

**MINISTÈRE DE L'ENSEIGNEMENT SUPÉRIEUR ET DE LA
RECHERCHE SCIENTIFIQUE UNIVERSITÉ Abderrahmane Mira -**

Béjaia

FACULTÉ DES SCIENCES EXACTES

Département de Mathématiques

**Mémoire présenté pour l'obtention du diplôme de Licence
en mathématiques**

Option : Statistique et analyse décisionnelle

Par

Bouamara M'hand

THÈME

Optimisation dans les graphes probabilistes

Soutenu publiquement, le 30/06/2016 devant le jury composé de:

Mr. Président.

Mme. Promoteur.

Mme.

Promotion :2015/2016

Remerciment

Remerciements

On dit souvent que le trajet est aussi important que la destination. Ces cinq années de formation nous ont permis de bien comprendre la signification de cette phrase toute simple. Ce parcours, en effet, ne s'est pas réalisé sans défis et sans soulever de nombreuses questions pour lesquelles les réponses nécessitent de longues heures de travail.

En premier lieu, nous remercions **Dieu**, le tout puissant pour ses faveurs et ses grâces, de nous avoir donné le courage et la patience pour avoir mené ce travail durant cette année.

Nous remercions également l'honorable jury pour avoir consenti à évaluer et à juger notre travail.

De plus, nous exprimons notre gratitude à Monsieur K. Kabyl qui nous a accordé toute l'assistance nécessaire à l'élaboration de ce mémoire. Sa bienveillance nous a accompagné durant toute notre formation. C'est donc avec reconnaissance que nous lui présentons nos remerciements. Nous le remercions aussi pour son aide précieuse et pour sa contribution à la réalisation de ce présent travail.

Nos remerciements sont aussi adressés à Monsieur x et Monsieur y qui nous font l'honneur de juger notre travail.

Nous remercions tous les enseignants du département de Mathématiques qui ont assuré notre formation universitaire.

Dédicaces

je dédie ce modeste travail à tous ceux et celles qui m'ont aidé de proche et de loin,notamment:

Mes parents,qui ne m'ont jamais laissé tomber dans toutes les circonstances.

Mes sœurs que j'aime beaucoup.

Mes cousines,hommes et femmes qu'ils soient.

Tous mes amis qui m'ont toujours soutenu.

Tous mes enseignants qui ont contribué à ma formation.

Toute la famille BOUAMARA.

Tous ceux qui ont le mérite pour que ce travail soit réalisé.

Bouamara m/hand

Table des matières

1	Notations et notions de bases	3
1.1	Graphes	3
1.1.1	Définitions et terminologies	4
1.1.2	Graphes non orientés	4
1.1.3	Graphe orientés	5
1.2	Chemins, chaînes, cycles et circuits dans les graphes	10
1.3	Connexité et forte connexité dans les graphes	11
1.3.1	Connexité	11
1.4	Distance et diamètre dans les graphes	15
1.5	Opération sur les graphes	16
1.5.1	Somme cartésienne de deux graphes	16
1.5.2	Produit Cartésien de deux graphes	17
1.5.3	Morphismes de graphes	17
1.5.4	Isomorphisme de graphes	18
1.6	Représentations matricielles	18
1.6.1	Matrice d'adjacence	18
1.6.2	Matrice d'incidence	19
1.7	Fermeture transitive d'un graphe	20
1.8	Quelque types de graphes	22
1.8.1	graphe valué	22
1.8.2	Graphe complet	23
1.8.3	Graphe planaire	24

1.8.4	Graphe biparti	25
1.8.5	Hypercube	26
1.8.6	Arbre	26
1.8.7	Polyarbre	27
1.8.8	Arborescence	27
1.8.9	Graphes triangulés	28
1.8.10	Graphes pondérés	29
1.8.11	Graphe étiqueté	29
2	Modèles graphiques probabilistes	31
2.1	Introduction	31
2.2	Notions de probabilités	32
2.2.1	Définitions principales	32
2.2.2	Propriétés générales d'une probabilité	33
2.2.3	Probabilités sur plusieurs variables	33
2.2.4	Probabilités conditionnelles, Indépendance	34
2.2.5	Quelques règles de base en probabilité	35
2.3	Les graphes probabilistes	37
2.4	Matrice de transition	38
2.4.1	Etat probabiliste	38
2.4.2	Etat stable	39
2.5	Quelques Modèles graphiques probabilistes.	40
2.5.1	Modèle Graphique Probabiliste	40
2.5.2	Réseaux bayésiens	41
2.5.3	Chaînes de Markov	43
2.5.4	Lien entre chaînes de Markov et théorie des graphes	45
3	Problèmes d'optimisation pour les graphes valués	48
3.1	Problème de l'arbre couvrant de poids minimum	48
3.1.1	Algorithme de Kruskal	49

3.1.2	Algorithme de PRIM	50
3.2	Problème du plus court chemin	54
3.3	Problème d'affectation	55
3.3.1	La résolution d'un problème d'affectation par la méthode hongroise :	55
3.3.2	Algorithme de recherche d'une affectation minimale	56
3.3.3	Algorithme de recherche d'une affectation maximale	58
3.4	Problème d'ordonnancement	58
3.4.1	Le diagramme de Gantt	59
3.4.2	Méthode française MPM (Méthode Potentiel Métra)	60
3.4.3	Méthode américaine: PERT	61
4	Applications	62
4.1	Réseaux téléphoniques	62
4.1.1	Présentation de la problématique	62
4.1.2	Modélisation du problème sous forme d'un réseaux	62
4.1.3	Problème de plus long chemin	76

Introduction

L'histoire de la théorie des graphes débute avec les travaux d'Euler au 18^e siècle et trouve son origine dans l'étude de certains problèmes, tels que celui des ponts de Königsberg, la marche du cavalier sur l'échiquier ou le problème du coloriage de cartes et du plus court chemin entre deux points.

C'est plus récemment en 1822 que le mot **graphes** est introduit par le mathématicien et géomètre anglais James Joseph Sylvester. La théorie des graphes s'est alors développée dans diverses disciplines telles que la chimie, la biologie, les sciences sociales, l'informatique...

Depuis le début du 20^e siècle, elle constitue une branche à part entière des mathématiques, grâce aux travaux de König, Menger, Cayley, Berge et Erdős. De manière générale, un graphe permet de représenter des objets ainsi que les relations entre ses éléments (par exemple réseau de communication, réseaux routiers, interaction de diverses espèces animales, circuits électriques...)

En mathématiques, on retrouve les graphes dans la combinatoire, la théorie des ensembles, l'algèbre linéaire, la théorie des polyèdres, la théorie des jeux, l'algorithmique, les probabilités...

Les derniers travaux en théorie des graphes sont souvent effectués par des informaticiens, du fait de l'importance que revêt l'aspect algorithmique. L'optimisation combinatoire occupe une place très importante en recherche opérationnelle, en mathématiques discrètes et en informatique car il consiste à rechercher la meilleure solution optimale d'un critère au une fonction donnée. Son importance justifie d'une part par la grande difficulté des problèmes d'optimisation et d'autre part par de nombreuses applications pratiques

pouvant être formulée sous la forme d'un problème combinatoire. bien que les problèmes d'optimisation combinatoires soient souvent faciles à définir.

Il existe plusieurs types de graphes, dans notre travail nous allons étudier les graphes probabilistes qui sont des graphes orientés et pondérés, dans lequel chacune des arêtes est représentée par un nombre noté (C_{ij}) bornée entre 0 et 1.

L'étude présentée dans ce document porte essentiellement sur l'optimisation dans les graphes probabilistes.

Ce mémoire est constitué d'une introduction générale, de quatre chapitres, d'une conclusion et d'une bibliographie.

Le présent mémoire est organisé comme suit:

- Dans le premier chapitre, nous allons rappeler brièvement quelques notions sur la théorie des graphes et certaines définitions essentielles.
- Dans le deuxième chapitre, en introduisant des concepts de chaîne de Markov et le lien avec la théorie des graphes, comme nous décrivons précisément les différents modèles des graphes probabilistes.
- Dans le troisième chapitre en présentant quelques problèmes d'optimisation et techniques de résolutions pour chaque problème.
- Enfin dans le quatrième chapitre traite quelques problèmes réels avec une modélisation.

Notations et notions de bases

Ce chapitre a pour but de présenter les différentes notions relatives aux graphes, qui seront largement utilisées dans les chapitres qui suivent. Les graphes sont utilisés pour représenter les différents modèles dits graphiques (les réseaux bayésiens, les réseaux de contraintes par exemple). Généralement les sommets de ces graphes seront associés aux variables des modèles et les liens reliant les sommets seront associés aux différentes fonctions (tables de probabilités conditionnelles ou de transitions, fonctions de coûts, contraintes . . .) présentes dans les modèles. La théorie développée sur les graphes est importante, ici nous n'en présentons qu'une petite partie, elle est utilisée dans certains algorithmes des modèles graphiques pour les simplifier, les décomposer, établir des propriétés structurelles sur ces modèles.

1.1 Graphes

C'est en 1822 que le mot **graphe** est introduit par l'Anglais J.J.Sylvester et en 1936 qui paraît le premier livre sur la théorie des graphes, écrit par D.Knig .Un graphe $G = (V, E)$ est un dessin géométrique défini par la donnée d'un ensemble de points (appelés sommets ou noeuds), reliés entre eux par un ensemble de lignes ou de flèches (appelées arêtes ou arcs) chaque arêtes a pour extrémités deux points, éventuellement confondus.Des graphes

permettent la modélisation des relations entre des objets. Formellement, il n'y a pas de contraintes sur l'ensemble des arêtes. Il est notamment possible que E contiennent des arêtes multiples entre deux sommets (et donc identiques) ainsi que des boucles ($e = (v, v)$) qui relient un sommet v à lui-même.

La taille d'un graphe n est définie par le nombre de ses sommets. On écrit également $|V|$ et on note $\|E\|$ le nombre d'arêtes, chaque arête peut être décrite par les sommets qu'il joint. On dit que le sommet v_1 et v_2 sont voisins si $e = (v_1, v_2)$. Le nombre total d'arête incidentes à un sommet est dénommé son degré. Il existe des graphes orientés et non orientés. Dans les graphes orientés les arêtes ont une direction et sont dénommées arcs. À la différence des graphes non orientés, l'arc (v_1, v_2) n'est pas équivalent à l'arc (v_2, v_1) .

Exemple 1.1.1 Le graphe associé à l'ensemble des arêtes $A = \{(2, 1); (2, 2); (3, 1); (3, 2); (3, 3); (3, 4); (4, 1); (4, 2); (4, 3)\}$, et montré dans la figure 1.1.

Ce graphe est d'ordre $n = 4$ et de taille $m = 9$.

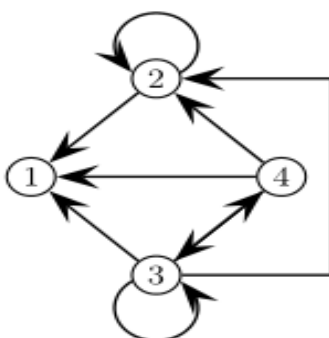


Figure 1.1: Graphe

1.1.1 Définitions et terminologies

1.1.2 Graphes non orientés

Un graphe simple G est un couple formé de deux ensembles : un ensemble $V = (v_1, v_2, \dots, v_n)$ dont les éléments sont appelés sommets, et un ensemble $E = (e_1, e_2, \dots, e_n)$ partie de

l'ensemble $P_2(V)$ des parties à deux éléments de V , dont les éléments sont appelés arêtes. On notera $G = (V; E)$. Une arête e de l'ensemble E est définie par une paire non-ordonnée de sommets, appelés les extrémités de e . Si l'arête e relie les sommets v_1 et v_2 , on dira que ces sommets sont adjacents, ou incidents avec e , ou encore que l'arête e est incidente avec les sommets v_1 et v_2 .

Exemple 1.1.2 La figure 1.2 montre un graphe simple à 4 sommets .

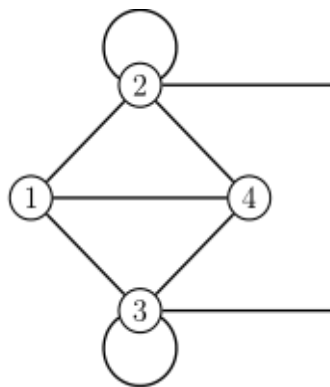


Figure 1.2 :Graphe non orienté

1.1.3 Graphe orientés

On dit que G est un graphe orienté s'il y a une distinction entre les liens (u, v) et (v, u) , c'est-à-dire $(u, v) \neq (v, u)$. Dans ce cas le lien est appelé un arc. On représente $e = (u, v)$ graphiquement par une flèche qui part de u pour joindre v qui sera la pointe de cette flèche. Dans ce cas, v sera appelé un successeur de u (ou u est un prédécesseur de v) et chaque sommet peut avoir plusieurs successeurs et plusieurs prédécesseurs. On appelle une boucle tout arc (u, u) dont ses extrémités se Coïncident.

Exemple 1.1.3 La figure 1.3 montre un graphe orienté à 4 sommets.

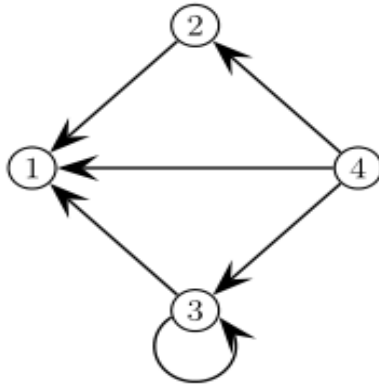


Figure 1.3: Graphe orienté

Sous-graphes et graphes partiels

Soit $G = (V, E)$ un graphe (orienté ou pas) alors un graphe partiel de G est un graphe G' ayant pour sommets tous les sommets de G et pour arcs/arêtes seulement un sous-ensemble de A , ce qui s'écrit :

$$G' = (V, E') \quad \text{avec} \quad E' \subset E$$

Un sous-graphe de G est un graphe G' ayant pour sommets un sous-ensemble S' des sommets de G et en ne conservant que les arcs/arêtes joignant les sommets de S' ce qui s'écrit :

$$G' = (S', A') \quad \text{avec} \quad S' \subset S \quad \text{et} \quad A' = \{(x, y) \in A \mid x \in S' \quad \text{et} \quad y \in S'\}$$

Exemple 1.1.4 *Le graphe partiel de G induit par $A' = A \setminus \{(2, 2); (3, 2); (4, 3)\}$, et donné par la figure 1.4.*

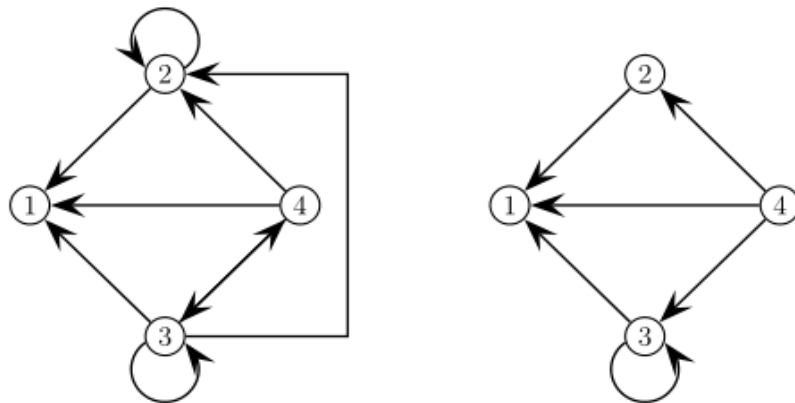


Figure 1.4: Graphe partiel

Le sous-graphe de G induit par $S' = \{1; 2; 4\}$, en supprimant le sommet 3 on supprime les arcs $\{(3, 1); (3, 2); (3, 3); (3, 4); (4, 3)\}$, et montre dans la figure 1.5.

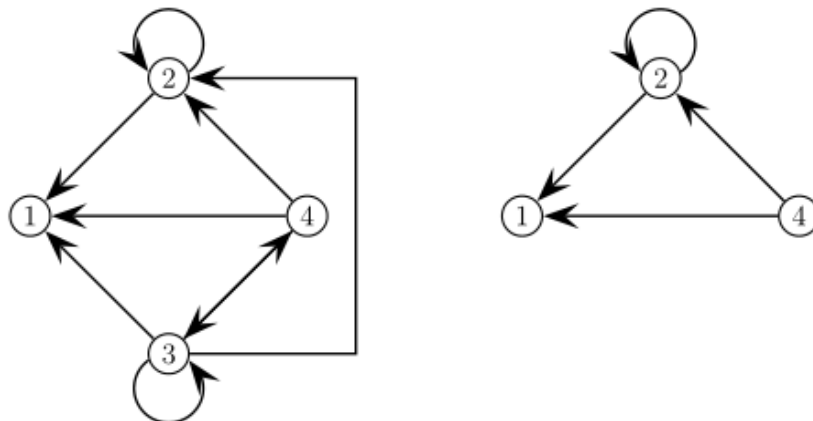


Figure 1.5: Sous-graphe.

Définition 1.1.1 Graphe complémentaire

Soit $G = (V, E)$ un graphe simple, on peut définir un graphe complémentaire $\bar{G} = (V, \bar{E})$ comme suit :

$$u \in \bar{E} \iff u \notin E$$

C'est à dire:une arc (arête) appartient au graphe complémentaire(\bar{G}) si il n'appartient pas au graphe initial G .

Degré d'un graphe

Soit $G = (V, E)$ un graphe orienté on a :

- Le demi -degré extérieur d'un sommet x est égale au nombre d'arcs ayant le sommet x comme extrémité initiale, on dit aussi le nombre d'arcs incidents extérieurs au sommet x . On le note :

$$d_G^+(x) = |\{e \in E / I(e) = x\}|$$

- Le demi-degré intérieur d'un sommet x est égal au nombre d'arcs ayant le sommet x comme extrémité terminale, on dit aussi le nombre d'arcs incidents intérieurs au sommet x . On le note:

$$d_G^-(x) = |\{e \in E / T(e) = x\}|$$

- Le degré d'un sommet x est le nombre d'arcs ayant x comme extrémité initiale ou terminale, on dit aussi le nombre d'arcs adjacents à x . On le note

$$d_G(x) = d_G^+(x) + d_G^-(x)$$

Théorème 1.1.1 (Lemme des poignées de mains)

Soit $G = (S, A)$ un graphe de taille m alors les sommes des degrés entrants et sortants des sommets de G sont égales au nombre d'arcs

$$\sum_{x \in S} d^+(x) = \sum_{x \in S} d^-(x) = m$$

En conséquence la somme des degrés est égale au double de nombre d'arcs

$$\sum_{x \in S} d(x) = \sum_{x \in S} d^+(x) + \sum_{x \in S} d^-(x) = 2m$$

Preuve. chaque arc(x, y) compte deux fois dans la somme des degrés : une fois dans $d^+(x)$ et une fois dans $d^-(y)$, d'où le résultat:

$$\sum_{x \in S} d^+(x) = \sum_{x \in S} d^-(x) = \sum_{i=1}^m 1 \implies \sum_{x \in S} d(x) = \sum_{x \in S} d^+(x) + \sum_{x \in S} d^-(x) = 2m. \blacksquare$$

• Voisinage d'un Sommet

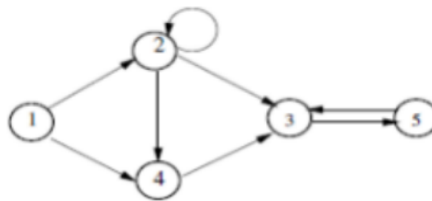
Si $uv \in E$, on dit que u est voisin de v et vice versa. On note $N(u)$ l'ensemble de tous les voisins de u :

$$N(u) = \{v \in X / uv \in E\}$$

De même on définit un voisinage fermé comme suit :

$$N^{\bar{}}(u) = \{\{v \in X / uv \in E\} \cup \{u\}\}$$

Dans le cas orienté on distingue entre l'ensemble des successeurs et l'ensemble des prédécesseurs qui sont donnés comme suit:



Sommet	1	2	3	4	5
Les successeurs $W^+(x)$	2 et 4	4 et 3	5	3	3
Les prédécesseurs $W^-(x)$		2 et 1	2 ; 4 et 5	1 et 2	3

Relations d'Adjacence

Dans un graphe G , une arête e reliant un sommet u à un sommet v est notée par UV et dans ce cas on dit :

- u et v sont adjacents;
- u et v sont les extrémités de e , e incidente à u et v . Deux arêtes sont dites adjacentes si elles sont incidentes à un même sommet.

1.2 Chemins, chaînes, cycles et circuits dans les graphes

La majeure partie des problèmes modélisés en théorie des graphes repose sur la notion de chemin. Cette notion est tout à fait intuitive, mais nous allons lui donner un sens mathématique très précis.

Soit $G = (S, A)$ un graphe orienté (resp, non orienté), un chemin (resp, une chaîne) dans G est une liste de sommets $C = (x_0, x_1, \dots, x_k)$ telle qu'il existe une arête (resp. une arc) entre chaque couple de sommets successifs de C . Ce qui s'écrit:

- si $G = (S, A)$ est orienté alors $\forall i = 0, 1, \dots, k-1, (x_i, x_{i+1}) \in A$
- si $G = (S, A)$ est non-orienté alors $\forall i = 0, 1, \dots, k-1, \{x_i, x_{i+1}\} \in A$

On appellera

- **Chaîne** le nombre d'arcs/arêtes du chemin.
- **(chemin/chaîne) simple** un chemin/chaîne dont tous les arcs/arêtes sont différents.
- **(chemin/chaîne) élémentaire** un chemin/chaîne dont tous les sommets sont différents sauf peut être le départ et l'arrivée (pour autoriser les circuits/cycles).
- **circuit** dans un graphe orienté un chemin simple finissant à son point de départ.
- **cycle** dans un graphe non-orienté un chemin simple finissant à son point de départ.

Théorème 1.2.1 (*lemme de Köning*)

Soient u et v deux sommets distincts d'un graphe G . S'il existe un chemin de G reliant u à v alors il existe un chemin élémentaire de u à v .

Preuve. On considère un chemin de u à v qui n'est pas élémentaire, il passe donc deux fois par un même sommet S , on est donc dans la situation suivante:

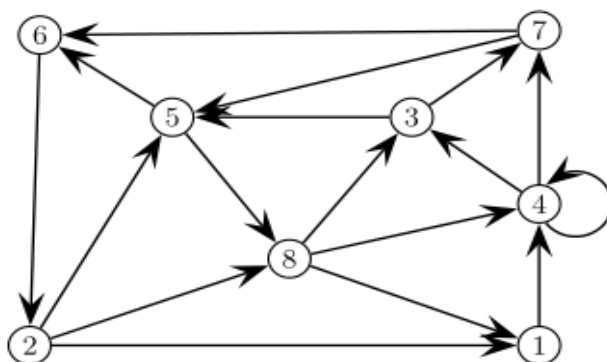
$$C = \left(x, x_1, x_2, \dots, \underbrace{S_1, \dots, S}_{\text{à supprimer}}, x_{n-1}, y \right)$$

la partie qu'on a supprimé est un cycle/circuit. Ensuite il suffit de recommencer tant que le chemin n'est pas élémentaire. ■

Remarque 1.2.1 *Les circuits les plus simples sont les boucles et les doubles flèches:*

- Une boucle (x, x) est le plus court circuit possible.
- Une double flèche correspondant à deux arcs (x, y) et (y, x) donne aussi un circuit (x, y, x) .
- Le chemin $C = (x)$ est de longueur nulle. Il ne correspond donc à aucun arc (c'est cohérent puisque la boucle $(x, x) \neq C$) on parle alors plutôt de chemin **nul**.

Exemple 1.2.1 Soit $G = (S, A)$ un graphe



- $(8, 4, 4)$: chemin simple mais pas élémentaire, longueur 2.
- $(1; 4; 3; 5; 8; 1)$: circuit élémentaire et simple, longueur 5.
- $(1; 4; 7; 6; 2)$: chemin élémentaire et simple, longueur 4.
- $(7; 6; 2; 1; 4; 3; 5; 8; 1; 4; 7)$: circuit ni simple ni élémentaire, longueur 10.

1.3 Connexité et forte connexité dans les graphes

1.3.1 Connexité

Soit $G = (S, A)$ un graphe tel que pour tout couple de sommets (x, y) il existe un chemin de x vers y :

$$\forall x, y \in S, \exists x_i \in S, i = 0, \dots, k, c = (x_0, \dots, x_k) \text{ chemin } G \text{ et } [x_0 = x \text{ et } x_k = y]$$

alors on dira que

- G est connexe si G est non-orienté.
- G est fortement connexe si G est orienté.

- On appelle aussi composante (fortement) connexe un sous-graphe de G de taille maximale qui est (fortement) connexe.



Figure 3.1: Graphe ayant trois composantes connexes

- Un graphe $G = (X, E)$ est dite connexe si tous ses sommets ont deux à deux la relation de connexité. Autrement dit, si G contient une seule composante connexe.

Un graphe est connexe \Leftrightarrow il possède une seule composante connexe.

Remarque 1.3.1 *Pour trouver les composantes connexes d'un graphe est assez facile mais pour trouver les composantes fortement connexes est plus difficile. On pourra utiliser l'algorithme suivant :*

Algorithme de recherche composante fortement connexe (cfc) :

Soit G un graphe orienté, un moyen de vérifier si ce graphe est fortement connexe est d'appliquer un algorithme de marquage suivant, qui permet de déterminer toutes ses composantes fortement connexe.

- **Le principe:**

L'idée de cet algorithme est de parcourir le graphe à partir d'un sommet x dans le sens direct (en suivant les flèches des arcs) et dans le sens opposé (en suivant les flèches des arcs en sens inverse), le premier ensemble obtenu regroupe les sommets accessibles à partir de x , le deuxième regroupe les sommets qui peuvent atteindre x . L'intersection de ces deux ensembles forment l'ensemble des sommets qui à la fois peuvent atteindre x et sont accessible à partir de x . On obtient ainsi la composante fortement connexe qui contient le sommet x .

• **Énoncé:**

Données: un graphe orienté $G = (X, U)$.

Résultat : le nombre k de composantes fortement connexes de (G) ainsi que la liste $\{C_1, C_2, \dots, C_k\}$ de ces composantes fortement connexes.

(0) Initialisation :

$$k = 0, W = X.$$

(1) Comporte les étapes suivantes:

(1.1) Choisir un sommet de W et le marquer d'un signe (+) et (-).

(1.2) Marquer tous les successeurs directs et indirects de x avec (+).

(1.3) Marquer tous les prédécesseurs directs et indirects de x avec (-).

(1.4) Poser $k = k + 1$ et C_k l'ensemble des sommets marqués avec (+) et (-).

(1.5) Retirer de W les sommets de C_k et effacer toutes les marques;

on pose $W = W - C_k$.

(1.6) On teste si $W = \emptyset$

Si oui, terminer et aller à (2)

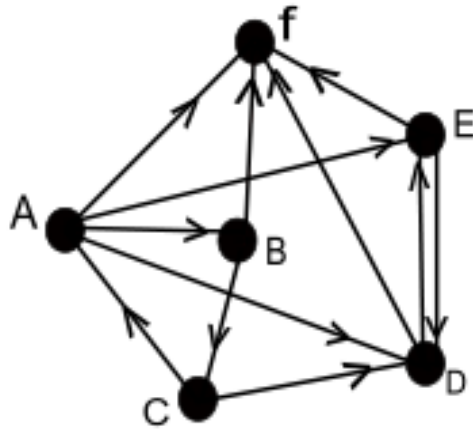
Sinon, aller à (1).

(2) Le nombre de composantes fortement connexes de (G) est k .

Chaque ensemble C_i , $i = 1, \dots, k$, correspond aux sommets d'une composante fortement connexe de (G) .

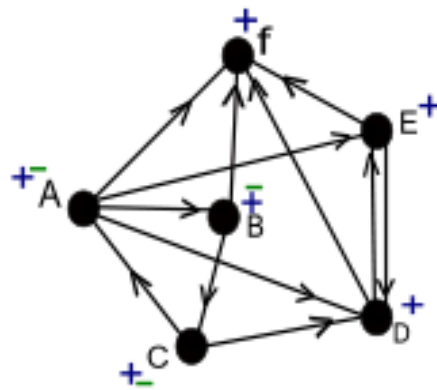
⇒ **Application :**

Soit le graphe $G = (V, E)$ suivant :



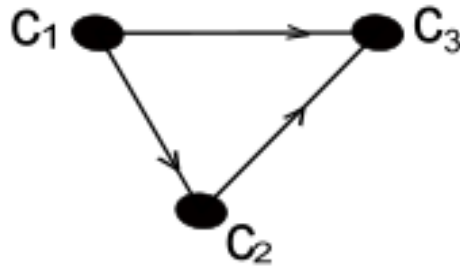
G est connexe. Il a donc une seule composante connexe, l'ensemble de tous ses sommets. Nous allons d'abord trouver la composante fortement connexe qui contient le sommet a . Rappel de l'algorithme:

- marquer par + et - le sommet A ;
- marquer par + chaque successeur non marqué + d'un sommet marqué + ;
- marquer par - chaque prédécesseur non marqué - d'un sommet marqué - ;
- quand on ne peut plus marquer, les sommets marqués + et - constituent la composante fortement connexe contenant a .



- La composante fortement connexe qui contient A est $C_1 = \{A, B, C\}$. On retire les sommets A, B et C et les arcs qui leur sont adjacents, puis on recommence par exemple avec le sommet D . On trouve que la composante fortement connexe qui contient D est $C_2 = \{D, E\}$. On retire ces sommets. Il n'y a plus que le sommet F . Ainsi, la troisième

composante fortement connexe est $C_3 = \{F\}$, donc le graphe réduit est donnée dans la figure



1.4 Distance et diamètre dans les graphes

Dans un graphe $G = (S, A)$ on appelle **distance** d'un sommet à un autre la longueur du plus court chemin d'un sommet à l'autre, ou ∞ s'il n'y a pas de tel chemin:

$$\forall x, y \in S, d(x, y) = \begin{cases} k & \text{si plus court chemin de } x \text{ vers } y \text{ est de longueur } k \\ \infty & \text{Sinon} \end{cases}$$

- **Le diamètre** d'un graphe est la plus grande distance entre deux sommets.

Remarque 1.4.1 Dans un graphe non-orienté $d(x, y) = d(y, x)$ mais ce n'est pas forcément vrai dans un graphe non-orienté, A noter aussi, la distance d'un sommet à lui-même est toujours nulle ($d(x, x) = 0$)

Exemple 1.4.1 On considère le même graphe de l'exemple 1.2.1 donc la distance et le diamètre: $d(1, 8) = 4$ et $d(8, 1) = 1$

Dans cet exemple nous sommes obligés de calculer tous les chemins possibles pour être sûr de trouver le plus court/long. Nous consacrerons dans le 3 chapitre au problème de la recherche du plus court chemin dans un graphe.

1.5 Opération sur les graphes

1.5.1 Somme cartésienne de deux graphes

On appelle somme cartésienne de deux graphes $G = (X(G), E(G))$ et $H = (X(H), E(H))$, notée $G \square H$.

Le graphe dont l'ensemble des sommets est le produit Cartésien $X(G) \times X(H)$ et/ou deux sommets $(u; u')$ et $(v; v')$ sont adjacents si et seulement si l'une des propriétés suivantes est vérifiée :

$$u = v \text{ et } u'v' \in E(H)$$

ou

$$uv \in E(G) \text{ et } u' = v'$$

La figure suivant montre la somme Cartésienne de deux graphes:

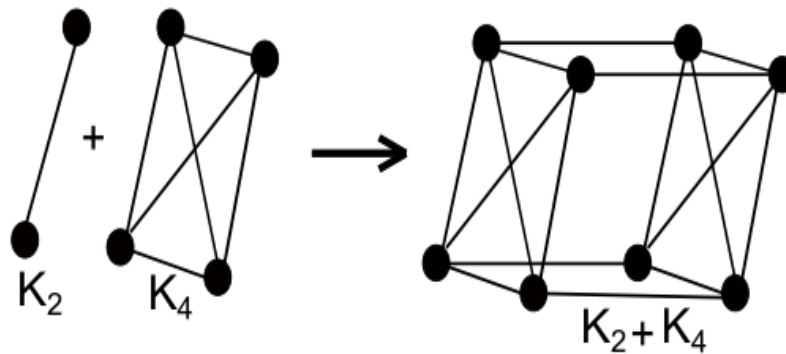


Figure 5.1: Somme cartésienne de deux graphes.

On note que le nombre de sommets dans $G \times H$ est $|X(G)| \cdot |X(H)|$ et que le nombre d'arêtes est : $|X(G)| \cdot |X(H)| + |X(H)| \cdot |E(G)|$.

1.5.2 Produit Cartésien de deux graphes

Le produit Cartésien de deux graphes $G = (X(G), E(G))$ et $H = (X(H), E(H))$ est le graphe note $G \times H$. Deux sommets (u, v) et (u', v') sont adjacents si et seulement si

$$uu' \in E(G) \text{ et } vv' \in E(H).$$

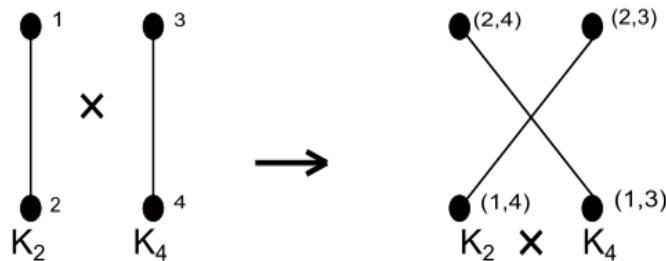


Figure 5.2: Produit Cartésien de deux graphes.

1.5.3 Morphismes de graphes

Graphes Orientés

Soit $G = (X, E)$ et $H = (T, B)$ deux graphes orientés. On dit qu'une application φ de X dans T réalise un morphisme du graphe G vers le graphe H si elle vérifie la propriété suivante:

$$(x, y) \in E \implies (\varphi(x), \varphi(y)) \in B$$

Si φ est une application injective, on dit que le graphe G s'injecte dans le graphe H . On dit aussi parfois que le graphe H contient le graphe G .

Graphes non-Orientés

Si φ est une application injective, on dit que le graphe G s'injecte dans le graphe H . On dit aussi parfois que le graphe H contient le graphe G .

1.5.4 Isomorphisme de graphes

Soient $G = (X, E)$ et $G' = (X', E')$ deux graphes. S'il existe une relation univoque $m : X \rightarrow X'$ telle que $(x_1; x_2) \in E, (m(x_1); m(x_2)) \in E'$, alors G et G' sont isomorphes.

1.6 Représentations matricielles

Une matrice $n \times m$ est un tableau de nombres réels à n lignes et m colonnes ou n et m sont des entiers naturels non nuls. n et m sont les dimensions de la matrice, une matrice est symbolisée par une lettre écrite en majuscule, par exemple A , On note l'élément a_{ij} l'élément situé à l'intersection de la ligne i et de la colonne j (La ligne i est toujours nommée en premier).

Exemple 1.6.1 Soit A est une matrice $n \times m$, on la note comme suit :

$$A = \begin{pmatrix} a_{11} & \cdots & a_{1j} & \cdots & a_{1m} \\ \cdot & & \cdot & & \cdot \\ \cdot & & \cdot & & \cdot \\ \cdot & & \cdot & & \cdot \\ a_{n1} & \cdots & \cdots & \cdots & a_{nm} \end{pmatrix}$$

1.6.1 Matrice d'adjacence

Les outils classiques d'algèbre linéaire peuvent également être utilisés pour coder les graphes. La première idée consiste à considérer chaque arc comme un lien entre deux sommets. Considérons un graphe $G = (X, E)$ comportant n sommets. La matrice d'adjacence de G est égale à la matrice $U = (u_{ij})$ de dimension $n \times n$ telle que

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

Une telle matrice, ne contenant que des $\{0\}$ et des $\{1\}$ est appelée de manière générale, une Matrice Booléenne. Dans le cas d'un graphe simple non orienté, on considère qu'à chaque

arête (i, j) correspondent deux arcs de la forme (i, j) et (j, i) , dans ce cas la matrice d'adjacence est symétrique.

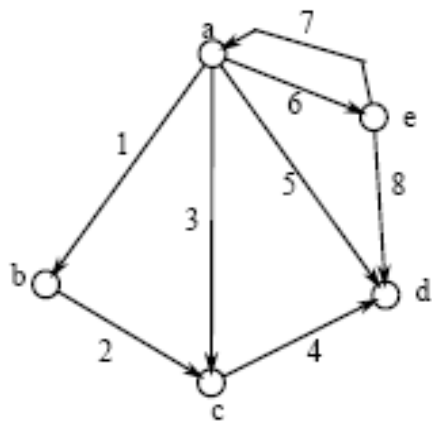


Fig 6.1 : graphe quelconque.

Sa matrice d'adjacence est :

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

1.6.2 Matrice d'incidence

La seconde idée permettant une représentation matricielle d'un graphe exploite la relation d'incidence entre arêtes et sommets. Considérons un graphe orienté sans boucle $G = (X, U)$ comportant n sommets $\{x_1, \dots, x_n\}$ et m arêtes $\{U_1, \dots, U_m\}$. On appelle matrice d'incidence (aux arcs) de G la matrice $M = (m_{ij})$ de dimension $n \times m$ telle que :

$$m_{ij} = \begin{cases} +1, & \text{si } x_i \text{ est l'extrémité initiale de l'arc } U_j \\ -1 & \text{si } x_i \text{ est l'extrémité terminale de l'arc } U_j \\ 0, & \text{sinon.} \end{cases}$$

Pour un graphe non orienté sans boucle, la matrice d'incidence (aux arêtes) est définie par:

$$m_{ij} = \begin{cases} 1, & \text{Si } x_i \text{ est une extrémité de } U_j; \\ 0 & \text{sinon} \end{cases}$$

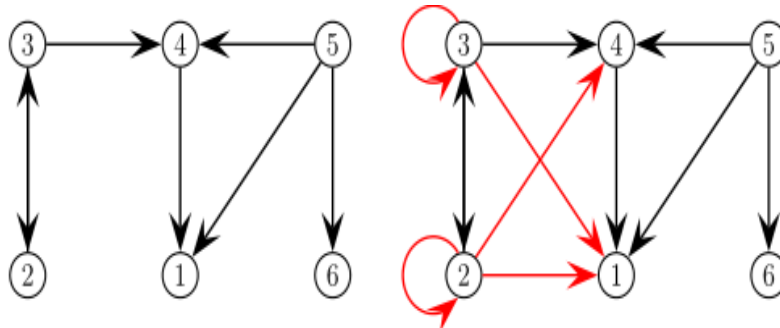
Propriété: Quel que soit le graphe non orienté G à n sommets $\{x_1, x_2, \dots, x_n\}$, ($n \in \mathbb{N}^*$) et de matrice d'adjacence de A , le terme a_{ij} de la matrice A^p ou $p \in \mathbb{N}^*$, est égal au nombre de chaîne de longueur p qui relie les sommets x_i et x_j .

1.7 Fermeture transitive d'un graphe

Définition 1.7.1

On appelle fermeture transitive d'un graphe $G = (S, A)$ le plus petit graphe G^* transitif et contenant G (c'est à dire tel que G soit un graphe partiel de $G^* = (S, B)$ avec $A \subset B$) Etant donné un graphe $G = (X, E)$, construire le graphe $G^+ = (X, E^+)$ tel que $(x, y) \in E^+$,il existe un chemin de G allant de x à y . G^+ s'appelle la fermeture transitive de G .

Calculer la fermeture transitive du graphe



La fermeture transitive d'un graphe peut être obtenue à partir de la matrice d'adjacence.

Théorème 1.7.1 (*fermeture transitive*) Soit $G = (S, A)$ un graphe de matrice d'adjacence M et de fermeture transitive G^* alors la matrice d'adjacence M^* de G vérifie

$$M^* = \sum_{k=1}^{\infty} M^k$$

Où la somme (au sens de l'algèbre de boole binaire) est en fait une somme finie.

Exemple 1.7.1 *Calcul de la fermeture transitive via la matrice d'adjacence :*

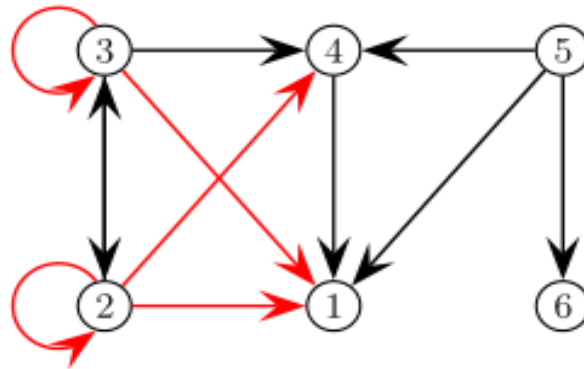


Fig 7.1-fermeture transitive de la matrice d'adjacence.

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

$$M + M^2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, M^* = M + M^2 + M^3 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\text{Car } M^* + M^4 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} = M^*.$$

1.8 Quelques types de graphes

Plusieurs types de graphes ont été définies par rapport à leurs propriétés structurelles. On a les types suivantes présentées qui sont celles nous retrouverons plus tard, dans les chapitres suivants.

1.8.1 graphe valué

Un graphe valué $G = (S, A, \nu)$ est un graphe (S, A) muni d'une application $\nu : A \longrightarrow R$.

Le graphe peut être représenté par la matrice des valuations :

$W \in M_n(\mathbb{R})$ telle que

$$W_{ij} = \begin{cases} \infty & \text{si } (x_i, x_j) \notin A \\ \nu((x_i, x_j)) & \text{si } (x_i, x_j) \in A \end{cases}$$

Pour un graphe valué on ajoutera sur le diagramme sagittal les valuations de chaque arc ou arête.

Exemple 1.8.1 On calcule la matrice des valuations et d'adjacence de graphe suivant:

$$W = \begin{pmatrix} \infty & \infty & \infty & \infty \\ -2 & 3 & \infty & \infty \\ 4 & 4 & -1 & 2 \\ -1 & -1 & -1 & \infty \end{pmatrix} \quad \text{et} \quad A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

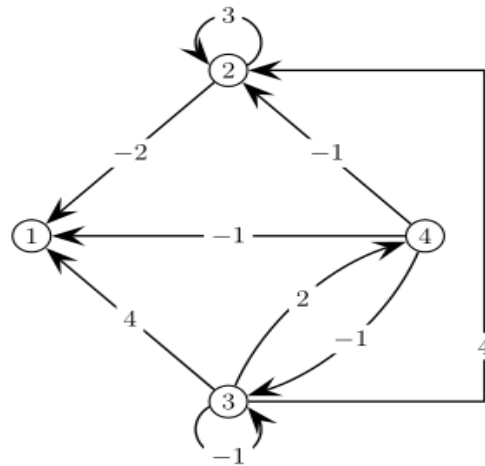


Figure 8.1: graphe valué orienté

1.8.2 Graphe complet

On appelle graphe complet un graphe dont tous les sommets sont adjacents. Un graphe simple complet d'ordre n est noté K_n , le graphe de la figure 8.2 montre un graphe complet K_2 .

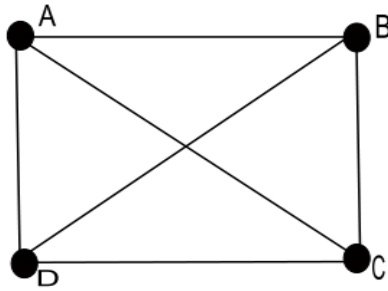


Figure 8.2: Graphe complet.

1.8.3 Graphe planaire

Un graphe est dit planaire si on peut le dessiner sur un plan de telle façon que les arêtes ne se coupent pas, en dehors de leurs extrémités. Ce type de graphe est particulièrement utilisé dans les problèmes de circuits imprimés (ces circuits, construits sur des surfaces planes, constituent actuellement l'une des limitations des développements de l'informatique).

- Un graphe fini est planaire si et seulement si il ne contient pas de sous-graphe qui est une expansion de K_5 (la clique à 5 sommets) ou $K_{3,3}$ (le graphe complet biparti à 3+3 sommets).

Exemple 1.8.2 *Rendre le graphe suivant planaire il faut déplacer les sommets 2 et 4.*

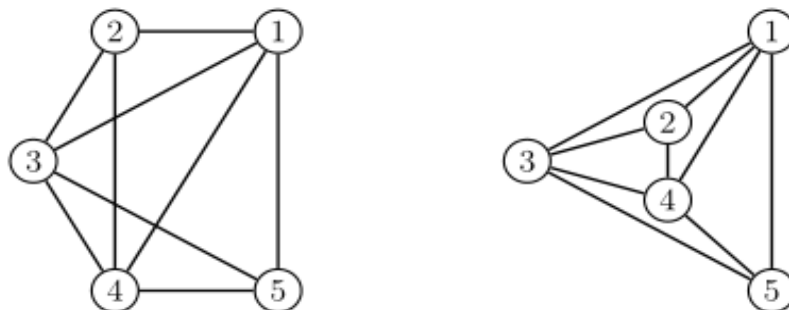
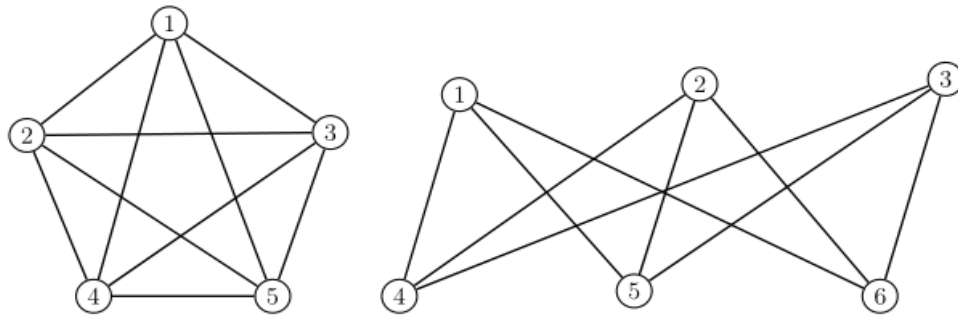


Figure 8.3-Graphe planaire.

Figure 8.4-Graphe K_5 et $K_{3,3}$

1.8.4 Graphe biparti

Définition 1.8.1 *Un graphe est dite biparti ses l'ensemble de si sommets peut être réparti en deux classes X_1 et X_2 telles que deux sommets de la même classe ne soient pas adjacents.*

On le note $G = (X_1, X_2, U)$ avec:

$$X_1 \cup X_2 = X$$

$$X_1 \cap X_2 = \emptyset$$

- Si $|X_1| = |X_2|$ on dit que G est biparti équilibré.
- Un graphe G est dit biparti complet, si tout sommet de X_1 est adjacent à tout sommet de X_2 . Si de plus le graphe G est simple, alors G est un graphe simple biparti -complet, on le note $K_{p,q}$ avec $|X_1| = p$ et $|X_2| = q$.

Exemple 1.8.3 *La figure 8.5 montre un graphe biparti complet.*

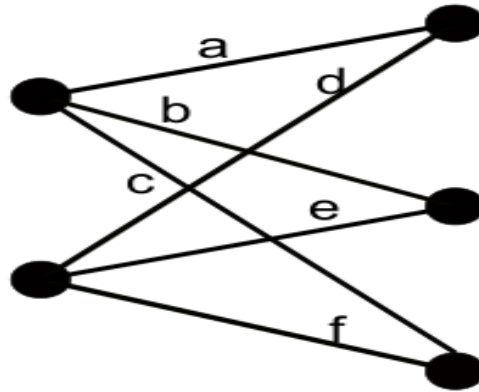


Figure 8.5-Graphe biparti Complet

1.8.5 Hypercube

Hypercube de dimension n , noté Q_n est le graphe dont l'ensemble des sommets, est formé de n -uplets, binaires et deux sommets sont adjacents, si et seulement s'il diffèrent exactement au une seule composante (coordonnée). Notons que $Q_0 = k_1$ (sommets isolés), $Q_1 = k_2$ (une chaîne de longueur 1) et d'une manière générale, Q_n peut être défini récursivement en utilisant la somme cartésienne par $Q_{n+1} = Q_n \square k_2$ ou bien $Q_{n+1} = Q_n \square Q_n$.

- Un sommet s d'un graphe G est une "racine" (resp. une "antiracine" s'il existe un chemin joignant s à chaque sommet du graphe G (resp. joignant chaque sommet de G à s), à l'exception du sommet lui-même. C'est -à-dire: $\forall x \in X - \{s\}$, il existe un chemin de s à x (resp. de x à s).

1.8.6 Arbre

Un arbre est un graphe $G = (X, U)$ connexe et sans cycle. D'après la définition, un arbre est un graphe simple sans boucles, ayant $(n - 1)$ arcs.

On appellera:

- **racine de l'arbre** le seul sommet de G qui n'a pas de prédécesseur.
- **feuilles de l'arbre** les sommets qui n'ont pas de successeur.

- **nœuds de l'arbre** tous les autres sommets.
- **branche de l'arbre** tout chemin de la racine vers une feuille.
- **descendant de x** les successeurs de x .
- **ascendant de x** le prédécesseur de x .

Lorsque chaque sommet à au plus 2 successeurs on parle aussi d'arbre binaire.

Exemple 1.8.4 Représenter l'arbre de la figure 8.6 par une liste de prédécesseurs

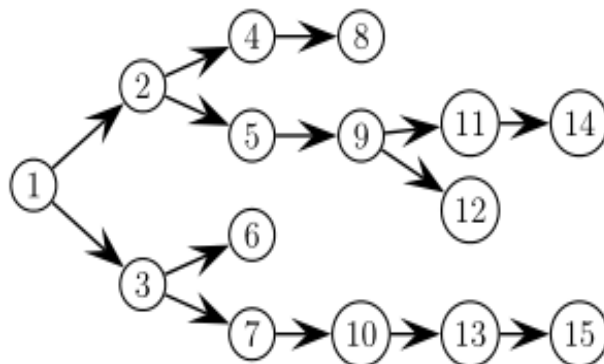


Figure 8.6 :Arbre

- Pred = 0 1 1 2 2 3 3 4 5 7 9 9 10 11 13
- La racine du graphe est le sommet 1
- Les feuilles sont les sommets 8, 14, 12, 6 et 15
- Une branche de l'arbre $C = (1, 2, 5, 9, 12)$ (chemin de la racine jusqu'à 12).

1.8.7 Polyarbre

Un polyarbre est un graphe orienté $G = (X, U)$, connexe et acyclique tel qu'il existe au maximum une chaîne entre deux sommets de X .

1.8.8 Arborescence

Un graphe $G = (X, U)$, avec $|X| = n \geq 2$ sommets est une arborescence de racine s si:

- ▶ G est un arbre.
- ▶ s est un racine de G .

1.8.9 Graphes triangulés

Un graphe non orienté $G = (X, U)$ est triangulé si et seulement si chaque cycle de longueur supérieure à quatre possède une corde c'est-à-dire deux sommets non consécutifs dans le cycle sont voisins.

Exemple 1.8.5 Soit les deux graphes suivantes:

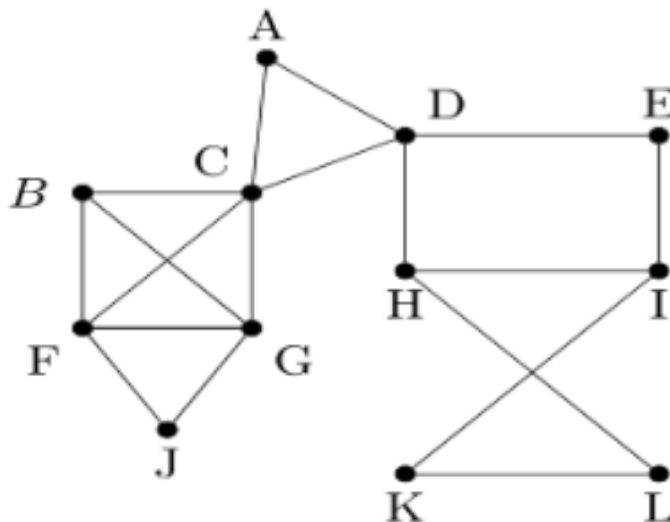


Figure 8.7: Un graphe non orienté.

•Le graphe de la figure 8.7 n'est pas triangulé car dans le cycle $[D E I H D]$ les sommets D et I ne sont pas voisins dans le graphe ni les sommets E et H . De même, le cycle $[H I K L H]$ n'admet pas de corde. Par contre dans le graphe de la figure 8.8, pour les mêmes cycles, il existe bien des cordes,

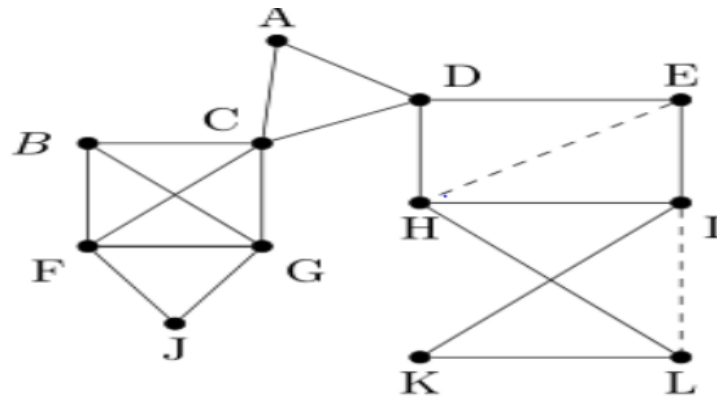


Figure 8.8: Un graphe simple.

ici $\{E, H\}$ et $\{I, L\}$, ainsi que pour tous les autres cycles, le graphe est triangulé.

Remarque 1.8.1 *Tout sous-graphe d'un graphe triangulé est triangulé.*

1.8.10 Graphes pondérés

On dit qu'un graphe est pondéré si on affecté à chaque arête un nombre positif (quel que soit sa signification).

- ▶ Ce nombre positif est alors appelé poids de l'arête.
- ▶ Le poids d'une chaîne est la somme des poids des arêtes qui la compose.
- ▶ La plus courte chaîne entre deux sommets est la chaîne de poids minimal entre ces deux sommets.

1.8.11 Graphe étiqueté

Les graphes étiquetés, ou automates, ont donné lieu depuis une cinquantaine d'années à une théorie mathématique abstraite, riche et diversifiée, possédant de nombreuses applications. On appelle graphe étiqueté un graphe où toutes les arêtes portent une étiquette (lettre, mot, nombre, symbole, code, ...). Ces symboles sont appelés des étiquettes. Soit G un graphe orienté étiqueté où un sommet est marqué début (D) et un autre fin (F). Dans un graphe étiqueté, une suite d'étiquettes est un mot reconnu par le graphe s'il existe une

chaîne telle que chacune des étiquettes soit associée (dans l'ordre) à une arête de cette chaîne. Sinon, le mot est dit refusé.

Exemple 1.8.6 On considère le graphe étiqueté de la figure 8.7 :

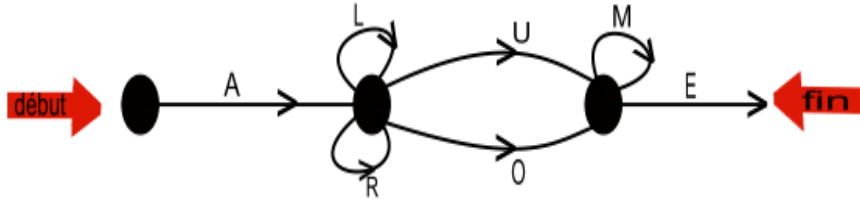


Figure 8.7: Mots reconnus par un graphe orienté.

► Les mots ALLUME et AROME sont reconnus par ce graphe, mais pas le mot ARME.

Modèles graphiques probabilistes

2.1 Introduction

Les modèles graphiques probabilistes sont des représentations graphiques de distributions de probabilité, qui ont des propriétés très utiles et qui rendent la manipulation algébrique des modèles probabilistes plus facile. L'intérêt de l'utilisation des graphes est double:

- les graphes permettent de construire des systèmes complexes basés sur la combinaison de sous-graphes simples qui sont reliés par des relations;
- les graphes sont une structure de données associée à plusieurs algorithmes efficaces. Dans le reste de cette section, certains concepts importants liés aux modèles graphiques probabilistes sont introduits.

Les modèles graphiques portent de nombreux noms : réseaux de croyance, réseaux probabilistes, réseaux d'indépendance probabiliste ou encore réseaux bayésiens. Il s'agit d'un formalisme pour représenter de façon factorisée une distribution jointe de probabilités sur un ensemble de variables aléatoires. Ils ont révolutionné le développement des systèmes intelligents dans de nombreux domaines. Ils sont le mariage entre la théorie des probabilités et la théorie des graphes. Ils apportent des outils naturels permettant de traiter deux grands problèmes couramment rencontrés en intelligence artificielle, en mathématiques appliquées : l'incertitude et la complexité. Ils jouent en particulier un rôle grandissant dans la

conception et l'analyse d'algorithmes liés au raisonnement ou à l'apprentissage [Jordan, 1999],[Becker and Naïm, 1999], [Dawid, 1992].Donc ce chapitre a pour but de présenter, de manière superficielle, deux modèles graphiques probabilistes que sont les réseaux bayésiens et les chaînes de Markov. Avant de pouvoir les présenter nous rappelons les notions essentielles de probabilités pour leur compréhension.

2.2 Notions de probabilités

La théorie des probabilités a pour objectif de modéliser ce qui n'est pas déterminé à l'avance (par exemple un lancer de dé). La théorie des probabilités ne va pas permettre de prédire quelle issue va se réaliser mais quelle chance a chaque issue de se réaliser. Ainsi, nous allons associer à chaque issue possible un nombre entre 0 et 1 qui traduit notre estimation des chances que cette issue a de se réaliser, on appelle ce nombre la probabilité de cette issue. On appelle événement un ensemble d'issues.

2.2.1 Définitions principales

Soient Ω un ensemble fini non vide, \mathcal{E} une famille d'événements observables sur Ω . On appelle probabilité sur (Ω, \mathcal{E}) toute fonction $P : \mathcal{E} \rightarrow [0, 1]$ vérifiant :

1. $P(\Omega) = 1$.
2. Pour toute suite $(A_i)_{i \geq 1}$ d'événements de \mathcal{E} deux à deux disjoints $:(\forall i, j, A_i \cap A_j = \emptyset)$; $P(\bigcup_{i \in \mathbb{N}^*} A_i) = \sum_{i=1}^{+\infty} P(A_i)$.
3. Le triple (Ω, \mathcal{E}, P) s'appelle espace probabilisé.

Une variable aléatoire est une fonction X définie sur Ω

$$X : \Omega \rightarrow D_X$$

$$\omega \longrightarrow X(\omega)$$

Pour $x \in D_X$ on note alors $\{X = x\}$ l'événement $\{\omega \in \Omega | X(\omega) = x\}$. D_X est le domaine de définition de X . Donc une variable aléatoire n'est rien d'autre qu'une fonction classique. Elle permet de caractériser des événements par une valeur. On parle de variable aléatoire à valeurs discrètes ou variable aléatoire discrète lorsque le domaine de définition de la variable aléatoire est fini.

Dans ce chapitre l'ensemble Ω est fini, les variables aléatoires sont toujours considérées comme discrètes. La plupart des ensembles seront représentés par une lettre en gras (S, X, \dots). Pour les ensembles de variables, nous noterons D_X le produit cartésien des domaines des variables composant l'ensemble X .

La distribution de la loi de X est $x \rightarrow P(X = x)$. Pour connaître la loi d'une variable aléatoire il faut connaître l'ensemble de ses valeurs possibles et la probabilité avec laquelle elle réalise chaque valeur.

2.2.2 Propriétés générales d'une probabilité

Toute probabilité P sur (Ω, ε) vérifie les propriétés suivantes:

1. $P(\emptyset) = 0$.
2. Si $(A \cap B) = \emptyset, P(A \cup B) = P(A) + P(B)$.
3. Si les $A_i (1 \leq i \leq n)$ sont deux à deux disjoints:

$$P(\cup_i A_i) = \sum_{i=1}^n P(A_i).$$

2.2.3 Probabilités sur plusieurs variables

• Probabilité jointe

Soient X et Y deux variables aléatoires sur le même univers Ω , la probabilité jointe de X et Y est la fonction définie sur $D_X \times D_Y$ par

$$P : D_X \times D_Y \rightarrow [0, 1]$$

$$(x, y) \rightarrow P(x, y) \text{ où } P(x, y) = P(\{X = x\} \cap \{Y = y\})$$

Cette définition peut être étendue à tout ensemble fini de variables aléatoires. La distribution jointe est définie par:

$$P(\mathbf{X}_1, \dots, \mathbf{X}_n) = \frac{\prod_{i=1}^m f_i(\mathbf{S}_i)}{\sum_{x \in \mathbf{D}_X} \prod_{i=1}^m f_i(X[\mathbf{S}_i])}$$

- **Probabilité marginale**

Soient U un ensemble fini, non vide de variables aléatoires, $V \subset U$ un sous-ensemble non vide, et la probabilité jointe sur les variables de U . On appelle marginalisation de P sur V la fonction

$$\forall v \in D_v, P(v) = \sum_{v' \in D_{v'}} P(v, v')$$

Cette fonction est la probabilité jointe des variables V .

2.2.4 Probabilités conditionnelles, Indépendance

- **Probabilité conditionnelle**

Soit deux événements A et M , on suppose que M s'est produit (donc $P(M) \neq 0$). Alors s'il existe un lien (de cause et effets) entre A et M , cette information va modifier la probabilité de A . $P(A|M)$ est la probabilité de A conditionnellement à M ou sachant M et elle est définie comme suit :

$$P(A|M) = \frac{P(A, M)}{P(M)}$$

$P(A|M)$ définit bien une probabilité. Remarquons que si A et M sont indépendants, alors la connaissance de M n'apporte rien sur la connaissance de A .

- **Indépendance conditionnelle**

L'indépendance conditionnelle est un concept très important pour les distributions de probabilités portant sur plusieurs variables. Elle permet de simplifier la structure d'un modèle graphique et les calculs nécessaires pour faire de l'inférence ou de l'apprentissage. La propriété d'indépendance conditionnelle d'une distribution jointe peut être directement lue sur le modèle graphique.

Définition 2.2.1 *Soit trois variables aléatoires A, B et C , on suppose que la distribution conditionnelle de A sachant B et C est telle, qu'elle ne dépend pas de la valeur de B :*

$$P(A|B, C) = P(A|C)$$

► On dit que A est conditionnellement indépendant de B étant donné C . Si A est conditionnellement indépendant de B étant donné C , alors la distribution jointe de A et B conditionnées sur C vaut : $P(A, B|C) = P(A|B, C)P(B|C) = P(A|C)P(B|C)$

Cela signifie que conditionnées sur C , les variables A et B sont statistiquement indépendantes.

2.2.5 Quelques règles de base en probabilité

Définition 2.2.2 *Soient N événements M_1, \dots, M_N complets et mutuellement exclusifs. La règle de la somme est définie par*

$$P(A) = \sum_{K=1}^N P(A, M_K)$$

Définition 2.2.3 *La règle du produit est définie par*

$$P(A, B) = P(A|B) \times P(B) = P(B|A) \times P(A)$$

Définition 2.2.4 *La règle de Bayes est définie par*

$$P(M_i|A) = \frac{P(M_i, A)}{P(A)} = \frac{P(A|M_i) \times P(M_i)}{P(A)}$$

- ▶ $P(M_i)$ est la probabilité a priori de M_i .
- ▶ $P(A|M_i)$ est la probabilité de A conditionnellement à M_i (vraisemblance).
- ▶ $P(M_i|A)$ est la probabilité a posteriori de M_i (conditionnellement à A).

Théorème 2.2.1 *Probabilités totales*

Soient N événements M_1, \dots, M_N complets et mutuellement exclusifs, alors:

$$P(A) = \sum_{K=1}^N P(A|M_K) \times P(M_K)$$

Théorème 2.2.2 *Bayes*

Soient N événements M_1, \dots, M_N complets et mutuellement exclusifs, alors:

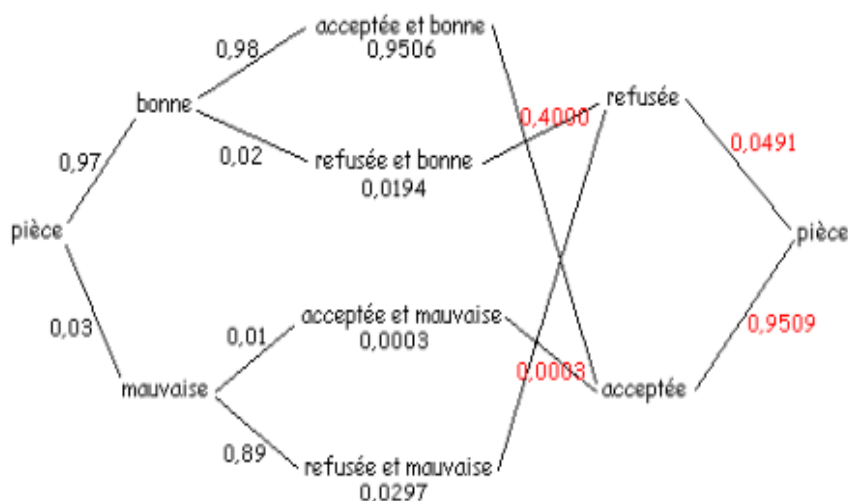
$$P(M_i|A) = \frac{P(A|M_i) \times P(M_i)}{\sum_K P(A|M_K) \times P(M_K)}$$

Exemple 2.2.1 *Un lot contient 3% de pièces défectueuses.*

a) on prélève au hasard un échantillon de 10 pièces. Les pièces étant très nombreuses, on admet que le tirage peut être considéré comme fait au hasard et avec remise. Soit X la variable aléatoire "nombre de pièces défectueuses dans l'échantillon".

- ▶ La probabilité pour qu'une pièce soit défectueuse est $p = 0,03$.
- ▶ La loi de probabilité de X est la loi binomiale de moyenne $\langle X \rangle = np = 10 \times 0,03 = 0,3$ et d'écart-type $\sigma = (npq)^{0,5} = 0,54$ donc $p(X = k) = C_{10}^k (0,03)^k (0,97)^{10-k}$.
- ▶ En particulier $p(X = 0) = (0,97)^{10} = 0,737$ et $p(X \geq 1) = 1 - p(X = 0) = 0,263$

b) On contrôle toutes les pièces mais le mécanisme de contrôle est aléatoire. Une pièce bonne est acceptée avec une probabilité de 0,98, une pièce défectueuse est refusée avec une probabilité de 0,99. Contrôle des pièces



- La probabilité de refuser une pièce à tort est $p(\text{bonne si refusée}) = p(\text{refusée et bonne})/p(\text{refusée}) = 0,4$.
- La probabilité d'accepter une pièce à tort est $p(\text{mauvaise si acceptée}) = p(\text{acceptée et mauvaise})/p(\text{acceptée}) = 0,0003$.

2.3 Les graphes probabilistes

Un graphe probabiliste est un graphe orienté et pondéré dans lequel :

- il y a au plus un arc d'un sommet à l'autre.
- la somme des poids des arcs issus d'un même sommet est égale à 1.

1. Les poids des arcs sont alors des probabilités (nombres réels compris entre 0 et 1).
2. Un graphe probabiliste indique les différents états possibles d'un système (sommets du graphe) et les probabilités de passage d'un état à l'autre (poids des arcs), comme il est possible de rester dans le même état, un graphe probabiliste peut avoir des boucles. La figure 2.1 montre un graphe probabiliste.

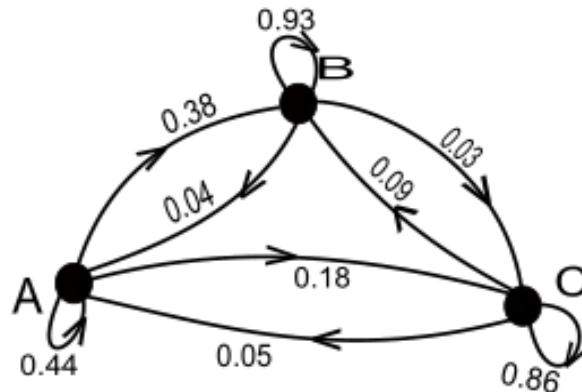


Figure 2.1: Graphe probabiliste.

2.4 Matrice de transition

Soit G un graphe probabiliste d'ordre n dont les sommets sont numérotés de 1 à n . La matrice de transition M de G est la matrice carrée d'ordre n telle que m_{ij} est égal à la probabilité portée par l'arc reliant le sommet i au sommet j s'il existe et 0 sinon.

Exemple 2.4.1 La matrice de transition de graphe de la figure 2.1 est :

$$m_{ij} = \begin{pmatrix} 0,44 & 0,38 & 0,18 \\ 0,04 & 0,93 & 0,03 \\ 0,05 & 0,09 & 0,86 \end{pmatrix}$$

Propriété: Le produit de deux matrices de transition est une matrice de transition. En particulier, si T est une matrice de transition, T^n est aussi une matrice de transition.

2.4.1 Etat probabiliste

L'état probabiliste d'un système est une loi de probabilité définie sur l'ensemble des états possibles. Cette loi sera représentée par une matrice ligne. L'état probabiliste à l'étape n

est une matrice ligne de la forme $P_n = [a_n \ b_n]$ pour deux états ou $P_n = [a_n \ b_n \ c_n]$ pour trois états telle que la somme des termes est égale à 1.

Propriétés:

1. Soit M la matrice de transition d'un graphe probabiliste, P_0 l'état probabiliste initial, P_n l'état probabiliste à l'étape n et P_{n+1} l'état probabiliste à l'étape $n+1$. On a pour tout entier n :

$$P_{n+1} = P_n \times M.$$

2. Soit M la matrice de transition d'un graphe probabiliste ayant 2 sommets, P_0 la matrice ligne décrivant l'état probabiliste initial et P_n la matrice ligne décrivant l'état probabiliste à l'étape n . On a, pour tout entier n : $P_n = P_0 \times M^n$

Remarque 2.4.1 *On admet que ces propriétés restent vraies pour les graphes probabilistes ayant trois sommets ou plus.*

Soit la matrice de transition suivante :

$$M = \begin{pmatrix} 0,6 & 0,4 \\ 0,2 & 0,8 \end{pmatrix}$$

- ▶ L'état probabiliste initial est donc $P_0 = (0,1 \quad 0,9)$.
- ▶ On a donc, par exemple, $P_1 = P_0 \times M = (0,24 \quad 0,76)$.
- ▶ On a aussi, par exemple, $P_2 = P_0 \times M^2 = (0,296 \quad 0,704)$.

2.4.2 Etat stable

On dit qu'un état probabiliste P est stable s'il reste le même dans la répétition de l'expérience décrite par le graphe probabiliste de matrice de transition M , c'est-à-dire

$$P \times M = P$$

Pour tout graphe probabiliste d'ordre 2 dont la matrice de transition M ne comporte pas de zéro, l'état probabiliste P_n converge toujours vers l'état stable P vérifiant

$P \times M = P$, indépendamment de l'état probabiliste initial P_0 l'état P est appelé état stable du système.

Exemple 2.4.2 Soit la matrice de transition :

$$M = \begin{pmatrix} 0,7 & 0,3 \\ 0,2 & 0,8 \end{pmatrix}$$

- L'état initial $P_0 = (0,4 \quad 0,6)$.
- On a donc, par exemple, $P_1 = P_0 \times M = (0,4 \quad 0,6)$.
- On a aussi, par exemple, $P_2 = P_0 \times M^2 = (0,4 \quad 0,6)$.

De proche en proche que, $P_n = P_0$ pour tout entier naturel n . L'état d'écrit par la matrice P_0 est donc un 'état stable.

2.5 Quelques Modèles graphiques probabilistes.

2.5.1 Modèle Graphique Probabiliste

Un modèle graphique probabiliste est un triplet $(\mathbf{X}, \mathbf{D}, \mathbf{F})$ avec $\mathbf{X} = \{X_1, \dots, X_n\}$, un ensemble de variables, $\mathbf{D} = \{D_{X_1}, \dots, D_{X_n}\}$, un ensemble de domaines finis, et $\mathbf{F} = \{f_1, \dots, f_m\}$, un ensemble de fonctions à valeurs réelles positives, chacune est définie sur un sous-ensemble de variables $\mathbf{S}_i \subseteq \mathbf{X}$ (i.e. la portée). Les PGM couvrent les réseaux Bayésiens, et les chaînes de Markov, que nous présentons en partie dans les sections suivantes, la figure 5.1 montre un modèle graphique orienté.

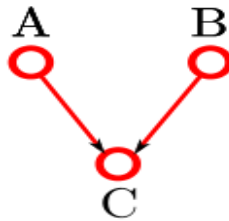


Figure 5.1 :Modèle graphique orienté

La figure 5.2 montre un mdèle de graphe non orienté.

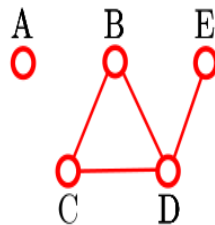


Figure 5.2 Modèle graphique non orienté

2.5.2 Réseaux bayésiens

Les **réseaux bayésiens** sont un modèle graphique orienté et sans circuit, sont utiles pour exprimer des relations de cause à effet ils entre les variables du modèle graphique,et ils sont souvent utilisés pour faire du raisonnement à partir d'informations incomplètes.Le raisonnement à partir de données incomplètes est l'étape d'inférence.Les réseaux bayésiens représentent graphiquement les indépendances conditionnelles entre les variables. Donc les réseaux bayésiens sont généralement définis à partir des variables aléatoires (qui forment les nœuds du graphe) et un ensemble de tables de probabilités conditionnelles.

La figure suivante représente un réseau bayésien contenant cinq variables. Il décrit par les saisons de l'année (X_1) si la pluie tombe (X_2), si l'arrosage est en marche (X_3), si le chemin est mouillé (X_4) et si le chemin est glissant (X_5). Toutes ces variables sont binaires. L'absence d'arc allant de X_1 à X_4 signifie que la saison n'influe pas directement sur l'état mouillé ou non du chemin. Ainsi le graphe indique que la probabilité jointe des cinq variables se décompose de la façon suivante:

$$P(\mathbf{X}_1, \dots, \mathbf{X}_5) = P(\mathbf{X}_1) \cdot P(\mathbf{X}_2 | \mathbf{X}_1) \cdot P(\mathbf{X}_3 | \mathbf{X}_1) \cdot P(\mathbf{X}_4 | \mathbf{X}_2, \mathbf{X}_3) \cdot P(\mathbf{X}_5 | \mathbf{X}_4)$$

Les tables de probabilités conditionnelles associées sont données à la figure 5.3. Par exemple, sachant que nous sommes en été ($X_1 = \text{été}$), nous avons une probabilité de 0.2 qu'il pleuve ($X_2 = \text{oui}$).

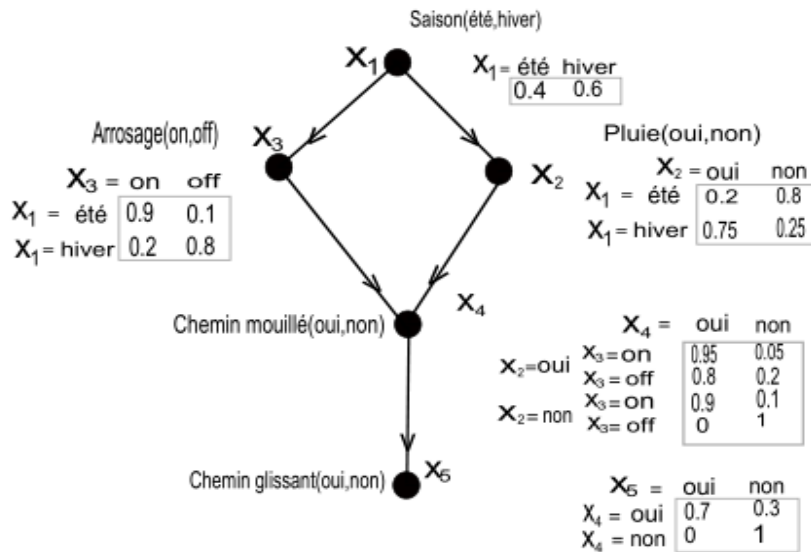


Figure 5.3: Un réseau bayésien représentant les dépendances entre 5 variables

2.5.3 Chaînes de Markov

Une chaîne de Markov est une suite de variables aléatoires $(X_n, n \in N)$ qui permet de modéliser l'évolution dynamique d'un système aléatoire : X_n représente l'état du système à l'instant n . La propriété fondamentale des chaînes de Markov, dite propriété de Markov, est que son évolution future ne dépend du passé qu'au travers de sa valeur actuelle.

Définition 2.5.1 *Un processus aléatoire $\{\mathbf{X}_t\}$ définie sur un espace d'états \mathbf{S} satisfait la propriété de Markov si : Pour tout sous ensemble \mathbf{I} de \mathbf{S} .*

$$P(\mathbf{X}_{t+\Delta t} \in \mathbf{I} / \mathbf{X}_u, 0 \leq u \leq t) = P(\mathbf{X}_{t+\Delta t} / \mathbf{X}_t); \forall \Delta u \geq 0$$

Un processus aléatoire vérifiant la propriété précédente est appelé processus Markovien. Formellement, une chaîne de Markov vérifie la propriété de Markov, à savoir, Sachant le présent, le futur est un dépendant du passé. Dans la suite de ce chapitre nous nous intéressons aux chaînes de Markov discrètes. Pour de tels processus la définition se simplifie car l'espace des étapes est dénombrable.

Définition 2.5.2 *Soit $\{\mathbf{X}_n\}$ une suite de variables aléatoires de (Ω, A, P) dans un espace fini ou dénombrable appelé espace des états. On dit que $\{X_n\}_{n \in N}$ est une chaîne de Markov si :*

$$P(\mathbf{X}_{n+1} = j / \mathbf{X}_0 = i, \mathbf{X}_1 = i_1, \dots, \mathbf{X}_n = i) = P(\mathbf{X}_{n+1} = j / \mathbf{X}_n = i)$$

Si de plus, $P(X_{n+1} = j / X_n = i)$ ne dépend pas de n , alors la chaîne de Markov est dit homogène si :

$$P(\mathbf{X}_n = j / \mathbf{X}_{n-1} = i = \dots = P(\mathbf{X}_3 = j / \mathbf{X}_2 = i) = P(\mathbf{X}_2 = j / \mathbf{X}_1 = i))$$

• Probabilité de transition et matrice de passage

La probabilité P_{ij} est appelée probabilité de transition ou bien de passage de l'état i à l'état j . On peut présenter ces probabilités sous forme matricielle appelé matrice de

► **Accessible:** j est accessible depuis i si il existe un chemin dans le diagramme de transition, partant de i et arrivant en j . Ceci se traduit également en termes des probabilités de transition en m pas et des probabilités de premier passage.

► **Communiquent:** On dit que deux états i et j communiquent si chacun est accessible depuis l'autre.

⇒ La relation de communication est symétrique et transitive, mais elle n'est pas nécessairement réflexive (quand la chaîne quitte un état i elle peut ne jamais revenir).

► **Irréductible:** On appelle classe irréductible tout sous ensemble d'états, maximal au sens de l'inclusion, composé d'états qui communiquent deux à deux.

⇒ Si tous les états de E communiquent deux à deux, E tout entier est la seule classe irréductible. On dit alors que la chaîne est irréductible.

► **Ergodicité:** T étant une matrice de transition, T est dite régulière (ou fortement ergodique) si, à partir d'une certaine puissance n , tous les éléments de T_n sont strictement positifs.

► **Périodique:** L'état i est dit périodique de période $k > 1$ si tous les entiers m tels que $p_{ii}^{(m)} > 0$ sont multiples de k . Un état qui n'admet pas de période est dit apériodique.

⇒ Si i est périodique de période k et communique avec j , on démontre que j est également de période k . Les classes irréductibles périodiques constituent un cas particulier que l'on ne rencontre pas dans les applications.

2.5.4 Lien entre chaînes de Markov et théorie des graphes

Dans le cas d'une chaîne de Markov où l'ensemble E (appelé ensemble des états) est fini, on peut représenter la chaîne sous forme graphique. En effet, les sommets représentent les états de la chaîne et les arrêtes représentent les probabilités de transition. Ainsi, en connaissant la matrice de transition, on peut déterminer le graphe associé à la chaîne.

Exemple 2.5.1 Si nous avons $E = \{a, b\}$, que la probabilité de passer de a à b (ou de b à a) est $\frac{1}{2}$ et la probabilité de rester en a ou b est aussi $\frac{1}{2}$, la matrice de transition sera

$$P = \begin{pmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{pmatrix}$$

et le graphe associé montre dans la figure 5.4.

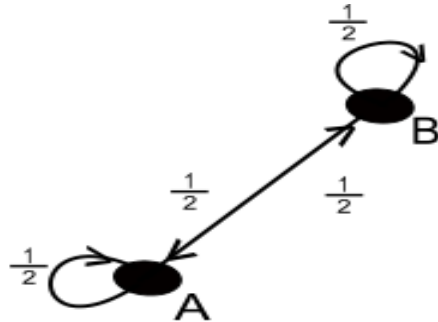


Fig 5.4 : Un graphe probabiliste
d'ordre 2.

Remarque 2.5.1 *On dit qu'une chaîne est irréductible si son graphe associé ne possède qu'une composante connexe. Nous tentons ici d'amener le lecteur à la découverte d'un lien entre le graphe*

d'une chaîne de Markov et sa loi stationnaire. Pour se faire, nous devons encore introduire deux définitions qui vont nous mener au théorème final de ce chapitre.

Définition 2.5.4 *On dit que T_i est le temps de premier passage dans l'état i si*

$$T_i = \inf \{n \geq 1; X_n = i\}$$

Le temps moyen de premier retour en i est défini comme

$$E_i(T_i).$$

On dira qu'une chaîne est positivement récurrente si pour tous les états i on a la propriété suivante

$$E_i(T_i) < \infty.$$

Théorème 2.5.1 *Une chaîne de Markov irréductible est positivement récurrente si et seulement si elle possède une unique loi stationnaire. Cela signifie que pour une chaîne de Markov possédant une composante connexe dans son graphe, il y a une loi stationnaire unique.*

Problèmes d'optimisation pour les graphes valués

Les graphes permettent de manipuler plus facilement des objets et leurs relations avec une représentation graphique naturelle. Beaucoup de problèmes peuvent être modéliser en utilisant des graphes valués. Les problèmes de cheminement dans les graphes, en particulier la recherche du plus court chemin, comptent parmi les problèmes les plus anciens de la théorie des graphes et les plus importants par leurs application. Dans ce chapitre nous allons rappeler brièvement quelques problèmes d'optimisations pour les quelles nous donnons quelque algorithmes de résolutions pour chaque problème. Parmi les problèmes nous citons: Problème du plus court chemin, Problème de l'arbre couvrant de poids minimum, Problème d'affectation, Problème d'ordonnancement.

3.1 Problème de l'arbre couvrant de poids minimum

Le problème de l'arbre couvrant, consiste à étudier le problème de l'arbre couvrant de poids minimum, qui joue un rôle important dans l'optimisation combinatoire et il figure parmi les problèmes les plus étudiés dans la littérature. Pour avoir un aperçu sur ses propriétés et les algorithmes qui permettent de le résoudre, et dans le cas des graphes non-orientés on s'intéresse aussi aux arbres couvrants.

Soit $G = (X, U)$ un graphe on appelle **arbre couvrant** de G un graphe $G' = (X, U')$ graphe partiel de G qui est un arbre. Cet arbre peut être obtenu par l'algorithme de Kruskal ou l'algorithme de Prim.

3.1.1 Algorithme de Kruskal

Si on associe à chaque arc d'un graphe $G = (X, U)$ une valeur (un poids). Le problème de coût minimum consiste à trouver un sous graphe qui est un arbre, dont la somme des poids des arcs est minimale. En 1956, J.B. Kruskal a donné un algorithme qui permet de résoudre un tel problème. L'idée de l'algorithme de Kruskal est tout d'abord de numéroter les arcs par ordre des poids croissants. Ensuite de construire progressivement l'arbre A en ajoutant dans leurs ordre, les arcs un par un. Un arcs est ajouté si seulement si son adjonction à A ne détermine pas de cycle, c'est à dire si A ne perd pas sa notion d'arbre, sinon on passe à l'arcs suivant dans l'ordre de la numérotation.

Énoncé :

► Données: un graphe valué $G = (X, U, c)$.

Résultat : ensemble d'arcs W .

(0) Initialisation :

Numéroter les arcs de G dans l'ordre des poids croissants :

$$c(u_1) \leq c(u_2) \leq \dots \leq c(u_m)$$

soit $W = \emptyset; i = 1$

(1) si $(X, W \cup \{u_1\})$ contient un cycle aller en (3)

sinon aller en (2)

(2) on pose $W := W \cup \{u_1\}$ aller en (3)

(3) si $i = m$ terminé ,

$A = (X, W)$ est l'arbre de poids minimum $c(W) = \sum c(u_i)$ pour $u_i \in W$

sinon $i := i + 1$ aller en (1)

L'algorithme s'arrête lorsque le nombre d'arcs retenus est égal à $n - 1$.

3.1.2 Algorithme de PRIM

L'algorithme de Prim, est un algorithme qui permet de trouver un arbre couvrant minimal dans un graphe connexe valué et non orienté. En d'autres termes, cet algorithme trouve un sous-ensemble d'arêtes formant un arbre sur l'ensemble des sommets du graphe initial, et tel que la somme des poids de ces arêtes soit minimale. Si le graphe n'est pas connexe, alors l'algorithme ne déterminera l'arbre couvrant minimal que d'une composante connexe du graphe. Il a été conçu en 1957 par Robert C.Prim.

Principe de l'algorithme de PRIM :

L'algorithme consiste à choisir arbitrairement un sommet et à faire croître un arbre à partir de ce sommet. Chaque augmentation se fait de la manière la plus économique possible.

Enoncé de l'Algorithme de PRIM :

⇒ **Données**: Soit $G = (X, E, P)$ un graphe probabiliste,

Soit I l'ensemble de nœuds déjà inclus dans l'arbre et NI l'ensemble de nœuds non encore inclus,

(0). Initialisation : $I = \emptyset$, $NI =$ l'ensemble de tous les nœuds;

(1). Mettre le nœud de départ dans I ;

(2). Si l'arbre est connexe alors Aller au (5);

(3). Trouver un nœud de NI dont la distance à un nœud quelconque a de I est minimal.

On relie ce nœud a au b et faire $I = I \cup b$, $NI = NI - b$;

(4). Aller au (3);

(5). Fin.

⇒ Le critère d'arrêt : L'algorithme s'arrête lorsque le nombre d'arcs retenus est égal à n .

⇒ **Résultat** : Trouver un arbre couvrant de poids minimum de G .

l'arbre couvrant de poids minimum associé au graphe de la figure 3.1 :

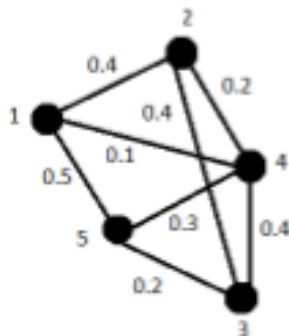


Fig 3.1—Graphe
probabiliste non
orienté.

Initialisation :

Soit $I = \emptyset$;

$NI = \{(1, 2), (1, 4), (1, 5), (2, 3), (2, 4), (3, 4), (3, 5), (4, 5)\}$

1^{er} Itération :

On a $e_2 = (1, 4)$ avec $p(1, 4) = 0.1 < p(1, 2) = 0.4 < p(1, 5) = 0.5$;

Soit $I = I \cup \{e_2\} = \{(1, 4)\}$;

Le graphe $G = (X, NI)$ n'est pas connexe, on pose alors :

$I = \{(1, 4)\}$;

$NI = \{(2, 3), (2, 4), (3, 4), (3, 5), (4, 5)\}$;

$|I| = 1$.

2^{ème} Itération :

On a $e_5 = (2, 4)$;

Soit $I = I \cup \{e_5\} = \{(1, 4)\} \cup \{(2, 4)\}$;

Le graphe $G = (X, NI)$ n'est pas connexe, on pose alors :

$I = \{(1, 4), (2, 4)\}$;

$NI = \{(3, 4), (3, 5), (4, 5)\}$;

$|I| = 2$.

Après l'application de tous les iterations de l'algorithme, on obtient un arbre couvrant minimal est représenté par le graphe $A = (X, NI)$, avec $|I| = n - 1 = 5 - 1 = 4$.

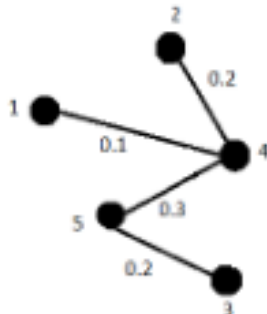


Fig 3.2- Arbre couvrant minimale

Théorème 3.1.1 *Soit G un graphe valué connexe. Les algorithmes de Kruskal et de Prim donnent un arbre recouvrant de poids minimal.*

Preuve. Algorithme de Kruskal

Nous allons montrer qu'à l'étape i , il existe un arbre recouvrant de poids minimal de G contenant le graphe A par récurrence sur i . Si $i = 0$, alors A est formé des sommets de G et d'aucune arête, donc est inclus dans tous les arbres recouvrant de poids minimal de G . Supposons le résultat vrai à l'étape $i - 1$. Soit alors un arbre recouvrant de poids minimal t , contenant le sous-graphe obtenu à l'étape $i - 1$. S'il contient l'arête ajoutée à l'étape i , cet arbre convient. Sinon, considérons le sous-graphe de G formé de t et de l'arête ajoutée à l'étape i . Il contient n arêtes, car t en contient $n - 1$, donc il contient un cycle. Comme t ne contient pas de cycle, ce cycle contient l'arête ajoutée à l'étape i . Si toutes les autres arêtes de ce cycle sont dans le sous-graphe de l'étape $i - 1$, alors l'arête ajoutée à l'étape i ne relie pas deux composantes connexes du sous-graphe de l'étape $i - 1$: contredit l'algorithme. Donc le cycle contient l'arête de l'étape i et au moins une autre arête qui n'est pas dans le graphe de l'étape $i - 1$. Par le choix de l'algorithme, cette arête est de poids supérieur à celui de l'arête de l'étape i . En retirant cette arête de t et en ajoutant l'arête de l'étape i , on obtient ainsi un sous-arbre recouvrant t' de poids inférieur à celui de t :

il s'agit donc d'un sous-arbre recouvrant de poids minimal, contenant le sous-graphe de l'étape i .

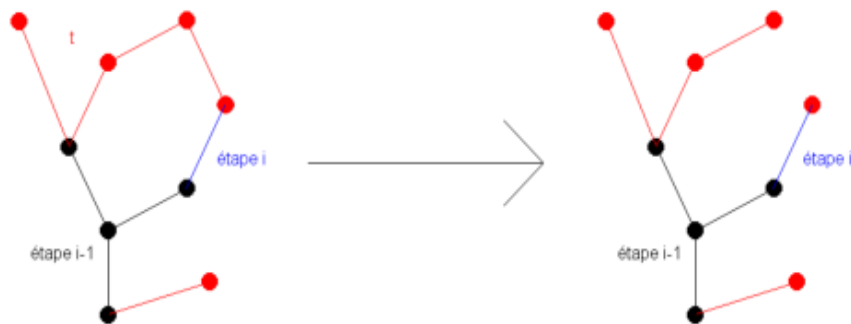


Fig 3.3-Sous-arbre recouvrant de poids minimal.

A l'étape $n - 1$, le sous-graphe obtenu est donc inclus dans un sous-arbre recouvrant de poids minimum et contient autant d'arêtes:c'est le sous-arbre recouvrant. Donc l'algorithme fonctionne.

Algorithme de Prim.

Nous allons montrer qu'à l'étape i , il existe un arbre recouvrant de poids minimal de G contenant le graphe A par récurrence sur i . Si $i = 0$, alors A est formé d'un seul sommet de G et d'aucune arête, donc est inclus dans tous les arbres recouvrant de poids minimal de G . Supposons le résultat vrai à l'étape $i - 1$. Soit alors un arbre recouvrant de poids minimal t , contenant le sous-graphe obtenu à l'étape $i - 1$. S'il contient l'arête ajoutée à l'étape i , cet arbre convient. Sinon, considérons le sous-graphe de G formé de t et de l'arête ajoutée à l'étape i . Il contient n arêtes, car t en contient $n - 1$, donc il contient un cycle. Comme t ne contient pas de cycle, ce cycle contient l'arête ajoutée à l'étape i . Si toutes les autres arêtes de ce cycle sont dans le sous-graphe de l'étape $i - 1$, alors l'arête ajoutée à l'étape i ne relie pas un sommet du graphe de l'étape i à un nouveau sommet : contredit l'algorithme. Donc le cycle contient l'arête de l'étape i et au moins une autre arête qui n'est pas dans le graphe de l'étape $i - 1$. Par le choix de l'algorithme, cette arête est de poids supérieur à celui de l'arête de l'étape i . En retirant cette arête de t et en

ajoutant l'arête de l'étape i , on obtient ainsi un sous-arbre recouvrant t' de poids inférieur à celui de t : il s'agit donc d'un sous-arbre recouvrant de poids minimum, contenant le sous-graphe de l'étape i . A l'étape $n - 1$, le sous-graphe obtenu est donc inclus dans un sous-arbre recouvrant de poids minimal et contient autant d'arêtes : c'est le sous-arbre recouvrant. Donc l'algorithme fonctionne. ■

3.2 Problème du plus court chemin

Le problème de la recherche du plus court chemin dans un graphe se rencontre dans de nombreuses applications. Lorsqu'un chemin existe entre deux sommets dans un graphe, l'être humain se pose rapidement la question non seulement de trouver un tel chemin, mais bien souvent il est intéressé par le plus court chemin possible entre ces deux sommets.

La longueur d'un tel chemin est appelée distance Minimale de x_n à x_y selon le cas. Par abus de langage, et bien que d'unicité ne soit pas nécessairement réalisée, on parle par fois "de plus court (long) chemin " de x_n à x_y , On peut s'intéresser à la recherche d'un plus court chemin dans un graphe:

1. entre deux sommets donnés;
2. d'un sommet à tous les autres;
3. entre tous les couples de sommets.

Il existe de nombreux algorithmes permettant l'obtention d'une solution. Nous nous contenterons simplement ici de décrire les algorithmes les plus classiques, sans explorer toutes les situations. Dans beaucoup de situations, les longueurs sont positives. On utilisera alors l'algorithme de Moore-Dijkstra pour calculer le plus court chemin d'un sommet (arbitrairement le sommet numéro 1) à tous les autres.

Définition 3.2.1 (*algorithme de Dijkstra-Moore*)

Soit un graphe orienté valué $G = (S, A, f)$, d'ordre n et de taille m , et x un sommet de G . L'algorithme de Dijkstra calcule deux matrices de taille $1 \times n$

- $Dist$ matrice des distances telle que $Dist(y) =$ distance optimale de x à y
- $Pred$ matrice des prédécesseurs telle que $Pred(y) =$ prédécesseur de y dans.

Pour le plus court chemin l'algorithme s'écrit

Fonction $[Dist, pred] = DIJKSTRA(G, s)$

Initialisation:

$n =$ nombre de sommets de G .

$Pred =$ tableau des prédécesseurs initialisé à 0.

$Dist =$ tableau des distances initialisé à $+\infty$ (sauf $Dist(s) = 0$)

$W =$ matrice des poids des arcs (∞ si l'arc n'existe pas)

$C = \{1; 2; \dots; n\}$ (liste des sommets restant à traiter)

$D = \emptyset$ (liste des sommets déjà traités)

Traitement :

tant que $C \neq \emptyset$ faire

$x =$ sommet de C le plus proche de s

retirer x de C et le mettre dans D

pour tout sommet $y \in C$ faire

si $Dist(x) + W(x, y) < Dist(y)$

alors modifier $Dist(y)$ et $Pred(y) = x$

fin

fin faire

fin faire.

3.3 Problème d'affectation

3.3.1 La résolution d'un problème d'affectation par la méthode hongroise :

La recherche d'une affectation optimale (maximale ou minimale) par la méthode hongroise est basée sur la notion de zéros indépendants dans une matrice carrée. C'est-à-

dire, contenant n ligne et n colonnes. On appelle zéros "0" indépendants, les zéros qui n'appartiennent ni à la même ligne ni à la même colonne d'une matrice.

Soit A une matrice suivante:

$$A = \begin{pmatrix} [0] & 1 & 2 \\ 0 & 1 & [0] \\ 3 & [0] & 0 \end{pmatrix}$$

Les zéros encadrés ne sont ni sur la même ligne ni sur la même colonne, ils sont dits alors "zéros indépendants"

• **Description globale de la méthode hongroise :**

• Une matrice hongroise est applicable sur une matrice d'affectation carrée. (une matrice de type (n, n)).

• L'algorithme est basé sur la détermination de zéros sur chaque ligne et chaque colonne de la matrice d'affectation .

• L'affectation d'une personne à une tâche peut se faire en choisissant un zéro sur une ligne ou une colonne.

• Pour que chaque personne doit soit affectée à une seule tâche .il faut choisir des zéros indépendants.

• Le but de la méthode hongroise est alors d'obtenir " n " zéros indépendant.

3.3.2 Algorithme de recherche d'une affectation minimale

Soit $A = (a_{ij})$ une matrice d'affectation de type (n, n)

⇒ **Première étape:**

(1) On fait apparaître au moins un zéro sur chaque ligne et chaque colonne de la matrice d'affectation, en retranchant de chaque ligne, le plus petit élément, on fait de même pour chaque colonne.

(2) On choisi sur chaque ligne un zéro en l'encadrant, et on barre les zéros qui se trouvent sur la même ligne et la même colonne.

(3) Si on a obtenu n zéros indépendants, alors l'affectation est optimale. Tout zéro encadré correspond à la valeur a_{ij} dans la matrice d'affectation, alors la personne p_i est affectée à la tâche T_j ; et la valeur de l'affectation minimale est égale à la somme des a_{ij} de la matrice de départ, avec $i = \{1, \dots, n\}$ et $j = \{1, \dots, m\}$. Sinon, on passe à la deuxième étape.

Deuxième étape:

1. Marquer toutes les lignes ne contenant pas de zéro encadré.
2. Marquer toutes les colonnes contenant un zéro barré sur une ligne marquée
3. Marquer toutes les lignes contenant un zéro encadré sur une colonne marquée
4. Revenir à (2) et reprendre le même procédé jusqu'à ne pouvoir plus marquer ni de lignes ni de colonnes
5. On barre toutes les lignes non marquées et toutes les colonnes marquées
6. On considère les éléments non barres de la matrice, soit m le plus petit élément d'entre-eux
 - À chaque élément non barré de la matrice, on retranche l'élément m
 - À chaque élément barre deux fois de la matrice, on rajoute l'élément m
 - On laisse inchangé les éléments barres une seule fois dans la matrice.
1. On intègre les nouveaux éléments calculés dans A , on obtient ainsi une nouvelle matrice,
2. On choisit de nouveaux zéros pour les lignes ne contenant pas de zéro encadré,
3. Si on a n zéros indépendants aller vers (3) de la première étape sinon aller vers (2) de la première étape.

3.3.3 Algorithme de recherche d'une affectation maximale

Soit $A = (a_{ij})$ une matrice d'affectation de type (n, m)

(1) On repère le plus grand élément de la matrice d'affectation A , soit M ce nombre, on lui retranche tous les éléments de la matrice A . On obtient une nouvelle matrice $B = (b_{ij})$ de type (n, m) telle que:

$$b_{ij} = \begin{cases} M - a_{ij} & \text{Si } a_{ij} \neq \infty \\ \infty & \text{Si } a_{ij} = \infty \end{cases}$$

(2) Si $n = m$; aller vers (3)

Si $n \neq m$; on complète la matrice B par des lignes ou des colonnes de zéros ; aller vers

(3)

(3) On applique l'algorithme de minimisation à la matrice B .

3.4 Problème d'ordonnancement

Un problème d'ordonnancement se pose lorsqu'il s'agit d'organiser dans le temps l'exécution d'un ensemble de tâches. De tels problèmes se rencontrent dans différents contextes tels que la gestion de grands projets, la conduite d'ateliers de fabrication, l'organisation d'activités de service, ...

La résolution d'un problème d'ordonnancement consiste à placer dans le temps des activités ou tâches, compte tenu de contraintes temporelle (délais, contraintes de précedence,...) et de contraintes portant sur l'utilisation et la disponibilité des ressources requises par les tâches. Un ordonnancement décrit l'exécution des tâches en précisant leurs dates de début et leurs dates de fin et l'allocation des ressources au cours du temps, et vise à satisfaire un ou plusieurs objectifs. Les problèmes d'ordonnancement que nous considérons dans cette thèse ne sont soumis qu'à des contraintes temporelles. Nous ne traitons pas les contraintes de ressources (nous supposons que celles-ci sont illimitées et disponibles à tout moment). Les méthodes d'ordonnancement les plus connues sont : le diagramme de Gantt, la méthode MPM et le PERT. Nous résumons dans les sections suivantes ces trois techniques d'ordonnancement.

3.4.1 Le diagramme de Gantt

Le diagramme de Gantt est un outil utilisé en ordonnancement et en gestion de projet et permettant de visualiser dans le temps les diverses tâches composant un projet. Il s'agit d'une représentation d'un graphe connexe, valué et orienté, qui permet de représenter graphiquement l'avancement du projet. L'objectif de diagramme de Gantt : planifier de façon optimale ainsi que communiquer sur le planning établi et les choix qu'il impose. permet de déterminer les dates de réalisation d'un projet, identifier les marges existantes sur certaines tâches et de visualiser d'un seul coup d'œil le retard ou l'avancement des travaux. Dans un diagramme de Gantt on représente :

- en abscisse les unités de temps (exprimées en mois, en semaine ou en jours);
- en ordonnée les différents postes de travail (ou les différentes tâches).

La figur 3.1 représente un diagramme de Gantt où chaque colonne représente une unité de temps, les traits épais représentent les durées d'exécution prévues des tâches et les traits pointillés représentent le déroulement d'exécution. Par exemple, la tâche *B*, qui dure 5 unités de temps, ne peut commencer son exécution qu'après la fin de la tâche *A* et elle peut s'exécuter en même temps que la tâche *C*.

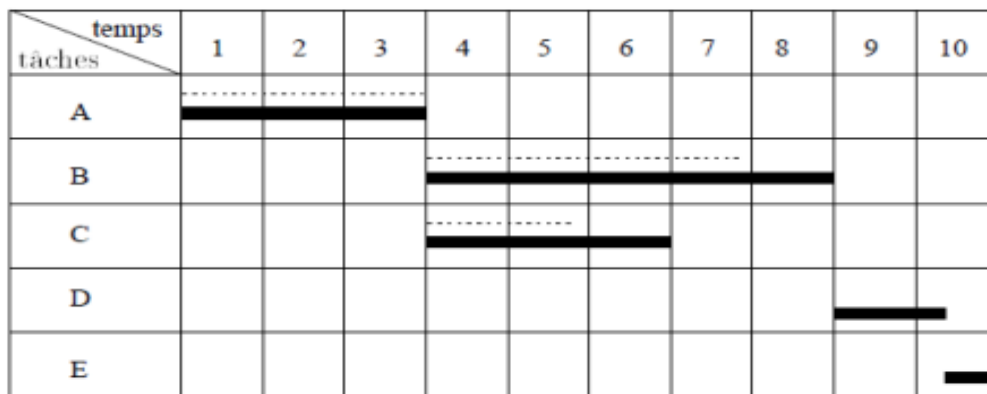


Fig 3.4—Un exemple du diagramme de Gantt

Le chemin critique est formé d'une succession de tâches sur le chemin le plus long en terme de durées (*A*, *B*, *D*, *E* dans l'exemple). Il est appelé chemin critique parce que tout

retard pris sur l'une des tâches de ce chemin entraîne du retard dans l'achèvement du projet. Le diagramme de Gantt permet de déterminer la date de réalisation d'un projet et d'identifier les marges existantes sur certaines tâches (avec une date de début au plus tôt et une date de fin au plus tard). Son point faible est que son application est limitée à des problèmes particuliers.

3.4.2 Méthode française MPM (Méthode Potentiel Métra)

Dans la méthode des potentiels-métra, le problème est représenté sous forme d'un graphe tel que les tâches sont représentées par des nœuds et les contraintes de succession par des arcs. À chaque nœud sont associées une date de début au plus tôt et une date de fin au plus tard. À chaque arc est associé un délai d'attente entre les tâches.

La figure 3.2 représente un exemple de la méthode des potentiels-métra. La date de début au plus tôt d'une tâche dépend de la date de fin des tâches qui la précèdent. La tâche DEBUT est initialisée avec une date de début au plus tôt égale à zéro. Cette méthode permet de déterminer la date de réalisation d'un projet ainsi que la date de début et de fin de chaque tâche mais elle est incapable de résoudre des problèmes qui prennent en compte plus de contraintes telles que l'incertitude et les coûts d'exécution des tâches.

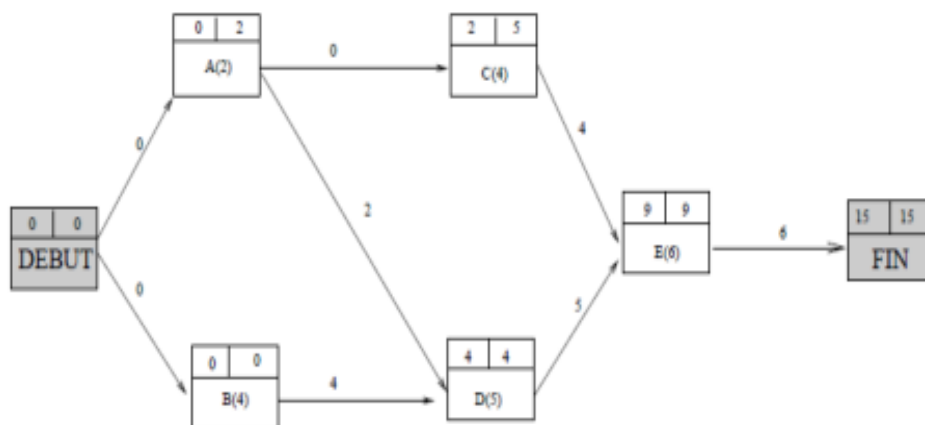


Fig. 3.5— Un exemple d'un graphe MPM.

3.4.3 Méthode américaine: PERT

La méthode PERT consiste à mettre en ordre sous forme de réseau plusieurs tâches qui grâce à leurs dépendances et à leur chronologie permettent d'avoir un produit fini. Elle représente le problème sous forme d'un graphe tel que les tâches sont représentées par un arc auquel on associe un nombre entre parenthèses qui représente la durée de la tâche. Un nœud représente la fin d'une ou de plusieurs tâches.

Exemple 3.4.1 de la méthode PERT

La figure 3.3 représente un graphe PERT. La construction d'un tel graphe se fait par niveau : le niveau 1 contient les tâches sans antécédents, le niveau i contient les tâches dont les antécédents sont exclusivement du niveau $i - 1$. La date de début au plus tôt d'une tâche dépend des dates de fin des tâches qui la précèdent. Cette méthode permet de déterminer la date de début et de fin de chaque tâche ainsi que le chemin critique c'est-à-dire un ensemble d'activités tel que tout retard dans leur exécution provoquerait un retard de la fin du projet. Par contre, elle ne présente pas d'échelle calendaire comme dans le cas de diagramme de Gantt.

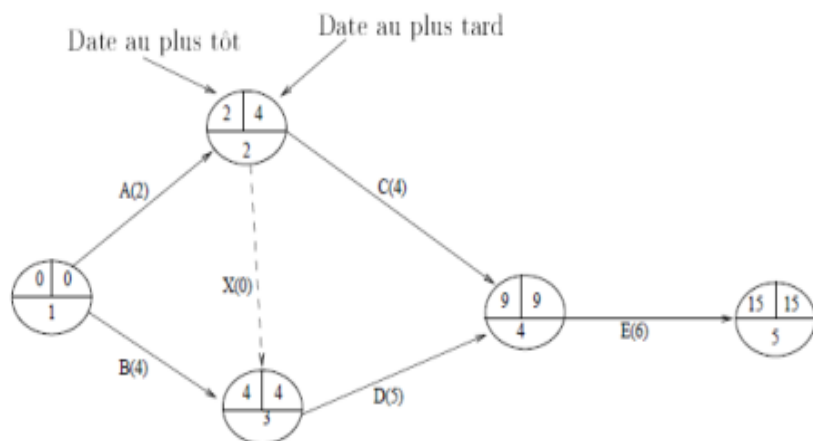


Fig. 3.3— Un exemple d'un graphe PERT.

Il existe plusieurs problèmes pour lesquels on peut utiliser des algorithmes afin d'optimiser une certaine propriété. On va considérer les cas suivants :

4.1 Réseaux téléphoniques

4.1.1 Présentation de la problématique

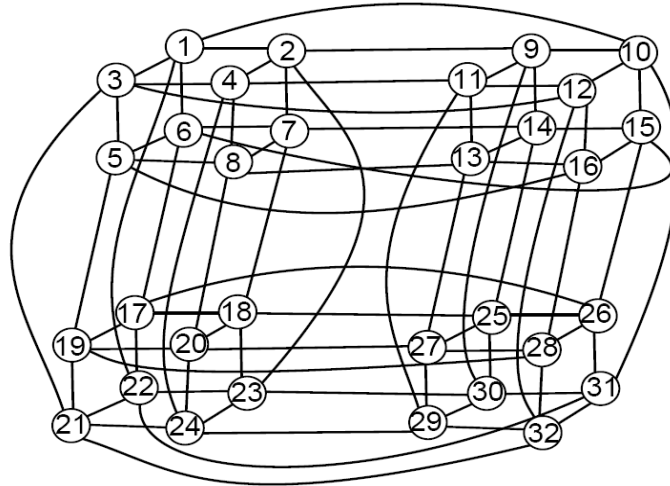
Soit une entreprise de télécommunication qui veut tester la confidentialité des services qu'elle offre, c'est-à-dire trouver la probabilité qu'un message ou un appel entre ses abonnés ne soit pas intercepté par d'autres personnes étrangères.

4.1.2 Modélisation du problème sous forme d'un réseau

Pour véhiculer de l'information entre un groupe de personnes, on modélise sous forme d'un graphe G à n sommets, tel que chaque sommet représente une personne et chaque arête représente la communication entre deux personnes.

Les séquences des messages transmis forment un arbre couvrant dans G . Donc le problème revient à trouver un arbre couvrant de poids minimum dans G qui minimise la fonction $(1 - \prod(1 - P_{ij}))$ tel que P_{ij} est la probabilité d'interception d'un message transmis entre x_i et x_j .

Soit un graphe $G = (X, E)$ tel que $X = x_1, x_2, \dots, x_n$, et $x_i x_j \in E(G)$ si une communication entre x_i et x_j est possible (de x_i à x_j et vice versa).



Résolution du problème

Application de l'algorithme de Kruskal Initialisation:

On considère l'ordonnancement des arêtes du graphe selon les probabilités croissantes dans le tableau ci-dessous:

i	1	2	3	4	5	6	7	8	9	10
e_i	(16.28)	(27.29)	(12.32)	(23.30)	(4.11)	(30.31)	(2.9)	(4.24)	(11.29)	(18.25)
$P(e_i)$	0.05	0.05	0.06	0.06	0.07	0.07	0.09	0.09	0.09	0.09

i	11	12	13	14	15	16	17	18	19	20
e_i	(21.22)	(8.20)	(7.18)	(3.4)	(6.17)	(5.19)	(25.30)	(28.32)	(29.32)	(1.2)
$P(e_i)$	0.09	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.10	0.11

i	21	22	23	24	25	26	27	28	29	30
e_i	(26.31)	(8.13)	(3.21)	(13.27)	(19.20)	(9.11)	(10.15)	(6.15)	(13.14)	(1.10)
$P(e_i)$	0.11	0.12	0.13	0.13	0.13	0.14	0.14	0.15	0.15	0.16

i	31	32	33	34	35	36	37	38	39	40
e_i	(25.26)	(7.8)	(9.30)	(20.27)	(17.22)	(19.21)	(23.24)	(1.22)	(7.14)	(7.6)
$P(e_i)$	0.16	0.17	0.17	0.17	0.17	0.17	0.18	0.20	0.20	0.20

i	41	42	43	44	45	46	47	48	49	50
e_i	(9.14)	(10.12)	(10.31)	(12.16)	(13.16)	(14.15)	(18.23)	(20.24)	(15.26)	(17.18)
$P(e_i)$	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.21	0.21

i	51	52	53	54	55	56	57	58	59	60
e_i	(17.26)	(22.31)	(1.3)	(2.4)	(2.23)	(3.12)	(6.5)	(5.16)	(14.25)	(27.28)
$P(e_i)$	0.22	0.22	0.23	0.23	0.24	0.24	0.25	0.25	0.25	0.25

i	61	62	63	64	65	66	67	68	69	70
e_i	(24.29)	(21.24)	(1.6)	(3.4)	(5.8)	(9.10)	(11.12)	(15.16)	(26.28)	(19.28)
$P(e_i)$	0.26	0.27	0.30	0.30	0.30	0.30	0.30	0.30	0.30	0.30

i	71	72	73	74	75	76	77	78	79	80
e_i	(17.19)	(4.8)	(22.23)	(2.7)	(21.32)	(11.13)	(25.27)	(31.32)	(18.20)	(30.29)
$P(e_i)$	0.30	0.31	0.32	0.33	0.34	0.40	0.40	0.40	0.40	0.50

Soit $W = \emptyset ; i = 1; m = 80$.

1^{re} **Itération:**

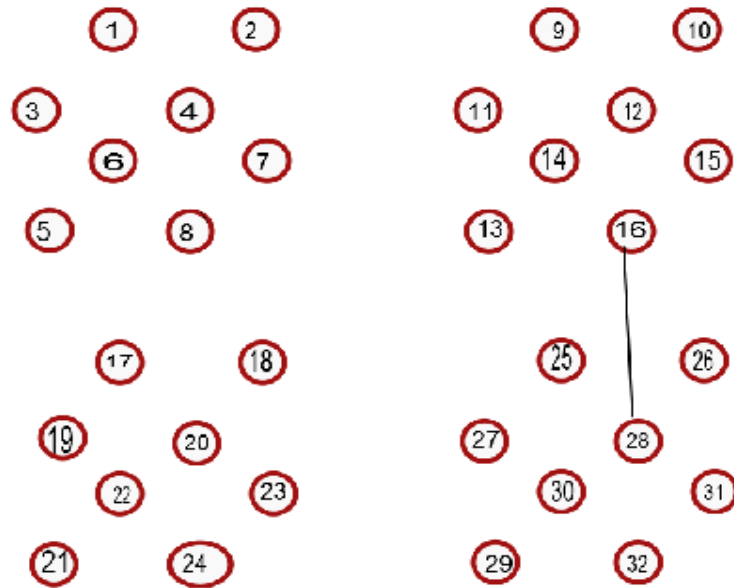
On a : $e_1 = (16, 28)$;

Soit : $W = W \cup \{e_1\} = W \cup \{(16, 28)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28)\}$, et $|W| = 1$

On a : $i \neq m$, alors on pose $i = i + 1 = 2$.



2^{me} Itération:

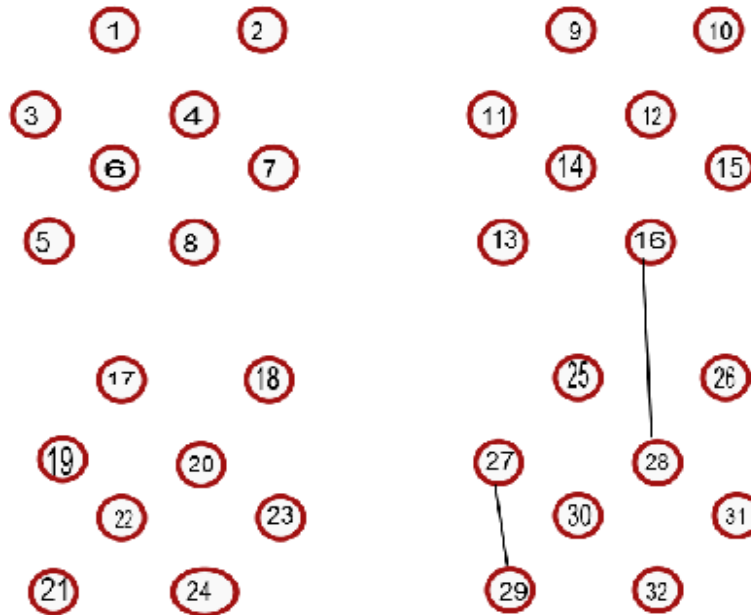
On a : $e_2 = (27, 29)$;

Soit : $W = W \cup \{e_2\} = W \cup \{(27, 29)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28); (27, 29)\}$, et $|W| = 2$

On a : $i \neq m$, alors on pose $i = i + 1 = 3$.

**3^{me} Itération:**

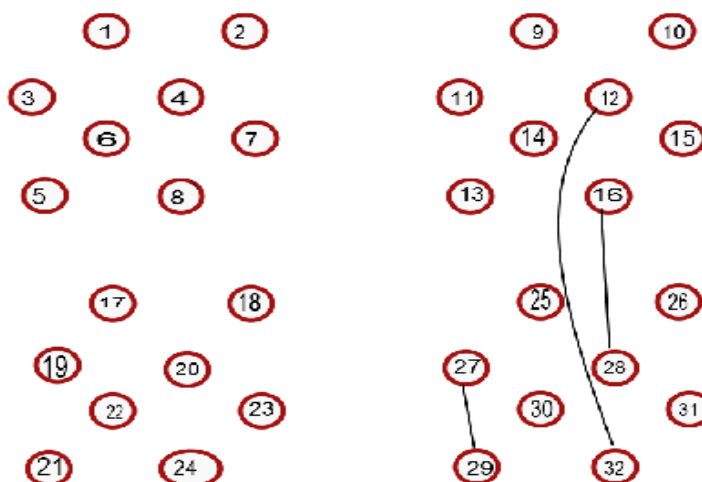
On a : $e_3 = (12, 32)$;

Soit : $W = W \cup \{e_3\} = W \cup \{(12, 32)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28); (27, 29); (12, 32)\}$, et $|W| = 3$.

On a : $i \neq m$, alors on pose $i = i + 1 = 4$.



4^{me} Itération:

On a : $e_4 = (23, 30)$;

Soit : $W = W \cup \{e_4\} = W \cup \{(23, 30)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle ,on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30)\}$, et $|W| = 4$.

On a : $i \neq m$, alors on pose $i = i + 1 = 5$.

5^{me} Itération:

On a : $e_5 = (4, 11)$;

Soit : $W = W \cup \{e_5\} = W \cup \{(4, 11)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle,on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11)\}$, et $|W| = 5$.

On a : $i \neq m$, alors on pose $i = i + 1 = 6$.

6^{me} Itération:

On a : $e_6 = (30, 31)$;

Soit : $W = W \cup \{e_6\} = W \cup \{(30, 31)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle,on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31)\}$, et $|W| = 6$

On a : $i \neq m$, alors on pose $i = i + 1 = 7$.

7^{me} Itération:

On a : $e_7 = (3, 12)$;

Soit : $W = W \cup \{e_7\} = W \cup \{(2, 9)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle,on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9)\}$, et $|W| = 7$

On a : $i \neq m$, alors on pose $i = i + 1 = 8$.

8^{me} Itération:

On a : $e_8 = (4, 24)$;

Soit : $W = W \cup \{e_8\} = W \cup \{(4, 24)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle,on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24)\}$, et $|W| = 8$

On a : $i \neq m$, alors on pose $i = i + 1 = 9$.

9^{me} Itération:

On a : $e_9 = (11, 29)$;

Soit : $W = W \cup \{e_9\} = W \cup \{(11, 29)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29)\}$,

et $|W| = 9$.

On a : $i \neq m$, alors on pose $i = i + 1 = 10$.

10^{me} Itération:

On a : $e_{10} = (18, 25)$;

Soit : $W = W \cup \{e_{10}\} = W \cup \{(18, 25)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25)\}$,

et $|W| = 10$.

On a : $i \neq m$, alors on pose $i = i + 1 = 11$.

11^{me} Itération:

On a : $e_{11} = (21, 22)$;

Soit : $W = W \cup \{e_{11}\} = W \cup \{(21, 22)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \{(16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22)\}$,

et $|W| = 11$.

On a : $i \neq m$, alors on pose $i = i + 1 = 12$.

12^{me} Itération:

On a : $e_{12} = (8, 20)$;

Soit : $W = W \cup \{e_{12}\} = W \cup \{(8, 20)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20) \end{array} \right\}$,

et $|W| = 12$.

On a : $i \neq m$, alors on pose $i = i + 1 = 13$.

13^{me} Itération:

On a : $e_{13} = (7, 18)$;

Soit : $W = W \cup \{e_{13}\} = W \cup \{(7, 18)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18) \end{array} \right\},$$

et $|W| = 13$.

On a : $i \neq m$, alors on pose $i = i + 1 = 14$.

14^{me} Itération:

On a : $e_{14} = (3, 4)$;

Soit : $W = W \cup \{e_{14}\} = W \cup \{(3, 4)\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4) \end{array} \right\},$$

et $|W| = 14$.

On a : $i \neq m$, alors on pose $i = i + 1 = 15$.

15^{me} Itération:

On a : $e_{15} = (6, 17)$;

Soit : $W = W \cup \{e_{15}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17) \end{array} \right\},$$

et $|W| = 15$.

On a : $i \neq m$, alors on pose $i = i + 1 = 16$.

16^{me} Itération:

On a : $e_{16} = (5, 19)$;

Soit : $W = W \cup \{e_{16}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19) \end{array} \right\},$$

et $|W| = 16$.

On a : $i \neq m$, alors on pose $i = i + 1 = 17$.

17^{me} Itération:

On a : $e_{17} = (25, 30)$;

Soit : $W = W \cup \{e_{17}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30) \end{array} \right\},$$

et $|W| = 17$.

On a : $i \neq m$, alors on pose $i = i + 1 = 18$.

18^{me} Itération:

On a : $e_{18} = (28, 32)$;

Soit : $W = W \cup \{e_{18}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32) \end{array} \right\},$$

et $|W| = 18$.

On a : $i \neq m$, alors on pose $i = i + 1 = 19$.

19^{me} Itération:

On a : $e_{19} = (29, 32)$;

Soit : $W = W \cup \{e_{19}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32) \end{array} \right\},$$

et $|W| = 19$

On a : $i \neq m$, alors on pose $i = i + 1 = 20$.

20^{me} Itération:

On a : $e_{20} = (1, 2)$;

Soit : $W = W \cup \{e_{20}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2) \end{array} \right\},$$

et $|W| = 20$

On a : $i \neq m$, alors on pose $i = i + 1 = 21$.

21^{me} Itération:

On a : $e_{21} = (26, 31)$;

Soit : $W = W \cup \{e_{21}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31) \end{array} \right\},$$

et $|W| = 21$.

On a : $i \neq m$, alors on pose $i = i + 1 = 22$.

22^{me} Itération:

On a : $e_{22} = (8, 13)$;

Soit : $W = W \cup \{e_{22}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); \\ (8, 13). \end{array} \right\},$$

et $|W| = 22$.

On a : $i \neq m$, alors on pose $i = i + 1 = 23$.

23^{me} Itération:

On a : $e_{23} = (3, 21)$;

Soit : $W = W \cup \{e_{23}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); \\ (8, 13); (3, 21). \end{array} \right\},$$

et $|W| = 23$

On a : $i \neq m$, alors on pose $i = i + 1 = 24$.

24^{me} Itération:

On a : $e_{24} = (13, 27)$;

Soit : $W = W \cup \{e_{24}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13) \\ ; (3, 21); (13, 27). \end{array} \right\}$$

et $|W| = 24$.

On a : $i \neq m$, alors on pose $i = i + 1 = 25$.

25^{me} Itération:

On a : $e_{25} = (19, 20)$;

Soit : $W = W \cup \{e_{25}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); \\ (8, 13); (3, 21); (13, 27); (19, 20). \end{array} \right\},$$

et $|W| = 25$.

On a : $i \neq m$, alors on pose $i = i + 1 = 26$.

26^{me} Itération:

On a : $e_{26} = (9, 11)$;

Soit : $W = W \cup \{e_{26}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); \\ (21, 22); (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); \\ (8, 13); (3, 21); (13, 27); (19, 20); (9, 11). \end{array} \right\},$$

et $|W| = 26$.

On a : $i \neq m$, alors on pose $i = i + 1 = 27$.

27^{me} Itération:

On a : $e_{27} = (10, 15)$;

Soit : $W = W \cup \{e_{27}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15). \end{array} \right\}$$

et $|W| = 27$.

On a : $i \neq m$, alors on pose $i = i + 1 = 28$.

28^{me} Itération:

On a : $e_{28} = (6, 15)$;

Soit : $W = W \cup \{e_{28}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15); (6, 15). \end{array} \right\}$$

et $|W| = 28$.

On a : $i \neq m$, alors on pose $i = i + 1 = 29$

29^{me} Itération:

On a : $e_{29} = (13, 14)$;

Soit : $W = W \cup \{e_{29}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15); (6, 15); (13, 14). \end{array} \right\}$$

et $|W| = 29$.

On a : $i \neq m$, alors on pose $i = i + 1 = 30$.

30^{me} Itération:

On a : $e_{30} = (1, 10)$;

Soit : $W = W \cup \{e_{30}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15); (6, 15); (13, 14); (1, 10). \end{array} \right\}$$

et $|W| = 30$.

On a : $i \neq m$, alors on pose $i = i + 1 = 31$.

31^{me} Itération:

On a : $e_{31} = (25, 26)$;

Soit : $W = W \cup \{e_{31}\}$;

Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15); (6, 15); (13, 14); (1, 10); (25, 26). \end{array} \right\}$$

et $|W| = 31$.

On a : $i \neq m$, alors on pose $i = i + 1 = 32$.

32^{me} Itération:

On a : $e_{32} = (7, 8)$;

Soit : $W = W \cup \{e_{32}\}$;

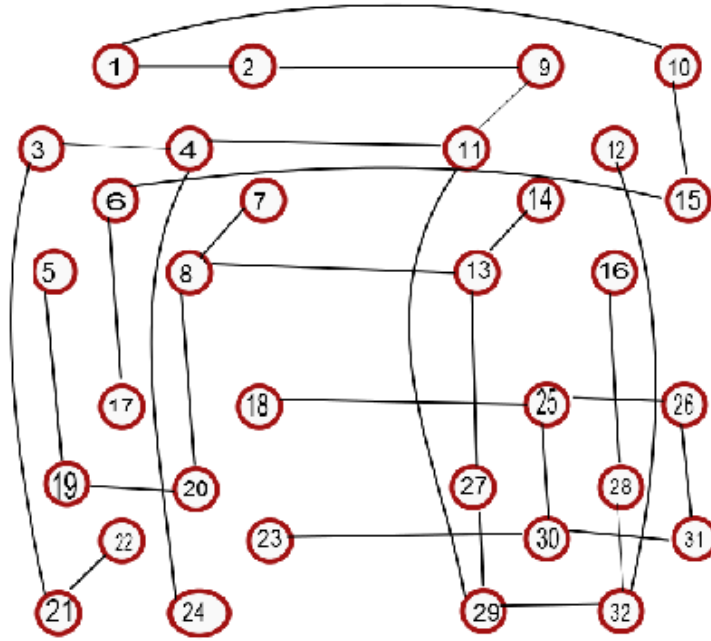
Le graphe $G = (X, W)$ ne contient pas de cycle, on pose alors:

$$W = \left\{ \begin{array}{l} (16, 28); (27, 29); (12, 32), (23, 30); (4, 11); (30, 31); (2, 9); (4, 24); (11, 29); (18, 25); (21, 22); \\ (8, 20); (7, 18); (3, 4); (6, 17); (5, 19); (25, 30); (28, 32); (29, 32); (1, 2); (26, 31); (8, 13); \\ (3, 21); (13, 27); (19, 20); (9, 11); (10, 15); (6, 15); (13, 14); (1, 10); (25, 26). \end{array} \right\}$$

et $|W| = 32$.

On a : $i \neq m$, alors on pose $i = i + 1 = 33$.

terminer. $|W| = n - 1 = 32 - 1 = 31$.



L'arbre de poids (probabilités) minimum est représenté par le graphe $A = (X; W)$. La probabilité minimale totale est :

$$P(W) = 1 - \prod_1^{32} (1 - P_{ij})$$

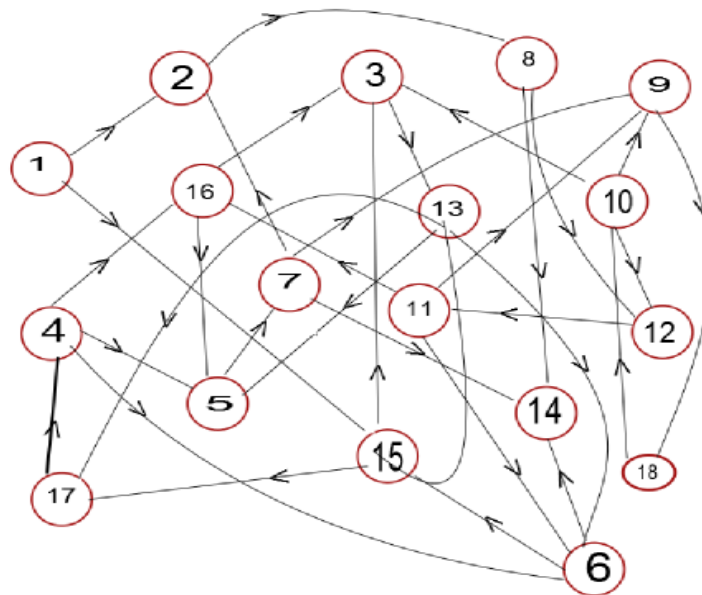
$$P(W) = 1 - [(1 - P(16, 28)) \times (1 - P(27, 29)) \times (1 - P(12, 32)) \times (1 - P(23, 30)) \times (1 - P(4, 11)) \times (1 - P(30, 31)) \times (1 - P(2, 9)) \times (1 - P(4, 24)) \times (1 - P(11, 29)) \times (1 - P(18, 25)) \times (1 - P(21, 22)) \times (1 - P(8, 20)) \times (1 - P(7, 18)) \times (1 - P(3, 4)) \times (1 - P(6, 17)) \times (1 - P(5, 19)) \times (1 - P(25, 30)) \times (1 - P(28, 32)) \times (1 - P(29, 32)) \times (1 - P(1, 2)) \times (1 - P(26, 31)) \times (1 - P(8, 13)) \times (1 - P(3, 21)) \times (1 - P(13, 27)) \times (1 - P(19, 20)) \times (1 - P(9, 11)) \times (1 - P(10, 15)) \times (1 - P(6, 15)) \times (1 - P(13, 14)) \times (1 - P(1, 10)) \times (1 - P(25, 26)) \times (1 - P(7, 8))]$$

$$P(W) = 1 - [(1 - 0,05) \times (1 - 0,05) \times (1 - 0,06) \times (1 - 0,06) \times (1 - 0,07) \times (1 - 0,07) \times (1 - 0,09) \times (1 - 0,09) \times (1 - 0,09) \times (1 - 0,05) \times (1 - 0,09) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,10) \times (1 - 0,11) \times (1 - 0,11) \times (1 - 0,12) \times (1 - 0,13) \times (1 - 0,13) \times (1 - 0,13) \times (1 - 0,13) \times (1 - 0,14) \times (1 - 0,14) \times (1 - 0,15) \times (1 - 0,15) \times (1 - 0,16) \times (1 - 0,16) \times (1 - 0,17)] = 0,975.$$

4.1.3 Problème de plus long chemin

Application de l'algorithme de dijkstra

Dans une région en proie à un conflit ou à catastrophe naturelle, un convoi doit être acheminé d'une ville s à une ville t . Le réseau routier de la région est donné par un graphe valué $G = (X, A, P)$. X désigne l'ensemble des villes et A l'ensemble des tronçons de route entre villes. pour tout arc (i, j) . La valuation $p(i, j)$ est la probabilités de parcourir sans dommage la route de i à j . on souhaite déterminer un itinéraire maximisant la probabilité pour le convoi d'arriver sans dommage en t , c'est-à-dire un chemin de fiabilité maximale.



Initialisation:

On considère l'ordonnancement des arêtes du graphe selon les probabilités qui représentent dans le tableau ci-dessous:

i	1	2	3	4	5	6	7	8	9	10
e _i	(1.2)	(1.15)	(2.8)	(3.13)	(4.16)	(4.5)	(4.6)	(5.7)	(6.14)	(6.15)
P(e _i)	0.9	0.1	1	1	0.7	0.1	0.2	1	0.3	0.7

i	11	12	13	14	15	16	17	18	19	20
e _i	(7.9)	(7.14)	(7.2)	(8.14)	(8.12)	(9.18)	(10.3)	(10.14)	(10.9)	(10.12)
P(e _i)	0.5	0.4	0.1	0.2	0.8	1	0.4	0.1	0.3	0.2

i	21	22	23	24	25	26	27	28	29	30
e _i	(11.16)	(11.6)	(11.9)	(12.4)	(13.15)	(13.17)	(13.5)	(13.6)	(15.3)	(15.17)
P(e _i)	0.3	0.5	0.2	1	0.2	0.3	0.2	0.3	0.3	0.7

i	31	32	33	34	35	36	37	38	39	40
e _i	(16.3)	(16.1)	(17.4)	(18.10)	(19.15)	(19.5)	(19.6)	(19.17)	()	()
P(e _i)	0.3	0.7	1	1	0.2	0.2	0.3	0.3		

$S = 1, \pi(1) = 1, \bar{S} = \{2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18\}$.

Les successeurs de 1 sont 2 et 15.

1^{re} Itération :

$i = 2$ car $\pi(2) = \max\{0.9, 0.1\} = 0.9, \pi(2) = 0.9, S = \{1, 2\}$;

$\bar{S} = \{3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18\}$, Successeur de 2 est 8.

2^{me} Itération

$i = 8$ car $\pi(8) = \pi(2) * \gamma(2, 8) = 0.9 * 1 = 0.9$. Donc $\pi(8) = 0.9, S = \{1, 2, 8\}$

$\bar{S} = \{3, 4, 5, 6, 7, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18\}$

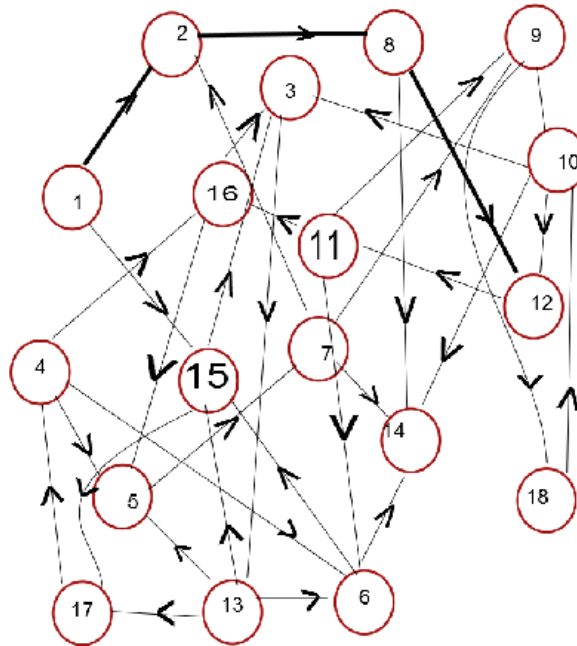
Les successeurs de 8 sont 12 et 14.

3^{me} Itération

$i = 12$ car $\pi(12) = \pi(8) * \gamma(8, 12) = 0.9 * 0.8 = 0.72$. Donc $\pi(12) = 0.72, S = \{1, 2, 8, 12\}$

$\bar{S} = \{3, 4, 5, 6, 7, 9, 10, 11, 13, 14, 15, 16, 17, 18\}$

successeurs de 12 est 11.

**4^{me} Itération**

$i = 11$ car $\pi(11) = \pi(12) * \gamma(12, 11) = 0.72 * 1 = 0.72$. Donc $\pi(11) = 0.72$,

$S = \{1, 2, 8, 12, 11\}$ et $\bar{S} = \{3, 4, 5, 6, 7, 9, 10, 13, 14, 15, 16, 17, 18\}$

Les successeurs de 11 sont 6, 9 et 16.

5^{me} Itération

$i = 6$ car $\pi(6) = \pi(11) * \gamma(11, 6) = 0.72 * 0.5 = 0.36 > (\pi(11) = \pi(11) * \gamma(11, 9) \text{ et } \pi(11) * \gamma(11, 16))$. Donc $\pi(11) = 0.36$, $S = \{1, 2, 8, 12, 11, 6\}$,

$\bar{S} = \{3, 4, 5, 7, 9, 10, 13, 14, 15, 16, 17, 18\}$

Les successeurs de 6 sont 14 et 15.

6^{me} Itération

$i = 15$ car $\pi(15) = \pi(6) * \gamma(6, 15) = 0.36 * 0.7 = 0.252 > \pi(6) * \gamma(6, 14) = 0.108$.

Donc $\pi(15) = 0.252$, $S = \{1, 2, 8, 12, 11, 6, 15\}$,

$\bar{S} = \{3, 4, 5, 7, 9, 10, 13, 14, 16, 17, 18\}$,

Les successeurs de 15 sont 3 et 17.

7^{me} Itération

$i = 17$ car $\pi(17) = \pi(15) * \gamma(15, 17) = 0.252 * 0.7 = 0.1764 > (\pi(15) * \gamma(15, 3) = 0.0756)$.

Donc $\pi(15) = 0.252$, $S = \{1, 2, 8, 12, 11, 6, 15, 17\}$,

$\bar{S} = \{3, 4, 5, 7, 9, 10, 13, 14, 16, 18\}$,

successeurs de 17 est 4.

8^{me} Itération

$i = 4$ car $\pi(4) = \pi(17) * \gamma(17, 4) = 0.1764 * 1 = 0.1764$. Donc $\pi(4) = 0.1764$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4\}$,

$\bar{S} = \{3, 5, 7, 9, 10, 13, 14, 16, 18\}$,

Les successeurs de 4 sont 5, 6 et 16.

9^{me} Itération

$i = 16$ car $\pi(16) = \pi(4) * \gamma(4, 16) = 0.1764 * 0.7 = 0.12348 > (\pi(4) * \gamma(4, 5) = 0.01764)$

Dans cette itération on supprime l'arc (4,6) car il contient un cycle.

Donc $\pi(16) = 0.12348$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16\}$,

$\bar{S} = \{3, 5, 7, 9, 10, 13, 14, 18\}$,

Les successeurs de 16 sont 3 et 5.

10^{me} Itération

$i = 5$ car $\pi(5) = \pi(16) * \gamma(16, 5) = 0.12348 * 0.7 = 0.086436 > (\pi(16) * \gamma(16, 3) = 0.037044)$

Donc $\pi(5) = 0.086436$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5\}$,

$\bar{S} = \{3, 7, 9, 10, 13, 14, 18\}$,

Successeurs de 5 est 7.

11^{me} Itération

$i = 7$ car $\pi(7) = \pi(5) * \gamma(5, 7) = 0.086436 * 1 = 0.086436$.

Donc $\pi(7) = 0.086436$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7\}$,

$\bar{S} = \{3, 9, 10, 13, 14, 18\}$,

Les successeurs de 7 sont 2, 9 et 14.

12^{me} Itération

$i = 9$ car $\pi(9) = \pi(7) * \gamma(7, 9) = 0.086436 * 0.5 = 0.043218 > (\pi(7) * \gamma(7, 14) = 0.0345744)$.

Dans cette itération on supprime l'arc (7, 2) car il contient un cycle.

Donc $\pi(9) = 0.043218$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9\}$,

$\bar{S} = \{3, 10, 13, 14, 18\}$,

Successeurs de 9 est 18.

13^{me} Itération

$i = 18$ car $\pi(18) = \pi(9) * \gamma(9, 18) = 0.043218 * 1 = 0.043218$.

Donc $\pi(18) = 0.043218$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9, 18\}$,

$\bar{S} = \{3, 10, 13, 14\}$,

Successeurs de 18 est 10.

14^{me} Itération

$i = 10$ car $\pi(10) = \pi(18) * \gamma(18, 10) = 0.043218 * 1 = 0.043218$.

Donc $\pi(10) = 0.043218$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9, 18, 10\}$,

$\bar{S} = \{3, 13, 14\}$,

Les successeurs de 10 sont 3, 9 et 12..

15^{me} Itération

$i = 3$ car $\pi(3) = \pi(10) * \gamma(10, 3) = 0.043218 * 0.4 = 0.043218 > \pi(10) * \gamma(10, 14) = 0.0043218$.

Dans cette itération on supprime l'arc (10,9) et l'arc (10,12) car il contient un cycle.

Donc $\pi(3) = 0.0172872$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9, 18, 10, 3\}$,

$\bar{S} = \{13, 14\}$,

Successeurs de 3est 13.

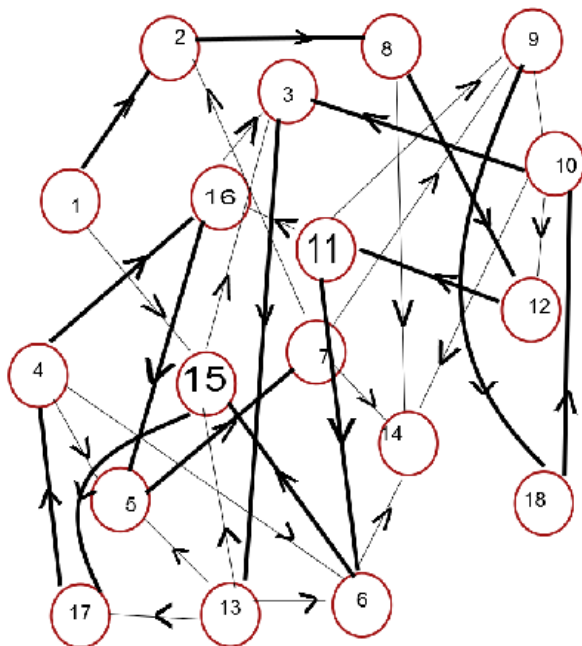
16^{me} Itération

$i = 13$ car $\pi(13) = \pi(3) * \gamma(3, 13) = 0.0172872 * 1 = 0.0172872$.

Donc $\pi(13) = 0.0172872$, $S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9, 18, 10, 3, 13\}$,

$\bar{S} = \{14\}$,

Les successeurs de 13sont 5, 6, 15 et 17.

.17^{me} Itération

$i = 13$, Donc le chemin pour lequel le produit des probabilités soit maximal est:

$$S = \{1, 2, 8, 12, 11, 6, 15, 17, 4, 16, 5, 7, 9, 18, 10, 3, 13\}.$$

Bibliographie

- [1] A. HENRY-LABORDERE << Cours de recherche opérationnel >> I.E.N.P.C. Paris.
- [2] Amour. Mémoire et Modélisation Sous la direction du prof. Christian Mazza.2010.
- [3] A.GUELAIN. Analyse probabiliste des structures en utilisant la praticité. Mémoire de Magister.
- [4] Bertrand Hauchecome, Daniel Surrateau. Des mathématiciens de A Z.
- [5] C. BERGE << Graphes et hypergraphe. Monographie universitaire de mathématiques >>.Dunod, Paris, 1970.p –502,(511.5BER : CD PS).1970.
- [6] C. PRINS << Algorithmes de graphes. Eyrolles >> Paris.1994.
- [7] C. Solnon, Théorie des graphes et optimisation dans les graphes.
- [8] C.Berge : Graphes et Hypergraphe.
- [9] D. WEST << Introduction to Graph Theory >> 2nd edition. Prentice Hall. 2001.
- [10] Declic Terminale ES << enseignement obligatoire et option >>. Hachette Education. 2006.
- [11] Etienne André, LS V-ENS, Cachant et CNRS France.
- [12] F. DROESBEKE, M. HALLIN, C. LEFEVRE, << Les graphes par l'exemple >>, Ellipse, 1970.

-
- [13] J.MICHEL HELARY, PEDRONO << Recherche Opérationnelle: exercices corrigés : graphe et programmation dynamique >> . Hermann, Paris, 1983.p.462 ISBN 2-7056-5954-4 (003.076, HEL: CD PS). 1983.
- [14] J.MICHEL HELARY, << Algorithmique des graphe >> 2004.
- [15] [http:// CreativeCommons.org/licenses/by-nc-sa/3.0/deed.fr](http://CreativeCommons.org/licenses/by-nc-sa/3.0/deed.fr).
- [16] [http:// mat.gsia.cmu.edu](http://mat.gsia.cmu.edu).
- [17] T.TOMESCU. << Problems in Combinatorics and Graph Theory >>. John Wiley et Sons. New York. 1985. Translated from Romanian by R.A.Melter. p 335 (511.6 TOM: CD PS). 1985.
- [18] J.Renaud, plani cation op erationnelle de projet, 2000.
- [19] << Les graphes Stochastique-Dynamique >>. L a cote 003/25, Université de Bejaia.
- [20] M.N.S SWAMY K.THULASIRAMAN. << Graphs: Theory and Algorithms >> John Wiley. New York. 1992. p 460 (511.5 THU: CD PS) 1992.
- [21] M.GONDRON.M.MI NOUX << Graphes et algorithms >> Eyrelles, Paris, 1984.
- [22] M.SAKAROVITCH << Optimisation Combinatoire. Tome 1 : Graphes et programmation linéaire Coll Enseignement des sciences >> Hermann. Paris. 1984. XIII. 294. PISSN 0768.0341.31ISBN
- [23] M. Sopena El ements de th eorie des graphes .
- [24] N.Akkouche M.Bentoubache. application de la gestion de projet a ressources limit ee au domaine de la construction bâtiment A mira béjaia,2002.
- [25] O. LOGIS. C. ROBERT. << Théorie des graphes >>. Vnibert. 2003.
- [26] Ph.La Comme.C. Paris . M.Sevaux. << Algorithmes de Graphes>> Eyrolles. 2003.
- [27] Robert 1918-1982 FAURE. Reseaux << Exercice et Problèmes résolus de recherche opérationnelles . Graphes : leurs usages, leurs algorithmes >> Masson.Paris.1983.

- [28] -S.SKIENA << Implementing Discrete Mathematics Combinatorics and Graph Theory with Mathematica>> Addison-Wesley. 1990.
- [29] Thèse de Wadji Tekaya, <<problème d'arbre couvrant de poids minimum probabiliste>>.
- [30] Thèse de Aurélie Favier << Décompositions fonctionnelles et structurelles dans les modèles graphiques probabilistes>> Université Toulouse

Conclusion générale

Dans la vie quotidienne, la plupart des problèmes "réels" peuvent être modélisés de manière plus ou moins fidèle à la réalité.

Il faut toujours garder en mémoire le nom probabilités d'inclure tous les facteurs environnants (ce qui base en soit le modèle créé). En effet, dans le modèle des graphes probabilistes, plusieurs chercheurs sont intéressés par le problème d'optimisation leurs travaux ont permis de donner plusieurs approches et algorithmes de résolution qui sont intéressants dans l'optimisation des situations concrètes qu'on peut modéliser sous forme des graphes probabilistes.

Dans le même contexte nous avons étudié certains techniques qui nous conduisent à modéliser l'évolution d'un système pouvant changer aléatoirement d'état et l'approche d'optimisation dans les graphes probabilistes qui nous permet à accéder à des solutions probables et optimales.