A'$
- et  $C'(u^{-1}) = \varphi(u)$ .

Le réseau résiduel indique le long de quels arcs on pourrait augmenter ou diminuer le flot sans remettre en cause sa nature (positif ou nul et inférieur ou égal aux capacités). Faire usage d'un arc en retour (axiome (iii)) revient à s'autoriser un « regret » dans la définition du flot ; la valeur que l'on peut retirer ne doit pas dépasser la valeur du flot antérieur, d'où (iii).

Enfin, le fait de parvenir à traverser le réseau de part en part selon le processus d'étiquetage revient, très exactement, à trouver un chemin dans le réseau résiduel de la source vers le sommet et à augmenter ou diminuer le flot d'une unité le long des arcs de ce chemin selon qu'ils sont orientés vers les puits ou vers la source.

### 6.2.3 Valeur maximale du flot. Théorème de Ford-Fulkerson

Il va de soi qu'on ne peut augmenter indéfiniment la valeur du flot par la méthode précédente. Considérons donc un flot  $\varphi^0$  qui ne puisse plus être amélioré ainsi.

Soit  $S$  l'ensemble des sommets marqués et  $T$  l'ensemble des sommets non marqués : ils réalisent une partition du réseau. Plus généralement, on peut poser la définition suivante.

**Définition 48.** On appelle **coupe** du réseau  $R = (X, A, C)$  toute partition de l'ensemble des sommets en deux sous-ensembles disjoints  $S$  et  $T$  tels que l'entrée (source)  $x_1$  appartienne à  $S$  et la sortie (puits)  $x_n$  à  $T$ .

On note  $A_{S,T}$  l'ensemble des arcs du réseau ayant leur origine dans  $S$  et leur extrémité dans  $T$ .

On appelle **capacité de la coupe**  $(S, T)$  la quantité :

$$C(S, T) = \sum_{u \in A_{S,T}} C(u).$$

Nous pouvons à présent comparer les concepts de valeur du flot et de la capacité d'une coupe.

**Théorème 28 (de Ford-Fulkerson).** La valeur maximale du flot sur le réseau  $(X, A, C)$  coïncide avec la capacité minimale d'une coupe du réseau, ce qui peut s'écrire :

$$\text{Max}_{\varphi} \varphi_{x_1} = \text{Min}_{S \subset X} C(S, X \setminus S)$$

$$x_1 \in S, x_n \in X \setminus S$$

**Preuve.**  $\diamond$  Considérons d'abord un flot  $\varphi$  et une coupe  $(S, T)$ . En ajoutant les identités de Kirchhoff relatives aux sommets de  $S$ , on trouve

$$\varphi_{x_1} + \sum_{\substack{u=(x,y) \\ y \in S, y \neq x_1}} \varphi(u) = \sum_{\substack{u=(x,y) \\ x \in S}} \varphi(u)$$

ce qui revient à dire que le flot total sortant de  $S$  est égal au flot entrant en  $S \setminus \{x_1\}$ . Décomposons les arcs ayant leur origine et leur extrémité dans  $S$  :

$$\varphi_{x_1} + \sum_{\substack{u=(x,y) \in A(S,S) \\ y \neq x_1}} \varphi(u) = \sum_{u \in A(S,T)} \varphi(u) \quad (2)$$

La positivité du flot amène alors l'inégalité

$$\varphi_{x_1} \leq \sum_{u \in A(S,T)} \varphi(u) \leq C(S, T),$$

donc la valeur du flot est inférieure ou égale à la capacité de la coupe.

En conséquence, s'il existe un flot dont la valeur atteint la capacité d'une coupe, il est certainement maximal.

Réciproquement, considérons un flot maximal  $\varphi$ . Il n'est certainement pas améliorable par la méthode exposée paragraphe 6.2.2. Si on considère la coupe formée des sommets marqués ( $S$ ) et des sommets non marqués ( $T$ ), il n'y a ni arc non saturé allant de  $S$  vers  $T$ , ni arc de flot non nul allant de  $T$  vers  $S$ , par le fait même que le processus d'étiquetage est arrivé à son terme. En reprenant l'équation (2), on trouve ici :

$$\varphi_{x_1} + 0 = \sum_{u \in A(S,T)} \varphi(u) = C(S, T)$$

et la capacité de la coupe est bien égale à la valeur de ce flot.  $\diamond$

### 6.2.4 Recherche du flot maximal : exemple

#### ■ Départ

Considérons l'exemple de la figure 28, où les quantités indiquées à côté des flèches représentent les capacités respectives. Dans cet exemple, l'entrée sera le sommet **S** (source) et la sortie le sommet **P** (puits).

On fait pénétrer en **S** un certain flot, en nombres entiers, qui sera propagé de sommet en sommet de telle sorte qu'en chaque sommet la propriété conservatrice de flot (définition 45, (ii)) soit vérifiée ; cela est représenté sur la figure 29. Ainsi, au sommet **A**, on a bien :

$$6 + 1 = 3 + 1 + 3.$$

Notons que si, à cause des capacités, le flot était trop grand, on le diminuerait jusqu'à faire passer un flot compatible avec les capacités.

#### ■ Recherche d'un flot complet

On commence par chercher un flot comportant au moins un arc saturé, puis on essaye de saturer des chemins allant de la source au

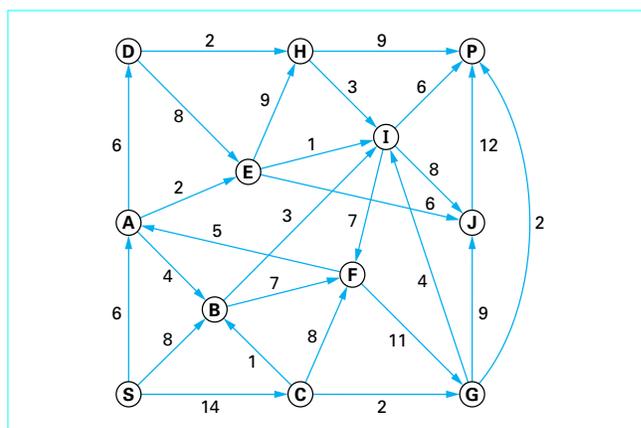


Figure 28 – Capacité respective de chaque arc

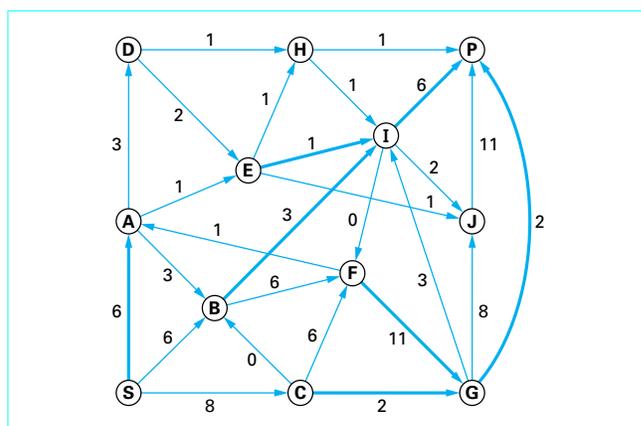


Figure 29 – Flot au jugé

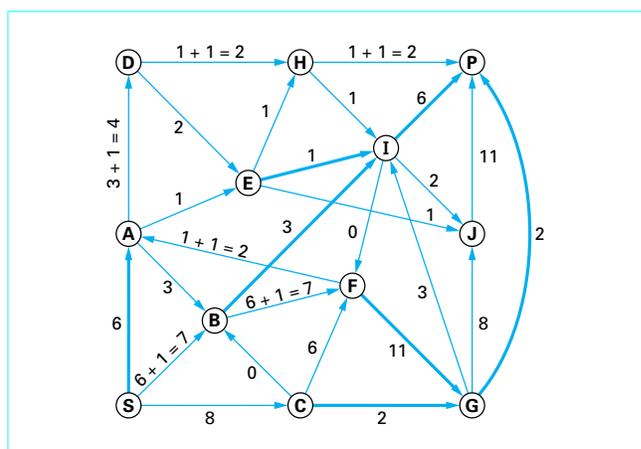


Figure 30 – Recherche d'un flot complet

puits. Sur la figure 29, on a indiqué en trait fort les arcs saturés. Le chemin  $(S, B, F, A, D, H, P)$  ne passe par aucun arc saturé.

On augmente alors le flot de ce chemin d'une unité (figure 30), ce qui saturé les arcs  $(B, F)$  et  $(D, H)$ .

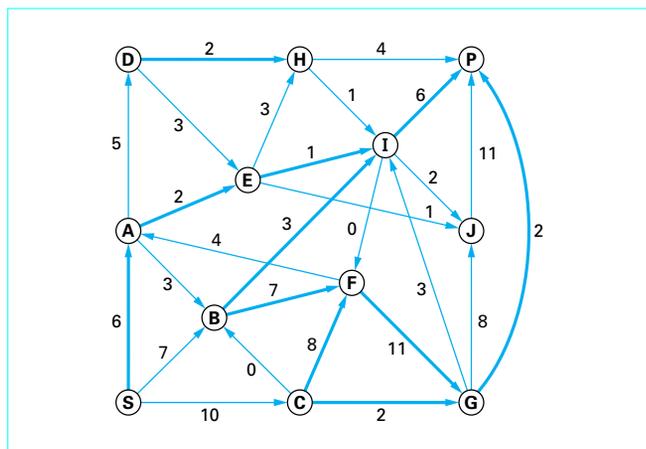


Figure 31 – Obtention d'un flot complet

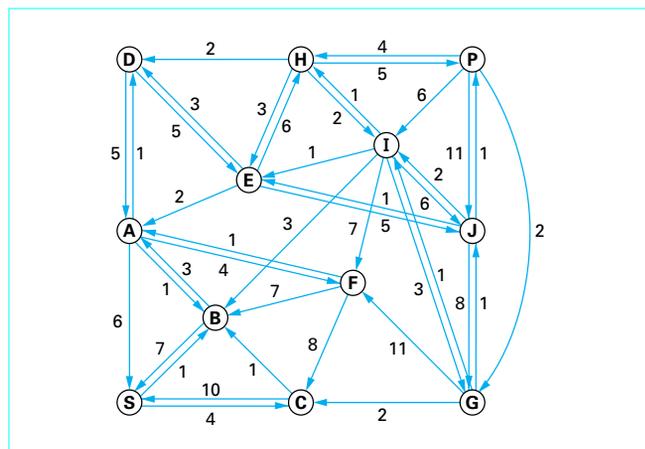


Figure 33 – Réseau résiduel

Après plusieurs opérations du même type on obtient la figure 31, qui définit un flot complet.

■ Amélioration du flot

On procède à l'étiquetage des sommets. S est marqué  $\oplus$  (figure 32). On choisit, par exemple, comme successeur du sommet S le sommet C qu'on marque avec  $\oplus C$ , l'arc (S,C) étant non saturé, et le sommet B avec  $\oplus B$ . Par contre, le sommet A doit être marqué  $\ominus A$ , l'arc non saturé étant orienté en sens inverse. De même, on marque le sommet D avec  $\oplus A$ , l'arc (A,D) étant non saturé, et on poursuit pareillement en passant par les sommets E et H, jusqu'au sommet P qui reçoit la marque  $\oplus H$ .

Comme on a pu marquer la sortie, le flot total n'est pas optimal.

On peut aussi examiner le réseau résiduel, qui est représenté à la figure 33.

On augmente alors, à partir de la figure 32, d'une unité le flot des arcs dirigés dans le sens du chemin et on diminue d'une unité les autres arcs. De cette façon, on sature les arcs (B,C), (A,D) et (H,P) (figure 34).

En reprenant le processus, on voit que le marquage n'est plus possible au-delà du sommet F. L'arc (F, I) étant dirigé dans le sens inverse du chemin et de flot nul, ne pourrait ensuite être diminué d'une unité, les flots étant toujours positifs.

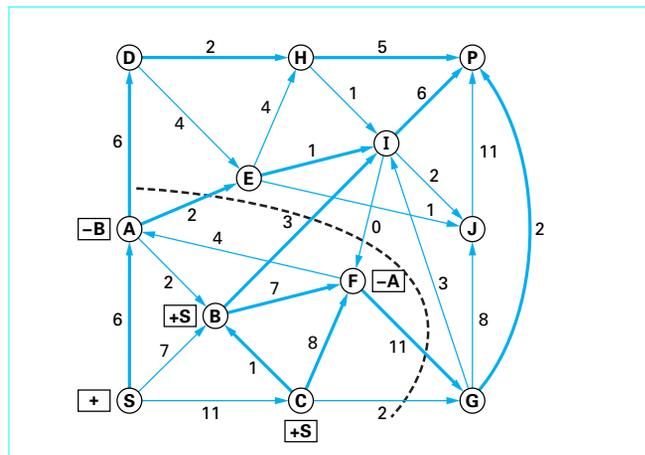


Figure 34 – Flot maximal et coupe de ce flot

■ Valeur maximale du flot

On a donc trouvé un flot global maximal. Pour le constater, réalisons sur la figure 34 une coupe au sens défini au paragraphe 6.2.3.

Il suffit de tracer une ligne séparant les sommets marqués des sommets non marqués. La coupe est délimitée par des arcs saturés orientés des sommets marqués vers les sommets non marqués, ainsi que par des arcs de flot nul orientés en sens inverse. On voit alors qu'il n'est pas possible d'augmenter le flot. Le flot maximal a pour valeur

$$6 + 2 + 3 + 11 + 2 = 24.$$

### 6.3 Problème des flots de coût minimal

#### 6.3.1 Position du problème

En pratique, la plupart des réseaux de transport sont associés à une notion de coût unitaire du transport, associé à l'usage de chaque arête. Pour simplifier, nous supposons que le coût du passage de  $\varphi(u)$  unités le long de l'arc  $u$  est proportionnel à  $\varphi(u)$ . Dans un modèle plus réaliste, il faudrait inclure une « prise en charge » fixe pour l'usage de chaque arc.

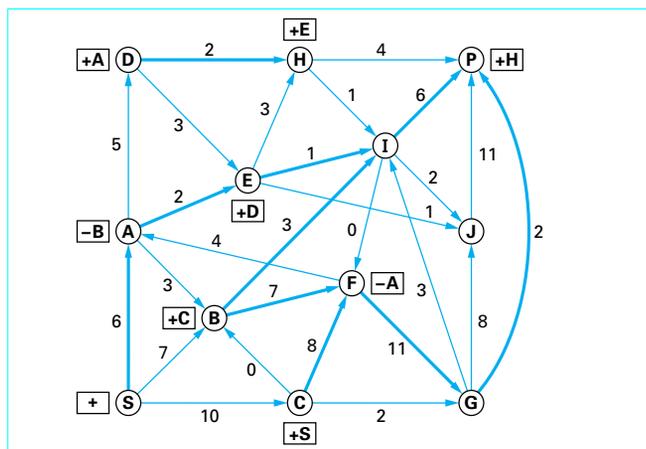


Figure 32 – Amélioration du flot

Le coût est donc modélisé par une fonction  $w : A \rightarrow \mathbb{N}$  et le coût d'un flot  $\varphi$  est :

$$W(\varphi) = \sum_{u \in A} w(u)\varphi(u).$$

Dans ces conditions, le problème est de trouver un flot de coût minimal parmi les flots de débit maximal (ou valeur maximale).

Nous admettrons la proposition suivante.

**Proposition 11.**

Un flot  $\varphi$  sur le réseau  $R$  est de coût minimal si, et seulement si, le réseau résiduel associé à  $\varphi$  ne contient aucun cycle de coût strictement négatif.

### 6.3.2 Algorithme de Busacker-Gowen

Cet algorithme est une modification de l'algorithme de Ford et Fulkerson qui permet une prise en compte des coûts de différents arcs.

On commence avec un flot nul, de coût nul. Puis, de manière itérative, un flot  $\varphi$  de valeur  $\Phi$  étant obtenu, on recherche, dans le réseau résiduel associé à  $\varphi$ , un chemin allant de  $x_1$  à  $x_n$  qui soit de coût minimal. Cette recherche peut être accomplie avec l'algorithme de Bellman-Kalaba (§ 4.4.4). Si aucun chemin de cette sorte n'existe, c'est que le flot  $\varphi$  est maximal. Sinon, on améliore  $\varphi$  au moindre coût, obtenant un flot  $\varphi'$  de valeur  $\Phi'$ , et on peut recommencer.

On peut démontrer que, parmi tous les flots de valeur  $\Phi'$ ,  $\varphi'$  est de coût minimal.

Le lecteur trouvera une discussion détaillée sur l'algorithme de Busacker-Gowen dans l'ouvrage [46].

## 7. Graphes bipartis et couplages

### 7.1 Position du problème et vocabulaire

#### 7.1.1 Généralités

**Définition 49.**

(i) Un **graphe biparti** est la donnée d'un graphe  $G = (S, \Gamma)$  et d'une partition de  $G$  en deux sous-ensembles  $X$  et  $Y$ , tels que, pour tout  $x \in X$ , on ait :

$$\Gamma x \subset Y$$

et  $\Gamma^{-1}x \subset Y$ .

Cela revient à exiger que toute arête ait l'une de ses extrémités dans  $X$  et l'autre dans  $Y$ . On notera alors aussi :

$$G = (X, Y, \Gamma).$$

(ii) Soit un graphe  $G = (S, A)$ . Un **couplage**  $W$  est un ensemble d'arcs (ou arêtes)  $W$  deux à deux non adjacents.

(iii) Un couplage  $W$  sur  $G$  est dit **maximal** s'il possède le nombre maximal d'arcs.

(iv) Un couplage  $W$  sur  $G$  est dit **parfait** si tout sommet de  $G$  est extrémité de l'un des arcs de  $W$ .

Ces définitions sont intimement liées entre elles. En effet, un couplage parfait est certainement maximal, la réciproque étant généralement fautive (notamment, parce que le couplage parfait exige un nombre pair de sommets).

En outre, la notion de graphe biparti est liée à celle de bicoloration (voir le § 3.7.1) : si  $G = (S, \Gamma)$  est biparti avec  $S = X \cup Y$ , il suffit de colorier en rouge les sommets situés dans  $X$  et en bleu ceux qui appartiennent à  $Y$  pour avoir une bicoloration et, réciproquement, toute bicoloration de  $G$  permet une bipartition. Le théorème 11 entraîne aussitôt le résultat suivant.

**Proposition 12.**

Un graphe peut être biparti si, et seulement si, il n'a aucun cycle de longueur impaire.

Lorsque le graphe est biparti (soit  $G = (X, Y, \Gamma)$ ), on ne s'intéresse généralement qu'aux couplages bipartis, c'est-à-dire aux couplages  $W$  dont chaque arête a une extrémité dans  $X$  et l'autre dans  $Y$ . Lorsque tout sommet de  $X$  appartient à une arête du couplage, on dit qu'on a un **couplage de  $X$  dans  $Y$** . La figure 35 représente un exemple de graphe biparti avec un couplage parfait sur celui-ci.

**Exemple :** Les problèmes d'allocations ou d'affectations fournissent des exemples de couplages.

■ Ainsi, considérons un atelier où se trouvent  $p$  ouvriers et  $q$  postes de travail. On voudrait affecter les ouvriers aux postes de travail, sachant que chaque ouvrier est qualifié pour un ou plusieurs postes de travail (chaque poste de travail ne pouvant recevoir qu'un seul ouvrier). Est-il possible d'affecter chacun à un poste pour lequel il est qualifié ? On désigne par  $\Gamma x$  l'ensemble des postes pour lesquels l'ouvrier  $x$  est qualifié. Le problème posé consiste à considérer un graphe  $(X, Y, \Gamma)$  et à coupler  $X$  dans  $Y$ .

Un couplage maximal est une affectation dans laquelle un nombre maximal d'ouvriers sont occupés.

Un couplage parfait nécessite  $p = q$  et consiste à occuper tous les ouvriers et tous les postes de travail.

■ Une autre situation intéressante est rencontrée lors de l'affectation d'un ensemble de cours  $X$  à un ensemble d'enseignants  $Y$ . Au départ, on connaît seulement les habilitations des enseignants qui définissent un graphe biparti  $G = (X, Y, \Gamma)$ . On cherche à assurer tous les cours sachant qu'un même cours ne peut pas être assuré par deux enseignants différents et qu'un enseignant ne prend qu'un seul cours. La meilleure solution est fournie par un couplage maximal. Dans la réalité, les choses peuvent être plus compliquées si on considère que certains enseignants peuvent assurer plusieurs cours (par regroupement de classes) !

■ Dans un cadre plus mathématique, soit deux ensembles finis disjoints  $X$  et  $Y$ , et une application surjective  $\Gamma$  de  $X$  dans  $Y$ . On obtient ainsi un graphe biparti  $G = (X, Y, \Gamma)$ . Un couplage de  $X$  dans  $Y$  consiste à trouver une restriction de  $\Gamma$  à un sous-ensemble  $X'$  sur lequel  $\Gamma$  soit injective.

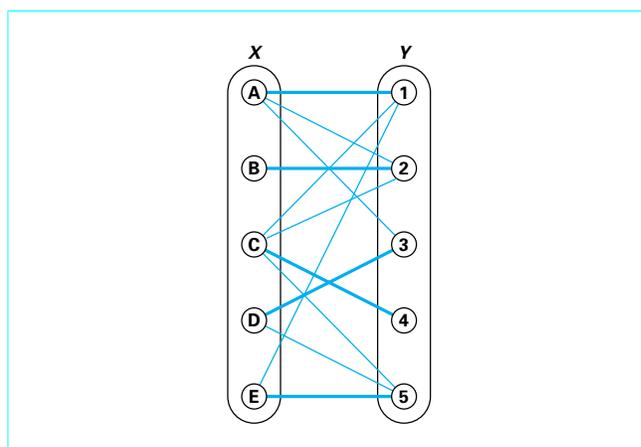


Figure 35 – Graphe biparti et couplage parfait

### 7.1.2 Considérations théoriques

■ **Couplabilité d'un ensemble dans un autre**

Étant donné un graphe biparti  $G = (X, Y, \Gamma)$ , il n'est pas toujours possible de coupler  $X$  dans  $Y$ , notamment, si  $X$  a plus d'éléments que  $Y$ . Le théorème suivant donne un critère d'existence.

**Théorème 29 (de König-Hall).** Soit un graphe biparti  $G = (X, Y, \Gamma)$ . On peut coupler  $X$  dans  $Y$  si, et seulement si,  $|\Gamma A| \geq |A|$  pour tout ensemble  $A \subset X$ .

Ce théorème revient à affirmer que si de tout sous-ensemble  $A$  de  $X$  contenant  $k$  sommets partent au moins  $k$  arcs, on peut toujours trouver un couplage.

■ **Usage d'un flot pour le couplage maximal**

Pour déterminer un couplage maximal sur un graphe biparti  $G = (X, Y, \Gamma)$ , on peut recourir à l'artifice suivant. On construit un réseau de transport (définition 45) en ajoutant au graphe une origine  $S$  liée à tous les sommets de  $X$  et une destination  $P$  liée à tous les sommets de  $Y$  (figure 36). On donne des capacités égales à 1 à tous les arcs de ce réseau.

Un couplage permet de sélectionner un certain nombre d'arêtes entre  $X$  et  $Y$ ; si on fait passer une unité de flot par ces arêtes et sur les arêtes correspondantes vers  $S$  et  $P$  et zéro sur les autres arêtes, on obtient bien un flot sur le réseau. En effet, la loi de Kirchhoff en un sommet  $x$  de  $X$  (ou de  $Y$ ; voir la définition 45) exprime ici simplement qu'il y a zéro ou une arête du couplage qui y parvient. Réciproquement, un flot sur ce réseau ne peut transporter qu'une unité de flot au plus sur chaque arête, et l'ensemble des arêtes transportant une unité de flot entre  $X$  et  $Y$  détermine un couplage sur  $G$ ; en effet, si deux arêtes ainsi retenues étaient adjacentes en  $x \in X$ , la loi de Kirchhoff en  $x$  imposerait un flot de 2 unités au moins sur l'arc entre  $S$  et  $x$ , ce qui dépasse la capacité imposée. La valeur du flot est égale au nombre d'arêtes figurant dans le couplage.

Le problème du couplage maximal se ramène ainsi à un problème de flot maximal dans un réseau de transport, qui peut être résolu à l'aide de l'algorithme de Ford-Fulkerson (§ 6.2). La figure 36 montre un flot complet et non maximal, qui peut être amélioré en un flot maximal représenté à la figure 37.

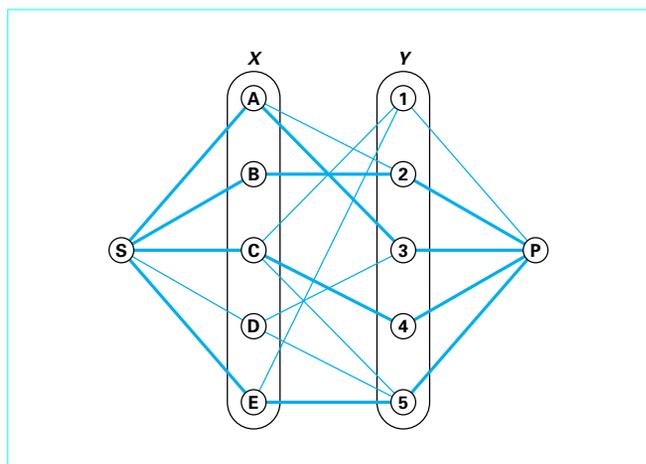


Figure 36 – Réseau de transport associé au graphe biparti

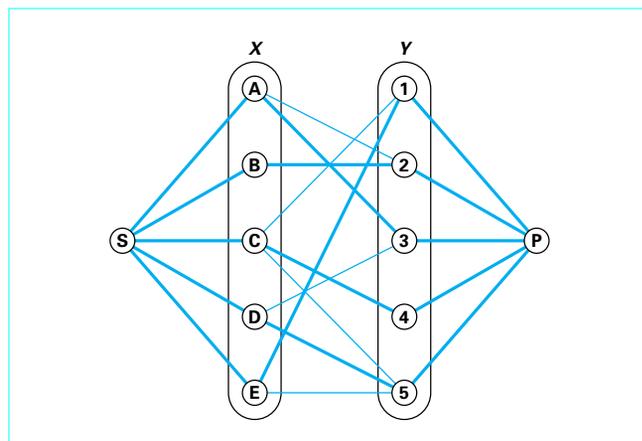


Figure 37 – Flot et couplage maximaux sur le graphe de la figure 36

## 7.2 Problèmes d'affectation

### 7.2.1 Généralités

Ils peuvent être résolus par l'utilisation de ce que l'on a appelé l'algorithme hongrois.

Soit  $p$  ouvriers  $x_1, x_2, \dots, x_p$  et  $p$  postes  $y_1, y_2, \dots, y_p$ . A toute affectation  $(x_i, y_j)$  ( $i$  et  $j$  étant compris entre 1 et  $p$ ) est attribué un « poids » :

$$C_{ij} \geq 0.$$

Si l'affectation envisagée est impossible, on pose

$$C_{ij} = +\infty.$$

Les  $C_{ij}$  peuvent être, suivant le cas, des coûts, des durées, etc.

Le problème consiste à affecter les  $p$  ouvriers aux  $p$  postes de façon que tous les ouvriers aient un poste et un seul (il s'agit donc d'un couplage parfait) et que la valeur totale des affectations soit minimale.

Le problème, plus simple, de la recherche d'un couplage biparti parfait peut être traité de cette manière en définissant des poids tous égaux à 1 ou à l'infini.

**Avertissement :** les personnes sont repérées alphabétiquement de **A** à **E** ; la matrice des poids utilise aussi l'ordre alphabétique pour l'indexation des lignes, ainsi on a (figure 38) :

$$C_{A1} = 7.$$

**Exemple :** les données sont indiquées figure 38 par le graphe et le tableau (ou matrice) des  $C_{ij}$ . Ainsi, la personne **A** est qualifiée pour les tâches **1** à **5**, mais son affectation à la tâche **2** coûte 3 unités tandis que son affectation à la tâche **5** coûte 10 unités.

Le but poursuivi est d'affecter à chaque symbole alphabétique de ligne  $i$  un numéro de colonne  $\sigma(i)$ , distinct pour chaque ligne, de sorte que la somme des poids  $C_{i\sigma(i)}$  soit minimale (et non infinie). En conséquence, tout revient à choisir  $\sigma$  qui est une permutation de l'ensemble des entiers de 1 à  $p$ .

Comme il y a  $p! = p \cdot (p - 1) \dots 2 \cdot 1$  telles permutations, on voit qu'il ne serait pas efficace de toutes les essayer.

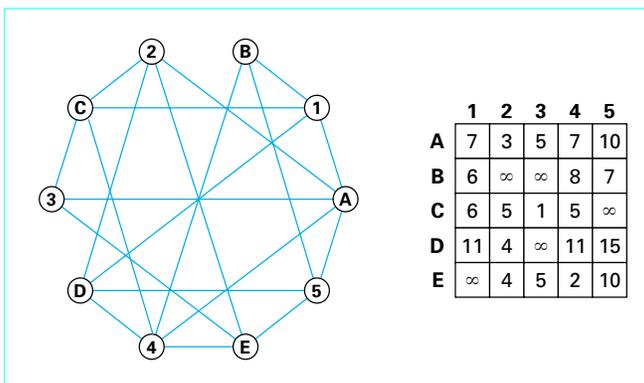


Figure 38 – Problème d’affectation

**Lemme 3.** On ne change pas la ou les solutions optimales en augmentant ou en diminuant d’une même quantité  $\lambda$  tous les éléments d’une même ligne (ou d’une même colonne) de la matrice des  $C_{ij}$  (les valeurs infinies restant infinies). Après une telle opération, la valeur totale est augmentée ou diminuée de  $\lambda$ , mais la solution optimale est inchangée.

**Preuve.**  $\diamond$  Cela provient du fait que, dans la somme  $\sum_{i=1}^p C_{i\sigma(i)}$ , chaque ligne et chaque colonne interviennent une, et une seule, fois.  $\diamond$

### 7.2.2 Méthode pratique de résolution

On procède en trois phases.

#### 7.2.2.1 Première phase : obtention de valeurs nulles

A tous les éléments de chacune des colonnes on enlève le plus petit élément de cette colonne ; puis, dans la matrice ainsi obtenue, on enlève à tous les éléments d’une même ligne le plus petit élément de la ligne. On obtient de la sorte une matrice ayant au moins un zéro par ligne et par colonne qui sera notée  $(C_{ij}^{(1)})$ .

L’exemple de la figure 38 donne successivement les deux tableaux ci-après.

	1	2	3	4	5
A	1	0	4	5	3
B	0	∞	∞	6	0
C	0	2	0	3	∞
D	5	1	∞	9	8
E	∞	1	4	0	3

	1	2	3	4	5
A	1	0	4	5	3
B	0	∞	∞	6	0
C	0	2	0	3	∞
D	4	0	∞	8	7
E	∞	1	4	0	3

#### 7.2.2.2 Seconde phase : recherche directe d’une solution optimale

Avec les zéros de  $(C_{ij}^{(1)})$  on va chercher à former une solution de valeur zéro. Si cela est possible, on aura trouvé la solution optimale, sinon on procédera comme indiqué au paragraphe 7.2.2.3.

On considère d’abord une ligne ayant un nombre minimal de zéros (pour l’exemple, ce peut être la première). On encadre l’un des zéros de cette ligne (ici,  $C_{A2}^{(1)}$ ), puis on barre les zéros qui se trouvent sur la même ligne ou la même colonne que le zéro encadré (ici,  $C_{D2}^{(1)}$ ). On procède de même pour toutes les lignes, en tenant compte du travail déjà accompli. On obtient pour l’exemple de la figure 38 :

	1	2	3	4	5
A	1	0	4	5	3
B	0	∞	∞	6	∅
C	∅	2	0	3	∞
D	4	∅	∞	8	7
E	∞	1	4	0	3

Une affectation de valeur nulle est obtenue si chaque ligne contient un zéro encadré. On voit que dans l’exemple de la figure 38, ce n’est pas le cas.

#### 7.2.2.3 Troisième phase : recherche d’un ensemble minimal de lignes et colonnes contenant les zéros

On procède comme suit :

- a) on marque d’une astérisque toutes les lignes qui ne contiennent aucun zéro encadré (ligne D de l’exemple) ;
- b) on marque d’une astérisque toute colonne qui a un ou plusieurs zéros barrés dans une des lignes marquées (colonne 2 de l’exemple) ;
- c) on marque toute ligne qui a un zéro encadré dans une colonne marquée (ligne A de l’exemple) ;
- d) on répète ces trois marquages successifs jusqu’à épuisement (terminé pour l’exemple donné) ; on obtient ainsi un ensemble de lignes et ou de colonnes en nombre minimal contenant tous les zéros encadrés ou rayés ;
- e) on trace un trait sur toute ligne non marquée et toute colonne marquée (lignes B, C et D, colonne 2). Le tableau devient :

	1	2	3	4	5
A	1	0	4	5	3
B	0	∞	∞	6	∅
C	∅	2	0	3	∞
D	4	∅	∞	8	7
E	∞	1	4	0	3

f) on considère le tableau partiel des éléments non traversés par les traits tracés en e) et on prend le plus petit nombre de ce tableau partiel, nécessairement non nul ; on l’enlève aux colonnes non traversées par un trait et on l’ajoute aux lignes traversées par un trait.

L’exemple de la figure 38 donne le tableau suivant, que l’on note  $(C_{ij}^{(2)})$  :

	1	2	3	4	5
A	0	0	3	4	2
B	0	∞	∞	6	0
C	0	3	0	3	∞
D	3	0	∞	7	6
E	∞	2	4	0	3

Sur ce nouveau tableau, on recommence le cycle (§ 7.2.2.2). Si une solution optimale est obtenue, on arrête ; sinon on recommence et on définit les tableaux  $(C_{ij}^{(3)}, \dots, C_{ij}^{(p)})$ .

Sur le tableau  $(C_{ij}^{(2)})$ , obtenu dans l'exemple, on est amené à encadrer  $C_{D2}^{(2)}$  et barrer  $C_{A2}^{(2)}$ , encadrer  $C_{E4}^{(2)}$ , encadrer  $C_{A1}^{(2)}$  et barrer  $C_{B1}^{(2)}$  et  $C_{C1}^{(2)}$ , encadrer  $C_{B5}^{(2)}$ , encadrer  $C_{C3}^{(2)}$  :

	1	2	3	4	5
A	0	∞	3	4	2
B	∞	∞	∞	6	0
C	∞	3	0	3	∞
D	3	0	∞	7	6
E	∞	2	4	0	3

La solution optimale est alors atteinte grâce à

$$C_{A1}^{(2)} = C_{B5}^{(2)} = C_{C3}^{(2)} = C_{D2}^{(2)} = C_{E4}^{(2)} = 0$$

et à la permutation  $\sigma = \begin{bmatrix} A & B & C & D & E \\ 1 & 5 & 3 & 2 & 4 \end{bmatrix}$ . Si on revient aux données initiales on obtient :

$$C_{A1} + C_{B5} + C_{C3} + C_{D2} + C_{E4} = 7 + 7 + 1 + 4 + 2 = 21$$

On a tracé sur la figure 39 en traits forts le couplage à valeur optimale.

### 7.2.3 Théorème de König

Il découle de l'algorithme hongrois (§ 7.2.1 et § 7.2.2) : le nombre minimal de rangées (lignes et ou colonnes) contenant tous les zéros d'une matrice est égal au nombre maximal de zéros situés sur des lignes et des colonnes distinctes.

Ce théorème peut s'énoncer d'une manière différente en introduisant la notion de **support** d'un graphe  $G = (X, Y, \Gamma)$  constituant un couplage. Soit  $S$  un ensemble de points de  $X$  et/ou  $Y$  tel qu'un arc quelconque du graphe ait pour extrémité un point de  $S$  : c'est le support du graphe  $G$ . L'énoncé du théorème est le suivant.

**Théorème 30 (de König).** Dans tout graphe biparti  $G = (X, Y, \Gamma)$ , le nombre minimal de points d'un support est égal au nombre maximal d'arcs d'un couplage :

$$\text{Min}_{S \in S^*} |S| = \text{Max}_{C \in C^*} |C|$$

où  $S^*$  et  $C^*$  sont l'ensemble des supports et l'ensemble des couplages.

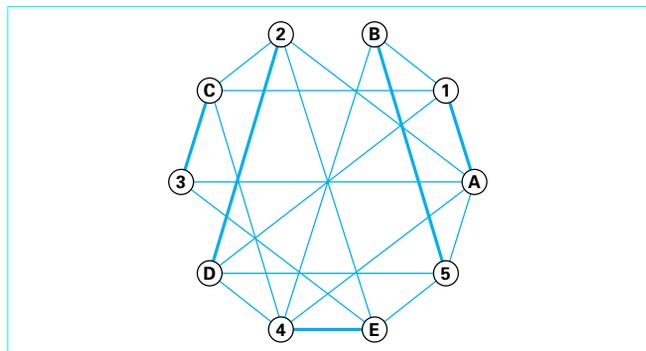


Figure 39 - Fin de l'algorithme hongrois et couplage optimal

## 8. Graphes planaires

### 8.1 Généralités

#### Définition 50.

On dit qu'un **graphe** ou un multigraphe est **planaire** si on peut le « dessiner » dans le plan, en matérialisant les arêtes par des arcs continus (des lignes brisées font aussi l'affaire) qui ne se croisent pas en dehors de leurs extrémités. Deux graphes que l'on peut amener à coïncider par déformation élastique du plan ne seront pas considérés comme distincts.

Les arêtes d'un  $p$ -graphe planaire  $G$  délimitent des surfaces planes, nommées **faces** de  $G$ , telles que deux points arbitraires choisis dans une même région peuvent toujours être reliés par un trait continu ne rencontrant ni sommets, ni arêtes.

La **frontière** d'une face de  $G$  est l'ensemble des arêtes qui touchent une face  $z$ .

Deux faces  $z$  et  $z'$  sont dites **adjacentes** si leurs frontières ont une arête commune.

On appelle **contour** de la face  $z$  celui des cycles élémentaires qui contient en son intérieur toutes les arêtes de la frontière.

Il y a toujours une face illimitée et une seule qui constitue la **face infinie**. Celle-ci n'admet évidemment pas de contour.

#### Exemple.

Un premier exemple pratique est schématisé par la figure 41 : on a trois villas **A, B, C**, que l'on veut relier par des conduites à des usines de production d'eau (**O**), de gaz (**G**), et d'électricité (**E**).

Le problème consiste à déterminer s'il est possible de placer, sur un plan, les trois villas et les conduites par rapport aux usines de sorte que deux conduites ne se croisent pas en dehors de leurs extrémités.

L'expérience montre que l'on peut toujours placer 8 conduites mais que la neuvième coupe toujours au moins une des huit premières ; le théorème de Kuratowski (§ 8.2) le confirme.

Un exemple important de graphe non planaire est le graphe complet  $K_n$  à  $n$  sommets pour  $n \geq 5$  (voir figure 3).

Dans les deux cas mentionnés, c'est le théorème de Jordan sur les arcs fermés qui est la clé de la réponse négative : un arc fermé découpe le plan en deux parties ouvertes connexes.

On a des exemples de graphes planaires avec les faces ou couches d'un circuit électronique imprimé. C'est précisément pour contourner les limitations de planarité que les circuits multicouches ont été inventés.

Enfin, une carte de géographie politique peut être modélisée par un graphe planaire dont les sommets sont les régions et dont les arêtes sont les paires de régions voisines. On peut aussi tenir compte des « enclaves » mais le problème obtenu est plus délicat et nous l'omettrons.

#### 8.1.1 Notion de dualité

**Définition 51.** Soit un  $p$ -graphe planaire connexe  $G$ . On construit le **dual**  $G^*$  de  $G$  de la façon suivante : à l'intérieur de toute face  $z$  de  $G$  on place un sommet  $x_z^*$  de  $G^*$ . A toute arête  $\bar{u}_k$  de  $G$  on fait correspondre une arête  $\bar{v}_m$  de  $G^*$  reliant les sommets  $x_z^*$  et  $x_s^*$  correspondant aux faces  $z$  et  $s$  se trouvant de part et d'autre de l'arête  $\bar{u}_k$ .

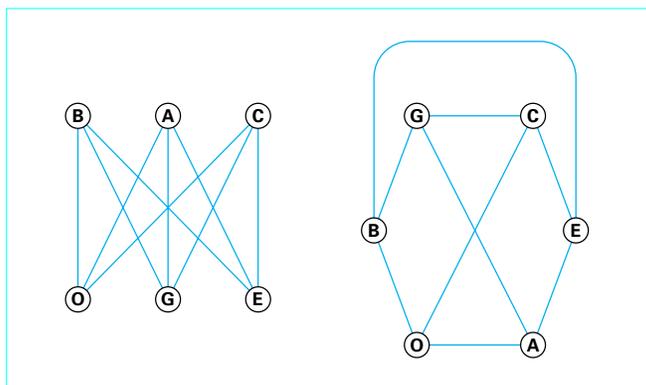


Figure 40 – Graphe non planaire  $K_{3,3}$

L'interprétation géographique permet de rendre cette définition plus parlante : on assimile  $G$  à une carte de géographie délimitant des pays par leurs frontières, sur une île entourée d'un océan. On place alors un sommet de  $G^*$  sur chaque capitale de chaque pays et un dans l'océan ; on crée une arête joignant deux sommets de  $G^*$  si, et seulement si, les deux pays correspondants sont limitrophes (c'est le trajet d'une route de capitale à capitale).

La figure 41 donne un exemple d'un graphe planaire avec son dual.

**Proposition 13.**

Soit un  $p$ -graphe planaire connexe  $G$  et  $G^*$  son dual. Alors :

- a)  $G$  et  $G^*$  ont des nombres cyclomatiques égaux ;
- b) si  $G$  admet un point d'articulation,  $G^*$  en admet aussi un.

**8.1.2 Quelques résultats théoriques**

■ **Faces et cycles**

Dans un  $p$ -graphe planaire  $G$ , les contours des diverses faces finies constituent une base fondamentale de cycles indépendants (§ 5.2.8). On peut utiliser cette propriété pour construire très facilement un repère de courants de mailles, linéairement indépendants dans un réseau électrique, constituant un  $p$ -graphe planaire.

■ **Formule d'Euler**

Soit un  $p$ -graphe planaire possédant  $n$  sommets,  $m$  arêtes et  $f$  faces ; on a :

$$n - m + f = 2$$

■ **A propos de la forme des faces**

Dans tout 1-graphe planaire connexe, il y a un sommet dont le degré est inférieur ou égal à 5. Par dualité, tout 1-graphe planaire connexe a au moins une face dont le contour est formé de 3, 4 ou 5 arêtes. Ainsi, dans une carte de géographie il y a au moins un pays ayant une forme de « triangle », « quadrilatère » ou « pentagone ».

**8.2 Test de planarité d'un graphe**

Étant donné un graphe connexe, la question est de savoir s'il est planaire ou non. Ce problème se pose chaque fois que l'on désire implanter un circuit électronique sur une face de circuit imprimé.

Le graphe  $K_{3,3}$  des villas et des usines (figure 40) et le graphe complet à 5 sommets (figure 3), dont nous admettrons qu'ils ne sont pas planaires, permettent de définir toute une famille de

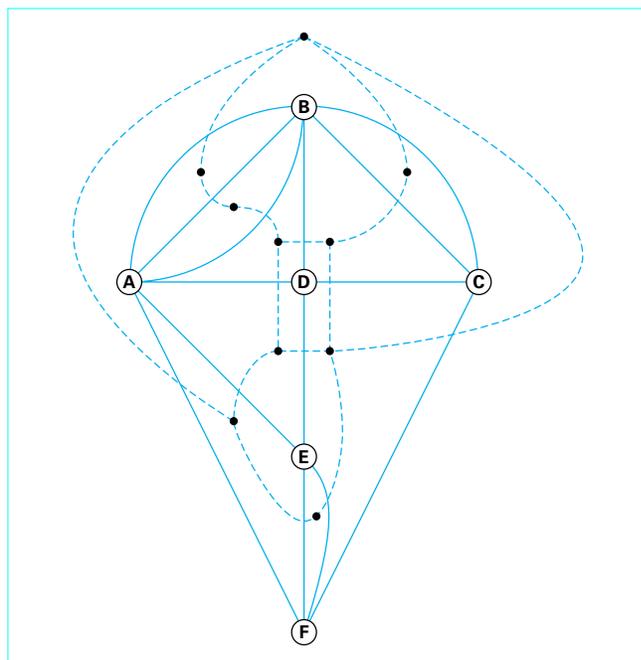


Figure 41 – Graphe et son dual planaire

graphes non planaires ; il suffit en effet de placer, sur chaque arête, autant de sommets de degré 2 que l'on veut pour définir d'autres graphes non planaires de mêmes types. Si l'on ajoute encore d'autres sommets et d'autres arêtes, le graphe obtenu n'est certainement pas planaire.

Inversement, définissons une opération simplificatrice pour l'étude de la planarité.

**Définition 52.** Soit un graphe connexe  $G$ . On dit qu'un graphe  $H$  se déduit de  $G$  par **contraction** s'il a été obtenu par application répétée des règles suivantes :

- (i)  $x$  étant un sommet de degré 1 de  $G$ ,  $H$  est le sous-graphe de  $G$  obtenu en retirant  $x$  ;
- (ii)  $x$  étant un sommet de degré 2 de  $G$ , de voisins  $y$  et  $z$ ,  $H$  est le sous-graphe de  $G$  obtenu en retirant  $x$ , auquel on ajoute une arête  $(y, z)$ .

C'est ainsi que la figure 42 représente un graphe non planaire. En effet, on peut le contracter en supprimant d'abord le sommet I, de degré 1, puis H, de degré 2. La suppression de ce dernier oblige à créer une arête (C,F).

On supprime de même G et J, ce qui finit par créer une arête (E,B), et on trouve le graphe des trois maisons, isomorphe au graphe  $K_{3,3}$  (figure 40).

La dualité entre 1-graphes planaires (§ 1.1.2) n'est parfaite, au sens où  $(G^*)^* = G$ , que si le graphe  $G$  n'a pas de sommet de degré 1 ou 2, c'est-à-dire s'il ne peut pas être contracté.

On peut alors montrer le résultat suivant.

**Théorème 31 (de Kuratowski).** La condition nécessaire et suffisante pour qu'un  $p$ -graphe  $G$  soit planaire est qu'il n'admette pas de sous-graphe se contractant en un graphe isomorphe à  $K_5$  ou à  $K_{3,3}$ .

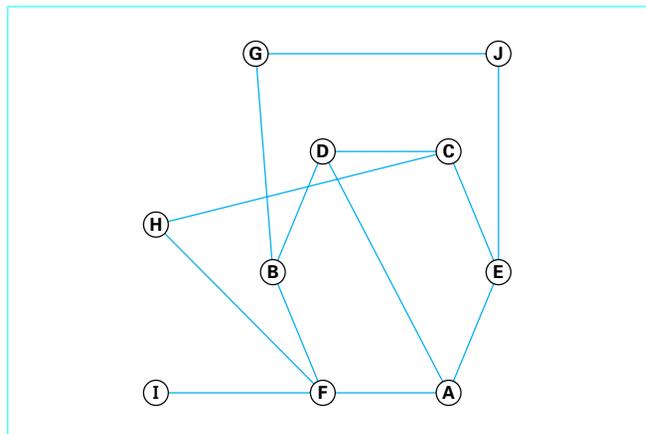


Figure 42 – Graphe non planaire

### 8.3 Coloriage des graphes planaires

Considérons un graphe  $G$  planaire et connexe, et interprétons-le comme une carte de géographie. On souhaite colorier la carte (les pays) de sorte que deux pays limitrophes n'aient jamais la même couleur.

Si on attache les couleurs des pays à leurs capitales, on voit qu'il s'agit en fait de colorier les sommets du graphe dual de sorte que deux sommets adjacents ne soient pas de la même couleur.

Quelques remarques permettent de limiter un peu la recherche d'un coloriage :

- les sommets de degré 1 n'influent aucunement (ils ne créent pas de face) ;
- les sommets de degré 2 peuvent être retirés par le processus de contraction du graphe, sans modifier le graphe dual ;
- les sommets de degré supérieur à 3 peuvent être aussi éliminés, car on peut les remplacer par de petits polygones (figure 43).

Il est alors clair que si on sait colorier tout graphe planaire régulier de degré 3 (définition 8) avec  $N$  couleurs, alors on sait colorier le graphe proposé initialement.

On démontre assez facilement, mais non sans technique, le résultat suivant (voir [41], page 115).

**Théorème 32 (des cinq couleurs).** Tout graphe planaire a un nombre chromatique inférieur ou égal à 5.

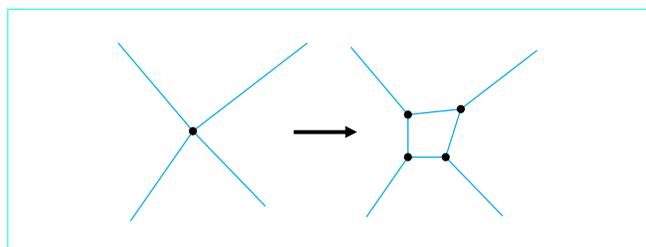


Figure 43 – Réduction au degré 3

En fait, les cartographes savent depuis longtemps que quatre couleurs suffisent pour une telle tâche. La première tentative de preuve est due à Kempe il y a un siècle, mais cette preuve s'est avérée incomplète. En 1976, Appel et Haken sont parvenus à une « démonstration » au moyen d'un ordinateur pour la vérification de nombreux cas. Plus récemment, un travail dans le même sens, mais moins gigantesque, a été accompli par Robertson, Seymour, Thomas et Sanders. La validité de la preuve reste à ce jour (1999) discutée. Un outil important pour ces recherches théoriques est le résultat suivant.

**Théorème 33 (de Heawood).** Si  $G$  est un graphe planaire homogène de degré 3, sans isthmes, les trois conditions suivantes sont équivalentes :

- on peut colorier les faces de  $G$  avec 4 couleurs sans que 2 faces voisines aient la même couleur ;
- on peut colorier les arêtes de  $G$  avec 3 couleurs sans que 2 arêtes ayant une extrémité commune aient la même couleur ;
- on peut affecter à chaque sommet  $x_i$  un coefficient  $p(x_i)$  égal à  $+1$  ou à  $-1$  de sorte que le contour  $\mu$  de chaque face vérifie :

$$\sum_{x \in \mu} p(x) \equiv 0 \pmod 3.$$

Le lecteur pourra lire l'article D61, *Réseaux électriques, lois de Kirchhoff, nœuds et mailles* (§ 5.3.3), au sujet de la dualité dans les réseaux électriques maillés linéaires.

## 9. Implantations d'un graphe dans un autre

### ■ Principe des implantations

**Exemple :**

Soit les deux graphes  $G_1 = (X, \Gamma)$  et  $G_2 = (Y, \Delta)$  représentés sur la figure 44, avec

$$X = \{A, B, C, D, E\},$$

$$Y = \{1, 2, 3, 4\}.$$

On se demande si les relations d'adjacence (ou d'incidence) propres à  $G_2$  pourraient être vues comme des « cas particuliers » de relations de  $G_1$ . Il s'agit donc d'associer les sommets de  $G_2$  à certains sommets de  $G_1$ , de sorte que tout arc liant deux sommets de  $G_2$  corresponde à au moins un arc liant les sommets homologues de  $G_1$ .

On constate que c'est possible si on associe **1** à **D**, **2** à **A**, **3** à **B** et **4** à **E** (il faut vérifier que les relations d'adjacence sont respectées, y compris les boucles).

Mais ce n'est pas la seule solution ; par exemple, on aurait pu aussi bien associer **1** à **D**, **2** à **E**, **3** à **D** et **4** à **B**.

De fait,  $G_1$  contient plus d'arcs entre ses sommets que ceux qui sont requis pour « implanter »  $G_2$  mais on peut les ignorer.

### ■ Cas général

**Définition 53.** Étant donné deux graphes  $G_1 = (X, \Gamma)$  et  $G_2 = (Y, \Delta)$ , tels que

$$|Y| \leq |X|,$$

une **implantation** de  $G_2$  dans  $G_1$  est un isomorphisme entre  $G_2$  et un graphe partiel de  $G_1$ , c'est-à-dire une application  $\varphi : Y \rightarrow X$  injective et telle que si  $b \in \Delta a$  alors

$$\varphi(b) \in \Gamma \varphi(a).$$

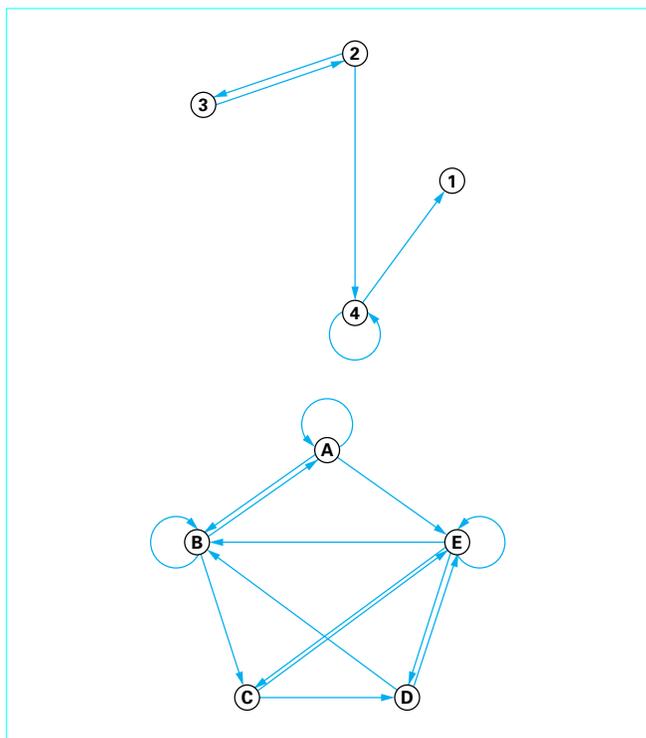


Figure 44 - Problème d'implantation

On trouve des problèmes pratiques d'implantation à propos de l'implantation rationnelle de machines dans un atelier, de services dans une entreprise, de câblage des machines électroniques, etc.

#### Recherche des implantations

La recherche systématique des implantations est, en général, un problème difficile, voire inextricable, en raison de sa complexité

(c'est sans doute un problème NP-complet). Si les graphes considérés ont un nombre de sommets restreint, on peut obtenir les implantations simplement par énumération des combinaisons possibles. Même le dénombrement des implantations n'est pas un problème simple puisque, dans le cas de deux graphes ayant autant de sommets, le dénombrement des implantations de l'un dans l'autre permet de savoir s'ils sont isomorphes.

On risque pourtant de devoir énumérer toutes les implantations si une notion de coût intervient, dans la recherche d'une implantation de coût minimal par exemple.

Un procédé d'élimination est parfois possible, en se basant sur les caractéristiques topologiques des deux graphes proposés.

**Exemple :** on reprend la figure 44 ; l'élimination est facile en commençant la recherche des sommets pouvant correspondre au sommet 4 (pourvu d'une boucle).

Si  $\varphi(4) = A$ , le seul arc entrant vers A venant de B, on doit avoir  $\varphi(2) = B$ . Ce cas est exclu car les seuls arcs symétriques entre B et un autre sommet vont vers A ; on devrait alors poser

$$\varphi(3) = A$$

et  $\varphi$  ne serait pas injective.

Si  $\varphi(4) = B$ , il faut encore trouver un sommet dans  $G_1$  lié à B et aussi source/extrémité d'une paire d'arcs symétriques. Deux choix apparaissent avec D et E. Si  $\varphi(2) = D$ , on doit poser

$$\varphi(3) = E$$

et

$$\varphi(1) = A \text{ ou } \varphi(1) = C \text{ (2 solutions).}$$

Si  $\varphi(4) = E$ , la même méthode conduit à poser

$$\varphi(2) = A,$$

$$\varphi(3) = B$$

et

$$\varphi(1) = C \text{ ou } \varphi(1) = D \text{ (2 autres solutions).}$$

Au total, il y a 4 implantations possibles de  $G_2$  vers  $G_1$ .

Un tel procédé d'élimination devient impossible dès que le nombre de sommets augmente un peu.