

Leçon 10

INTRODUCTION AUX PROBLEMES COMBINATOIRES "DIFFICILES" : LE PROBLEME DU VOYAGEUR DE COMMERCE ET LE PROBLEME DE COLORATION D'UN GRAPHE

Dans cette leçon, nous présentons deux problèmes très célèbres, celui du voyageur de commerce et celui de la coloration d'une carte de géographie.

Nous commençons par le problème du voyageur de commerce. Après en avoir donné la définition et des exemples d'application, nous aborderons le problème du temps de calcul de certains algorithmes, et proposerons des méthodes de résolution approchées.

Nous introduirons ensuite le problème de coloration des sommets d'un graphe. Après la définition et des exemples d'application, nous décrirons des méthodes de résolution approchées.

I Le problème du voyageur de commerce : exemple d'introduction

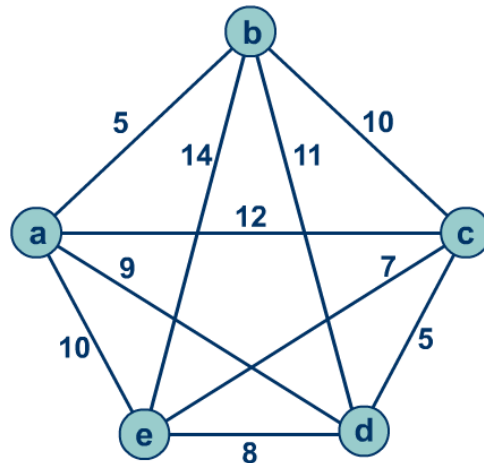
En 1962, Procter et Gamble lancèrent un concours pour résoudre le problème suivant :
Trouver un parcours passant par les 33 villes figurant sur la carte ci-dessous et le plus court possible.



C'était une des premières versions de ce qui deviendra le célèbre problème dit du "voyageur de commerce", problème étudié par Dantzig dès 1954. L'origine de ce nom est assez obscure et probablement peu de voyageurs de commerce l'utilisent !

Cf : <http://www.math.Princeton.edu/tsp/car54.html>

Le graphe ci-dessous correspond à un problème avec 5 villes en supposant que toutes les liaisons sont possibles. Les distances sont mises arbitrairement.



Il s'agit de partir d'un sommet et d'y revenir en parcourant une fois et une seule chaque sommet avec l'itinéraire le plus court. Cet itinéraire est indépendant de la ville de départ.
Par exemple, la longueur de l'itinéraire a b c d e a est égale à 38 alors que celle de a b e c d a vaut 40.

II Le problème du voyageur de commerce

Définition d'un cycle ou d'un circuit hamiltonien

Un **cycle hamiltonien** est un cycle qui passe une fois et une seule par chaque sommet du graphe. On dit aussi **tournée** ou **tour**.

Si le graphe est orienté, on définit de la même manière un **circuit hamiltonien**.

Dans un graphe de n sommets, le nombre de tournées est égal à $(n-1)!$, qui correspond aux nombres de permutations de $n-1$ éléments. ($(n-1)!$ et non $n!$ car on peut partir d'un sommet quelconque).

A chaque arête (ou arc si le graphe est orienté) on affecte un nombre c_{ij} qui peut, par exemple, représenter un coût, un temps ou... une longueur. On l'appelle longueur de l'arête (ou de l'arc).

Définition du problème du voyageur de commerce

Cas non orienté ou symétrique $c_{ij} = c_{ji}$

Il s'agit de déterminer parmi tous les **cycles** hamiltoniens, celui (ou ceux) de longueur minimale.

Cas orienté (ou non symétrique) $c_{ij} \neq c_{ji}$

Il s'agit de déterminer parmi tous les **circuits** hamiltoniens, celui (ou ceux) de longueur minimale.

Le problème du voyageur de commerce est le cas le plus simple du problème d'organisation de tournées de livraisons.

Exemple de problème modélisable par un problème de voyageur de commerce

Le processus de peinture des carrosseries automobiles est complexe et, dans certains cas, il comprend le passage des carrosseries dans une cuve de peinture. Après une série d'une peinture donnée, on change de peinture.

Le passage d'une peinture à une autre génère un coût fixe, dû au temps d'arrêt pour le nettoyage de la cuve ainsi qu'à la perte des matières premières résiduelles.

Le coût qui en découle dépend des peintures qui se succèdent.

Le problème est de déterminer l'ordre de passage des différentes peintures de manière à minimiser tous les coûts de changement.

Ce problème est modélisé par un problème de voyageur de commerce. On définit un graphe dont les sommets sont associés à différents coloris. Chaque arc (le problème est non symétrique car passer de i à j n'a pas le même coût que passer de j à i), représente le passage d'une peinture à la suivante. La longueur correspond au coût de changement de coloris.

Trouver le bon ordre de passage revient à déterminer une tournée optimale.

III Introduction à la complexité de certains problèmes

Il n'est pas rare de voir évoquer le problème du voyageur de commerce dans la presse traditionnelle ! Jusqu'à présent les algorithmes que nous avons rencontrés, par exemple dans la recherche de plus courts chemins, dans le problème central de l'ordonnancement ou du flot maximum sont de bons algorithmes en ce sens que le temps de calcul se comporte comme une fonction polynomiale de la taille du problème.

La taille du problème est, par exemple, dans le cas d'un problème de plus court chemin mesurée par le nombre de sommets et d'arcs du graphe et dans celui du voyageur de commerce par le nombre de villes.

Pour mesurer l'efficacité d'un algorithme, on évalue l'ordre de grandeur du nombre d'opérations élémentaires (addition, multiplication, affectation...) que devra faire l'ordinateur et ceci quelles que soient les données du problème lorsque sa taille tend vers l'infini.

Evaluation du temps de calcul en fonction de la nature de l'algorithme et de la taille du problème

Considérons 4 types d'algorithmes avec les comportements suivants :

- en "n" : le temps de calcul croît comme la taille du problème (c'est par exemple le cas de l'algorithme de Bellman)
- en n^2 (c'est le cas de l'algorithme de Moore Dijkstra)
- en n^3
- en e^n

Dans chacun des cas, on fait l'hypothèse qu'on résout en un millièmètre (10^{-3}) de secondes un problème de taille 10.

Que devient le temps de calcul lorsque la taille du problème est multipliée par 10, par 100 ou par 1000, donc passe à 100, 1000 ou 10 000 ?

Dans le tableau suivant les temps sont en secondes.

Taille ->	10	100	1000	10 000
Algo en n	10^{-3}	10^{-2}	10^{-1}	1
Algo en n^2	10^{-3}	10^{-1}	10	1 000
Algo en n^3	10^{-3}	1	1 000	1 000 000
Algo en e^n	10^{-3}	10^{39}		

Pour le premier algorithme en "n", lorsque la taille est multipliée par 10 le temps l'est également. On arrive donc, pour une taille de 10 000 (taille multipliée par 1000) à 1 seconde ($10^{-3} * 10^3$).

Pour l'algorithme en n^2 , quand la taille du problème est multipliée par 10, le temps de calcul est multiplié par 10^2 .

Pour l'algorithme en n^3 , quand la taille du problème est multipliée par 10, le temps est multiplié par 10^3 . Donc si la taille est multipliée par 100, le temps sera multiplié par $100^3 = 10^6 = 1$ million ! et 1 million de secondes égal 11,5 jours !

Pour l'algorithme en e^n , le temps est multiplié par $e^{100}/e^{10} = e^{90}$ dont l'ordre de grandeur est 10^{39} , un "1" suivi de 39 zéros !

Le problème de voyageur de commerce appartient à une catégorie de problème pour lequel on ne possède actuellement pas d'algorithme dont le temps de calcul croît de manière raisonnable, c'est-à-dire polynomiale, avec la taille du problème.

Lorsque la taille du problème devient élevée (selon les données cela peut être quelques centaines de villes) on se contente de petits problèmes ou l'on abandonne l'idée d'obtenir une solution optimale.

L'enjeu économique est de taille, au point qu'à l'aube du troisième millénaire le Clay Institute a mis en jeu 1 million de dollars pour qui trouverait un bon algorithme pour résoudre ce type de problème, celui du voyageur de commerce et tous ceux qui tombent dans cette même catégorie et il y en a

beaucoup ! Voir http://www.claymath.org/Millennium_Prize_Problems/

Face à ce challenge, on va se contenter de méthodes approchées encore appelées "heuristiques" qui vont permettre de trouver rapidement une solution que l'on espère pas trop éloignée de la solution optimale.

IV Résolution du problème du voyageur de commerce par des méthodes approchées

De très nombreuses heuristiques existent pour le problème du voyageur de commerce.

Heuristique du plus proche voisin

Principe

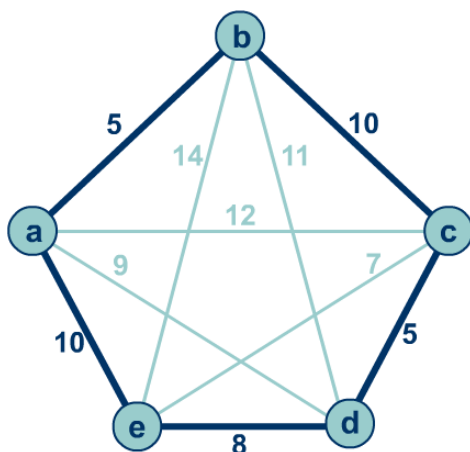
On part d'une ville quelconque et l'on se dirige vers la ville la plus proche sans repasser par une ville déjà visitée.

Partir d'un sommet

TANT QUE la tournée n'est pas complète

 Aller à la ville la plus proche non encore visitée

FINTANTQUE



Si on part de a, la ville la plus proche est b, puis c, puis d, puis e, puis a.

On obtient la tournée a b c d e a de longueur :
 $5 + 10 + 5 + 8 + 10 = 38$

Cette heuristique très simple peut donner des résultats arbitrairement mauvais.

Si la longueur de l'arête (a, e) est égale à M, nombre arbitrairement grand, l'heuristique du plus proche voisin donnera toujours la tournée a b c d e a dont la longueur (28 + M) peut être aussi éloignée que l'on veut de la solution optimale qui est en fait, dès que la longueur de (a, e) dépasse 11, le tour a b c e d a de longueur 39.

Une deuxième heuristique

Principe

On commence par trier les arêtes par longueur croissante.

Puis on parcourt la liste triée des arêtes et on sélectionne les arêtes de manière à ne pas créer de court circuit, ni de sommet de degré 3, c'est-à-dire une fourche, ceci jusqu'à ce qu'on puisse revenir au sommet de départ.

Trier les arêtes par ordre de longueur croissante

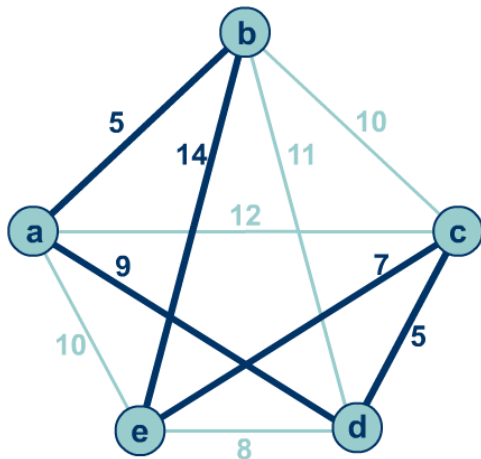
TANTQUE la tournée n'est pas complète

 Prendre la première arête non examinée de la liste

 Si elle ne crée ni cycle, ni fourche avec les arêtes précédemment sélectionnées, la retenir.

FINTANTQUE

Exemple



On commence par (a, b) de longueur 5 puis (c, d) de longueur 5 également, puis (c, e) de longueur 7.
On ne peut pas prendre (e, d) de longueur 8 qui créerait un sous-cycle decd, donc on prend (a, d) de longueur 9 et on termine par (b, e) de longueur 14.
On trouve la tournée a b e c d a de longueur 40.

Contrairement à l'heuristique du plus proche voisin, si la longueur de l'arête (a, e) passe à M, cette nouvelle arête n'a aucune chance d'être retenue puisque les arêtes sont examinées dans l'ordre des longueurs croissantes, cette heuristique conduit encore à la tournée a b e c d a.

Evaluation de la performance des heuristiques

Nous venons de voir que l'heuristique du plus proche voisin pouvait donner des résultats aussi éloignés que l'on veut de la solution optimale.

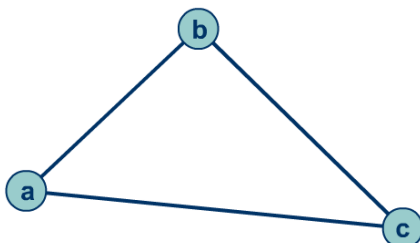
Pour évaluer la performance d'une heuristique, on peut comparer le résultat qu'elle fournit à une minoration de la tournée optimale (en pratique on ne connaît évidemment pas la tournée optimale).

Par exemple, pour le graphe précédent, on peut affirmer que toute tournée a une longueur au moins égale à la somme des longueurs des 5 arêtes les plus courtes, soit ici 34.

L'heuristique 2 donne une tournée de longueur 40. On est donc au pire à 6 sur 34 soit 17 % de la tournée optimale.

En fait, la tournée optimale a pour longueur 38. Il y a 2 solutions a b c d e a et a e b d c e a (dont le tour donné par l'heuristique du plus proche voisin !).

Dans certains cas, on est en mesure d'évaluer a priori les performances de certaines heuristiques, en particulier lorsque le graphe est "euclidien" : dans ce cas l'inégalité triangulaire est vérifiée.



$$\begin{aligned}c &< a + b \\b &< a + c \\a &< b + c.\end{aligned}$$

On peut alors démontrer que la longueur de la tournée fournie par l'heuristique précédente est, dans tous les cas de figure, c'est à dire quelles que soient les données du problème, inférieure à 2 fois celle

de la tournée optimale.

La meilleure heuristique connue actuellement pour le problème de voyageur de commerce euclidien est telle que, quel que soit le problème, la longueur de la tournée obtenue est, de manière certaine, inférieure ou égale à 1,5 fois la longueur de la tournée optimale. On dit qu'elle a un rapport de performance garantie de 1,5.

Le problème du voyageur de commerce donne lieu à beaucoup de travaux.

Par exemple, la solution optimale d'un problème sur 15112 villes en Allemagne a été obtenue en 2001. Les méthodes utilisées sont des méthodes mathématiques sophistiquées qui ont été mises en oeuvre sur tout un ensemble d'ordinateurs travaillant en parallèle.

cf : <http://www.math.princeton.edu/tsp/history.html>

V Coloration des sommets d'un graphe

Le problème de coloration d'une carte de géographie est également très célèbre. Il s'agit de trouver le nombre minimum de couleurs nécessaires pour colorier une carte géographique de telle manière que deux pays ayant une frontière commune n'aient pas la même couleur. La réponse a été apportée en 1976 : il suffit de 4 couleurs.

Afin de modéliser ce problème, nous définissons un nouveau type de problème d'optimisation sur un graphe.

Coloration des sommets d'un graphe

Il s'agit d'affecter à chaque sommet une couleur telle que 2 sommets adjacents n'aient pas la même couleur.

Trouver une coloration ne pose aucun problème ; il suffit de mettre une couleur différente par sommet.

Le vrai problème est de le faire avec un nombre minimum de couleurs.

Nombre chromatique d'un graphe

On appelle **nombre chromatique** d'un graphe le nombre minimum de couleurs nécessaires pour une coloration des sommets.

Le problème de coloration d'une carte géographique se ramène à la coloration des sommets d'un graphe : à la carte nous associons un graphe dont les sommets correspondent aux pays. Lorsque deux pays ont une frontière commune, on relie les sommets associés par une arête.

Dans ce cas, on sait que 4 couleurs suffisent. La raison est que le graphe est particulier : il est planaire, c'est à dire qu'il en existe une représentation telle que 2 arêtes ne se coupent pas

Exemple d'application

Les organisateurs d'un congrès sont confrontés à la réalisation du planning des conférences. Certaines peuvent avoir lieu simultanément alors que d'autres doivent impérativement être placées à des moments différents. Chaque conférence occupant une demi-journée, les organisateurs voudraient que les conférences puissent être placées dans un minimum de temps.

On suppose pour cet exemple qu'il n'y a que 7 conférences. Les incompatibilités entre elles sont traduites dans ce tableau.

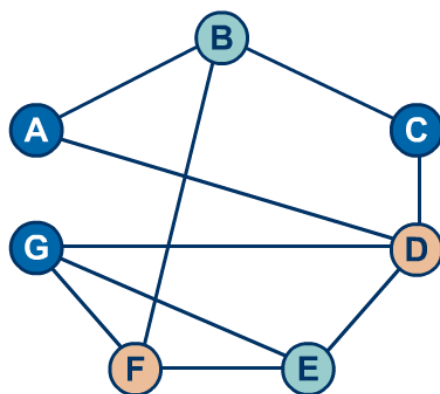
Combien faut-il de demi-journées au minimum ?

	A	B	C	D	E	F	G
A		***		***			
B			***			***	
C				***			
D					***		***
E						***	***
F							***
G							

"***" indique que les conférences ne doivent pas être organisées simultanément.

On modélise ce problème par un problème de coloration des sommets d'un graphe. Les sommets sont associés aux conférences. Deux conférences incompatibles sont reliées entre elles par une arête. Le graphe obtenu est un "graphe d'incompatibilité". Dans une coloration des sommets, on ne peut attribuer la même couleur à 2 conférences incompatibles.

On peut donc mettre sur la même demi-journée toutes les conférences associées à des sommets de même couleur. Le nombre minimum de couleurs correspond au nombre minimum de demi-journées. On a par exemple la solution suivante (il y en a d'autres) : A, C et G couleur 1, B et E couleur 2, D et F couleur 3.



Il suffit de 3 demi-journées. On ne peut pas faire mieux, par exemple parce que E, F et G sont incompatibles 2 à 2.

Résolution de problèmes de coloration par des méthodes approchées

Dans le cas général, on ne connaît pas de bons algorithmes pour résoudre le problème de coloration avec un nombre minimum de couleurs. C'est encore un problème à un million de dollars ! On fait donc appel à des heuristiques pour obtenir une bonne solution.

L'heuristique la plus simple consiste à prendre les sommets dans un ordre donné (qui peut être quelconque) et à leur affecter la première couleur possible.

La solution dépend bien sûr de l'ordre dans lequel les sommets seront examinés.

Numéroter les sommets de 1 à n

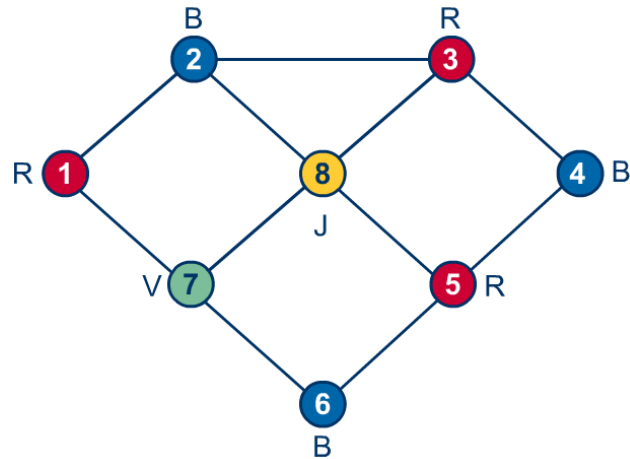
Numéroter les couleurs

POUR i de 1 à n

Colorier le sommet i avec la première couleur possible.

FINPOUR

Exemple



Ordre des couleurs : 1 : Rouge, 2 : bleu, 3 : vert, 4 : jaune

On commence par le sommet 1 à qui l'on attribue la première couleur : rouge.

La première couleur possible pour le sommet 2 est la couleur 2, le bleu.

Pour le sommet 3, la couleur possible de plus petit numéro est le rouge (couleur 1).

On peut ensuite mettre le sommet 4 en bleu, 5 en rouge et 6 en bleu.

Pour le sommet 7 on doit utiliser le vert et pour le sommet 8 la quatrième couleur, le jaune.

On vient d'obtenir une coloration avec 4 couleurs. Mais ce n'est pas la solution optimale !

L'amélioration de l'heuristique précédente passe par un choix plus judicieux de l'ordre d'examen des sommets.

On modifie l'heuristique précédente en numérotant les sommets dans l'ordre de leur degré décroissant. Pour le reste, le principe est inchangé.

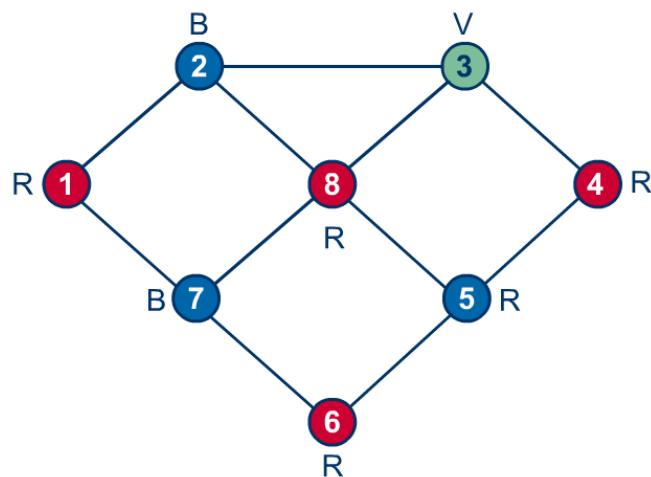
Numéroter les sommets dans l'ordre des degrés décroissants

Numéroter les couleurs

POUR i de 1 à n

Colorier le sommet i avec la première couleur possible.

FINPOUR



On commence donc la coloration par le sommet qui porte maintenant le numéro 1 car il est de degré 4, on passe ensuite au sommet 2 (de degré 3) puis au sommet 3 etc....

On constate que 3 couleurs suffisent maintenant et l'on ne peut trouver mieux en raison des sommets 1, 2 et 3 qui doivent être 2 à 2 de couleurs différentes.

Ici l'heuristique a fourni la solution optimale mais comme ce n'est qu'une méthode approchée ce n'est hélas pas toujours le cas !

Heuristique DSATUR

L'heuristique précédente peut encore être améliorée (statistiquement) de la manière suivante :
Le choix des sommets est fait sur un critère dynamique.

On définit d'abord le degré de saturation d'un sommet.

Le **degré de saturation** $DSAT(i)$ est défini de la manière suivante :

SI aucun sommet adjacent à i n'est colorié alors

$$DSAT(i) = \text{degré}(i)$$

SINON

$$DSAT(i) = \text{nombre de couleurs différentes utilisées par les sommets adjacents à } i$$

On a alors l'heuristique appelée DSATUR qui est une des plus connues pour le problème de coloration.

POUR i de 1 à n poser $DSAT(i) = \text{degré}(i)$

Numéroter les couleurs

TANTQUE il existe un sommet non colorié

 Choisir un sommet avec $DSAT$ maximum, en cas d'égalité celui de plus grand degré, en cas d'égalité celui de plus petit numéro

 Colorier le sommet i avec la première couleur possible

 Mettre à jour $DSAT$ pour les sommets adjacents à ce sommet

FINTANTQUE

L'heuristique DSATUR est en particulier à la base de la résolution du problème d'allocation de fréquence en téléphonie mobile. Ce problème se ramène à un problème de coloration de graphe, les sommets correspondent aux zones, 2 zones voisines ne devant pas avoir la même fréquence pour éviter les interférences, les sommets associés seront reliés par une arête.