Quelques éléments de théorie des graphes

Daniel Perrin & Damien Thomine

6 novembre 2017

Théorie des graphes au CAPES

Enseignement en terminale spécialité mathématiques :

- Filière S : graphes stochastiques.

Théorie des graphes au CAPES

Enseignement en terminale spécialité mathématiques :

- Filière S : graphes stochastiques.
- Filière ES : graphes stochastiques, graphes eulériens, matrice d'adjacence, recherche de plus court chemin, nombre chromatique...

Théorie des graphes au CAPES

Enseignement en terminale spécialité mathématiques :

- Filière S : graphes stochastiques.
- Filière ES : graphes stochastiques, graphes eulériens, matrice d'adjacence, recherche de plus court chemin, nombre chromatique...

Théorie des graphes au CAPES

Enseignement en terminale spécialité mathématiques :

- Filière S : graphes stochastiques.
- Filière ES : graphes stochastiques, graphes eulériens, matrice d'adjacence, recherche de plus court chemin, nombre chromatique...

Applications classiques : algorithme pagerank, diffusion dans une population (épidémiologie), chaînes de Markov avec peu détat, modèle d'Ehrenfest, codes et automates...

Théorie des graphes au CAPES

Enseignement en terminale spécialité mathématiques :

- Filière S : graphes stochastiques.
- Filière ES : graphes stochastiques, graphes eulériens, matrice d'adjacence, recherche de plus court chemin, nombre chromatique...

Applications classiques : algorithme pagerank, diffusion dans une population (épidémiologie), chaînes de Markov avec peu détat, modèle d'Ehrenfest, codes et automates...

Concours du CAPES : option informatique.

Théorie des graphes au lycée

Les graphes sont des objets utilisés dans la résolution de nombreux problèmes concrets. Il demandent peu de pré-requis.

Théorie des graphes au lycée

Les graphes sont des objets utilisés dans la résolution de nombreux problèmes concrets. Il demandent peu de pré-requis.

En cours : partir de problèmes concrets. Introduire les graphes progressivement, pour aider à la résolution et généraliser.

Théorie des graphes au lycée

Les graphes sont des objets utilisés dans la résolution de nombreux problèmes concrets. Il demandent peu de pré-requis.

En cours : partir de problèmes concrets. Introduire les graphes progressivement, pour aider à la résolution et généraliser.

En profiter pour expliciter la modélisation mathématique.

Bibliographie

[CG] **Olivier Cogis, Claudine Robert,** *Théorie des graphes, problèmes, théorèmes, algorithmes*, Vuibert 2003.

[A] **Arnoux Pierre et al,** *Graphes pour la terminale ES*, IREM Marseille 2002 (en ligne, 99 pages)

[S] **Sigward Éric,** *Illustration des nouveaux programmes 2002-2003 de mathématiques en terminale ES* (en ligne, 46 pages)

[T] **Tangente numéro 75-76, etc.**

Quelques définitions

Graphe (ou multigraphe)

On appelle **graphe** (ou multigraphe) la donnée :

- d'une ensemble fini V (les **sommets**) ;
- d'une ensemble fini E (les **arêtes**) ;
- d'une application $\varphi : E \to \mathcal{P}_1(V) \cup \mathcal{P}_2(V)$.

Graphe (ou multigraphe)

On appelle **graphe** (ou multigraphe) la donnée :

- d'une ensemble fini V (les **sommets**) ;
- d'une ensemble fini E (les **arêtes**) ;
- d'une application $\varphi : E \to \mathcal{P}_1(V) \cup \mathcal{P}_2(V)$.

Les **extrémités** d'une arête e sont les éléments de $\varphi(e)$. Les **boucles** sont les arêtes ayant une seule extrémité.

Graphe (ou multigraphe)

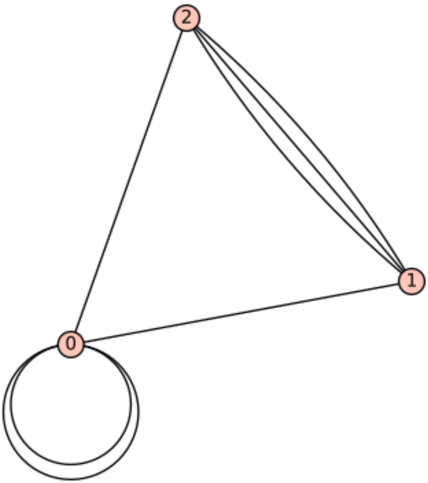
On appelle **graphe** (ou multigraphe) la donnée :

- d'une ensemble fini V (les **sommets**) ;
- d'une ensemble fini E (les **arêtes**) ;
- d'une application $\varphi : E \to \mathcal{P}_1(V) \cup \mathcal{P}_2(V)$.

Les **extrémités** d'une arête e sont les éléments de $\varphi(e)$. Les **boucles** sont les arêtes ayant une seule extrémité.

On dessine une arête entre x et y si $\{x, y\}$ pour chaque arête $e \in E$ telle que $\varphi(e) = \{x, y\}$.

Graphe : exemple



Graphe simple

On appelle **graphe simple** un graphe sans boucle, et tel que φ est injective (sans arête multiple).

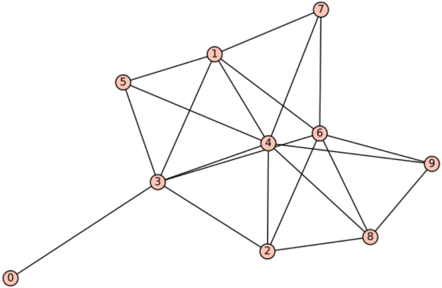
Graphe simple

On appelle **graphe simple** un graphe sans boucle, et tel que φ est injective (sans arête multiple).

De façon équivalente, un graphe simple est la donnée :

- d'une ensemble fini V ;
- d'une partie $E \subset \mathcal{P}_2(V)$.

Graphe simple : exemple



Graphe orienté

On appelle **graphe orienté** la donnée :

- d'une ensemble fini V ;
- d'une ensemble fini E ;
- de deux applications φ_-, $\varphi_+ : E \to V$.

Graphe orienté

On appelle **graphe orienté** la donnée :

- d'une ensemble fini V ;
- d'une ensemble fini E ;
- de deux applications φ_-, $\varphi_+ : E \to V$.

Chaque arête e a un sommet de départ $\varphi_-(e)$ et un sommet d'arrivée $\varphi_+(e)$.

Graphe orienté

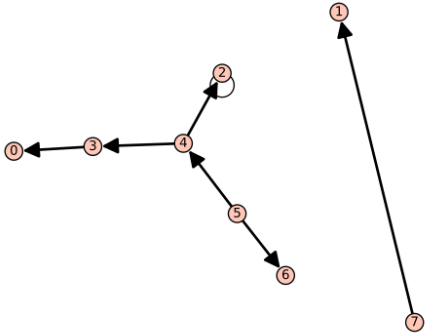
On appelle **graphe orienté** la donnée :

- d'une ensemble fini V ;
- d'une ensemble fini E ;
- de deux applications φ_-, $\varphi_+ : E \to V$.

Chaque arête e a un sommet de départ $\varphi_-(e)$ et un sommet d'arrivée $\varphi_+(e)$.

On dessine une flèche de x vers y pour chaque arête $e \in E$ telle que $\varphi_-(e) = x$ et $\varphi_+(e) = y$.

Graphe orienté : exemple



Graphe à poids

On appelle **graphe à poids** la donnée d'un graphe (V, E, φ), et d'une application $p : E \to \mathbb{R}_+^*$. Par exemple, on peut ainsi modéliser :

- une carte routière : les poids sont les longueurs des routes ;

Graphe à poids

On appelle **graphe à poids** la donnée d'un graphe (V, E, φ), et d'une application $p : E \to \mathbb{R}_+^*$. Par exemple, on peut ainsi modéliser :

- une carte routière : les poids sont les longueurs des routes ;
- un circuit électrique : les poids sont des résistances ;

Graphe à poids

On appelle **graphe à poids** la donnée d'un graphe (V, E, φ), et d'une application $p : E \to \mathbb{R}_+^*$. Par exemple, on peut ainsi modéliser :

- une carte routière : les poids sont les longueurs des routes ;
- un circuit électrique : les poids sont des résistances ;
- un réseau : les poids sont des capacités.

Graphe à poids

On appelle **graphe à poids** la donnée d'un graphe (V, E, φ), et d'une application $p : E \to \mathbb{R}_+^*$. Par exemple, on peut ainsi modéliser :

- une carte routière : les poids sont les longueurs des routes ;
- un circuit électrique : les poids sont des résistances ;
- un réseau : les poids sont des capacités.

Graphe à poids

On appelle **graphe à poids** la donnée d'un graphe (V, E, φ), et d'une application $p : E \to \mathbb{R}_+^*$. Par exemple, on peut ainsi modéliser :

- une carte routière : les poids sont les longueurs des routes ;
- un circuit électrique : les poids sont des résistances ;
- un réseau : les poids sont des capacités.

Par le même principe, on peut avoir des poids complexes, des étiquettes diverses attachées aux sommets ou aux arêtes...

Chemin

Dans un graphe, un **chemin** de longueur n est la donnée :

- d'une suite de $n+1$ sommets $(v_0, \ldots, v_n)$;
- d'une suite de n arêtes $(e_0, \ldots, e_{n-1})$;
- telles que e_k soit d'extrémités v_k et v_{k+1} pour tout $0 \leq k < n$.

Chemin

Dans un graphe, un **chemin** de longueur n est la donnée :

- d'une suite de $n+1$ sommets $(v_0, \ldots, v_n)$;
- d'une suite de n arêtes $(e_0, \ldots, e_{n-1})$;
- telles que e_k soit d'extrémités v_k et v_{k+1} pour tout $0 \leq k < n$.

Pour un graphe orienté, il faut de plus que le chemin respecte l'orientation : $\varphi_-(e_k) = v_k$ et $\varphi_+(e_k) = v_{k+1}$.

Chemin

Dans un graphe, un **chemin** de longueur n est la donnée :

- d'une suite de $n+1$ sommets $(v_0, \ldots, v_n)$;
- d'une suite de n arêtes $(e_0, \ldots, e_{n-1})$;
- telles que e_k soit d'extrémités v_k et v_{k+1} pour tout $0 \leq k < n$.

Pour un graphe orienté, il faut de plus que le chemin respecte l'orientation : $\varphi_-(e_k) = v_k$ et $\varphi_+(e_k) = v_{k+1}$.

Pour un graphe à poids, la longueur d'un chemin est parfois définie comme :

$$\sum_{k=0}^{n-1} p(e_k).$$

Cycle

Dans un graphe (orienté ou non), un **cycle** est un chemin dont les points de départ et d'arrivée sont identiques. Un cycle est :

Cycle

Dans un graphe (orienté ou non), un **cycle** est un chemin dont les points de départ et d'arrivée sont identiques. Un cycle est :

- **simple** si aucune arête n'apparaît 2 fois ou plus ;
- **élémentaire** si aucun sommet (sauf éventuellement les extrémités) n'apparaît 2 fois ou plus.

Connexité

Un graphe est **connexe** si entre deux sommets il existe toujours un chemin.

Connexité

Un graphe est **connexe** si entre deux sommets il existe toujours un chemin.

Pour un graphe orienté, il existe plusieurs notions de connexité. Un tel graphe est :

- **faiblement connexe** si le graphe obtenu en oubliant l'orientation est connexe ;
- **fortement connexe** si, d'un sommet à un autre, il existe toujours un chemin orienté.

Connexité

Un graphe est **connexe** si entre deux sommets il existe toujours un chemin.

Pour un graphe orienté, il existe plusieurs notions de connexité. Un tel graphe est :

- **faiblement connexe** si le graphe obtenu en oubliant l'orientation est connexe ;
- **fortement connexe** si, d'un sommet à un autre, il existe toujours un chemin orienté.

Un graphe non connexe peut être décomposé en composantes connexes (ou composantes faiblement connexes pour les graphes orientés).

Distance

La distance entre deux sommets a et b est la longueur minimale d'un chemin de a à b.

Distance

La distance entre deux sommets a et b est la longueur minimale d'un chemin de a à b.
Exercice : vérifier que, dans un graphe ou un graphe à poids connexes, on définit bien une distance sur l'ensemble des sommets du graphe.

Distance

La distance entre deux sommets a et b est la longueur minimale d'un chemin de a à b.
Exercice : vérifier que, dans un graphe ou un graphe à poids connexes, on définit bien une distance sur l'ensemble des sommets du graphe.
Pour un graphe orienté, la distance n'est en général pas symétrique.

Morphisme de graphes

Un **morphisme** d'un graphe (V, E, φ) vers un graphe (V', E', φ') est une application $\psi = (\psi_V, \psi_E) : V \times E \to V' \times E'$ qui respecte les extrémités :

$$\varphi' \circ \psi_E = \psi_{V*} \circ \varphi.$$

Un **isomorphisme** de graphes est un morphisme bijectif.

Morphisme de graphes

Un **morphisme** d'un graphe (V, E, φ) vers un graphe (V', E', φ') est une application $\psi = (\psi_V, \psi_E) : V \times E \to V' \times E'$ qui respecte les extrémités :

$$\varphi' \circ \psi_E = \psi_{V*} \circ \varphi.$$

Un **isomorphisme** de graphes est un morphisme bijectif.

Problème : comment déterminer si deux graphes sont isomorphes ? Applications à la chimie.

Degré

Le **degré** d'un sommet v d'un graphe, noté $d(v)$, est le nombre d'arêtes dont ce sommet est une extrémité. *Les boucles comptent pour* 2.

Degré

Le **degré** d'un sommet v d'un graphe, noté $d(v)$, est le nombre d'arêtes dont ce sommet est une extrémité. *Les boucles comptent pour* 2*.*

La **suite des degrés** d'un graphe est la liste de ses degrés, arrangés par ordre croissant. Un graphe est d-**régulier** si tous ses sommets sont de degré d.

Degré

Le **degré** d'un sommet v d'un graphe, noté $d(v)$, est le nombre d'arêtes dont ce sommet est une extrémité. *Les boucles comptent pour* 2.

La **suite des degrés** d'un graphe est la liste de ses degrés, arrangés par ordre croissant. Un graphe est d-**régulier** si tous ses sommets sont de degré d.

La suite des degrés aide à déterminer si deux graphes sont isomorphes (ou non). Le problème de l'isomorphisme reste algorithmiquement difficile, en particulier pour les graphes réguliers.

Degré

Le **degré** d'un sommet v d'un graphe, noté $d(v)$, est le nombre d'arêtes dont ce sommet est une extrémité. *Les boucles comptent pour* 2.

La **suite des degrés** d'un graphe est la liste de ses degrés, arrangés par ordre croissant. Un graphe est d-**régulier** si tous ses sommets sont de degré d.

La suite des degrés aide à déterminer si deux graphes sont isomorphes (ou non). Le problème de l'isomorphisme reste algorithmiquement difficile, en particulier pour les graphes réguliers.

Pour des graphes orientés, il y a un **degré sortant** $d_+(v)$ et un **degré entrant** $d_-(v)$:

$$d_+(v) = |\{e \in E : \varphi_-(e) = v\}|$$
$$d_-(v) = |\{e \in E : \varphi_+(e) = v\}|$$

Théorème des poignées de main

Théorème (Théorème des poignées de mains)

Soit (V, E, φ) *un graphe. Alors :*

$$\sum_{v \in V} d(v) = 2|E|.$$

Théorème des poignées de main

Théorème (Théorème des poignées de mains)

Soit (V, E, φ) *un graphe. Alors :*

$$\sum_{v \in V} d(v) = 2|E|.$$

Théorème des poignées de main

Théorème (Théorème des poignées de mains)

Soit (V, E, φ) *un graphe. Alors :*

$$\sum_{v \in V} d(v) = 2|E|.$$

Théorème (Version pour les graphes orientés)

Soit $(V, E, \varphi_-, \varphi_+)$ *un graphe orienté. Alors :*

$$\sum_{v \in V} d_-(v) = \sum_{v \in V} d_+(v) = |E|.$$

Graphe complet

Un **graphe complet** à n sommets, noté K_n, est un graphe simple tel que $|V| = n$ et $E = \mathcal{P}_2(V)$.

Graphe complet

Un **graphe complet** à n sommets, noté K_n, est un graphe simple tel que $|V| = n$ et $E = \mathcal{P}_2(V)$.

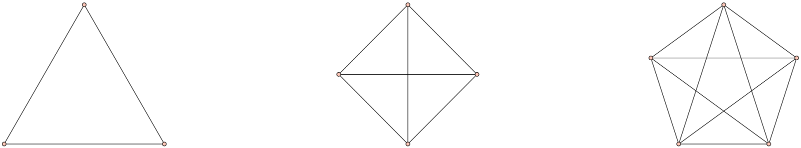


FIGURE : Les graphes complets K_3, K_4 et K_5

Graphe biparti complet

Un **graphe biparti complet** à $n = a + b$ sommets, noté $K_{a,b}$, est un graphe simple tel que $V = A \sqcup B$ avec $|A| = a$ et $|B| = b$, et $E = \{\{a, b\} : \ a \in A, \ b \in B\}$.

Graphe biparti complet

Un **graphe biparti complet** à $n = a + b$ sommets, noté $K_{a,b}$, est un graphe simple tel que $V = A \sqcup B$ avec $|A| = a$ et $|B| = b$, et $E = \{\{a, b\} : \ a \in A, \ b \in B\}$.

FIGURE : Les graphes complets bipartis $K_{2,3}$ et $K_{3,3}$

Graphe cyclique

Un **graphe cyclique** à $n \geq 3$ sommets, noté C_n, est un graphe simple tel que $V \simeq \mathbb{Z}_{/n\mathbb{Z}}$ et $E = \{\{k, k+1\} : \ k \in Z_{/n\mathbb{Z}}\}$.

Graphe cyclique

Un **graphe cyclique** à $n \geq 3$ sommets, noté C_n, est un graphe simple tel que $V \simeq \mathbb{Z}_{/n\mathbb{Z}}$ et $E = \{\{k, k+1\} : \ k \in Z_{/n\mathbb{Z}}\}$.

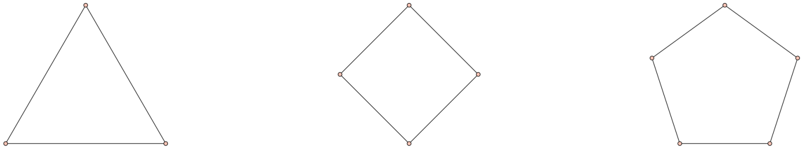


FIGURE : Les graphes cycliques C_3, C_4 et C_5

Arbre

Un **arbre** est un graphe simple connexe sans cycle simple non trivial.

Arbre

Un **arbre** est un graphe simple connexe sans cycle simple non trivial.

Un arbre à n sommets a $n-1$ arêtes. Ses sommets de degré 1 sont des **feuilles**.

Arbre

Un **arbre** est un graphe simple connexe sans cycle simple non trivial.

Un arbre à n sommets a $n-1$ arêtes. Ses sommets de degré 1 sont des **feuilles**.

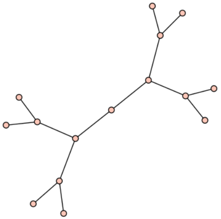


FIGURE : Un arbre

Et les autres...

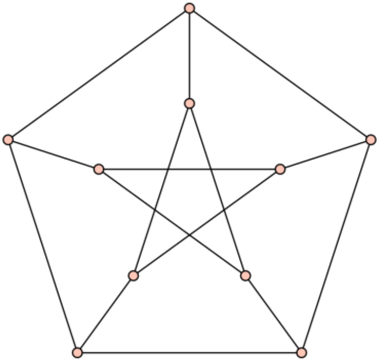
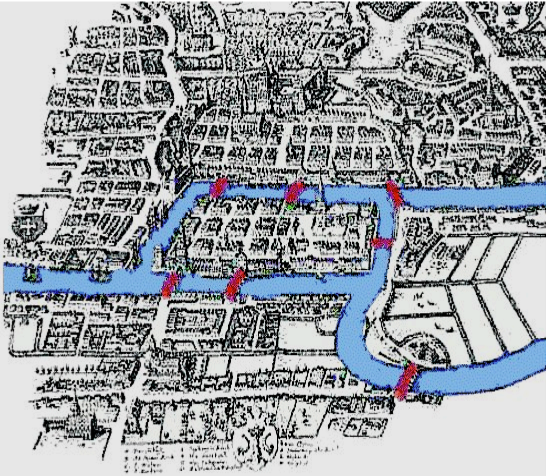


FIGURE : Le graphe de Petersen

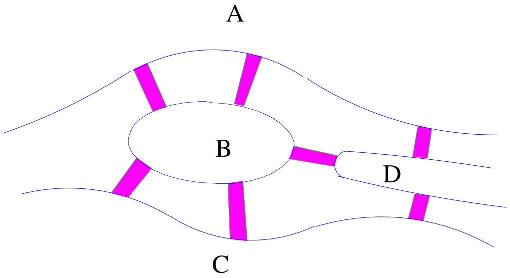
Graphes eulériens et hamiltoniens

Les ponts de Königsberg

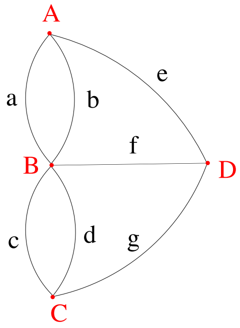


Les sept ponts de Königsberg

Les ponts de Königsberg : graphe

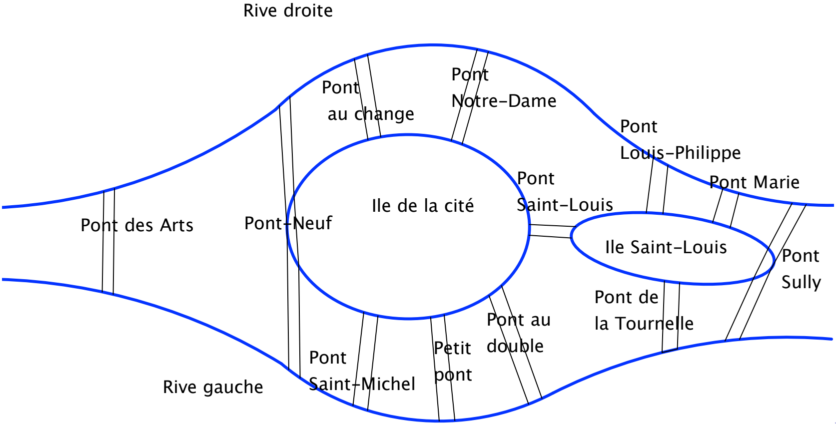


Les ponts de Königsberg : quel itinéraire faut-il emprunter pour passer une fois et une seule sur chaque pont ?



et leur graphe

... et ceux de Paris



Graphe eulérien

Soit $G = (V, E, \varphi)$ un graphe orienté ou non. Un chemin est dit **eulérien** si il emprunte une fois et une seule chaque arête.

Graphe eulérien

Soit $G = (V, E, \varphi)$ un graphe orienté ou non. Un chemin est dit **eulérien** si il emprunte une fois et une seule chaque arête.

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

Théorème d'Euler

Théorème (L. Euler, 1736 ; C. Hierzoler, 1873†)

Un graphe connexe G est eulérien si et seulement si tous les sommets de G sont de degré pair.

Théorème d'Euler

Théorème (L. Euler, 1736 ; C. Hierzoler, 1873†)

Un graphe connexe G est eulérien si et seulement si tous les sommets de G sont de degré pair.

Théorème d'Euler

Théorème (L. Euler, 1736 ; C. Hierzoler, 1873†)

Un graphe connexe G est eulérien si et seulement si tous les sommets de G sont de degré pair.

Théorème (Théorème d'Euler, version graphes orientés)

Un graphe orienté faiblement connexe G est eulérien si et seulement si $d_-(v) = d_+(v)$ pour tout sommet v de G.

Théorème d'Euler

Théorème (L. Euler, 1736 ; C. Hierzoler, 1873†)

Un graphe connexe G est eulérien si et seulement si tous les sommets de G sont de degré pair.

Théorème (Théorème d'Euler, version graphes orientés)

Un graphe orienté faiblement connexe G est eulérien si et seulement si $d_-(v) = d_+(v)$ pour tout sommet v de G.

Théorème d'Euler

Théorème (L. Euler, 1736 ; C. Hierzoler, 1873†)

Un graphe connexe G est eulérien si et seulement si tous les sommets de G sont de degré pair.

Théorème (Théorème d'Euler, version graphes orientés)

Un graphe orienté faiblement connexe G est eulérien si et seulement si $d_-(v) = d_+(v)$ pour tout sommet v de G.

Voir : CAPES, option informatique, sujet 0.

Démonstration du théorème d'Euler

La preuve de la condition nécessaire (si on a un cycle eulérien, tous les sommets sont de degré pair) est facile.

Démonstration du théorème d'Euler

La preuve de la condition nécessaire (si on a un cycle eulérien, tous les sommets sont de degré pair) est facile.

Pour la condition suffisante, la preuve est plus compliquée, et algorithmique :

- on décompose l'ensemble des arêtes en un nombre fini de cycles ;

Démonstration du théorème d'Euler

La preuve de la condition nécessaire (si on a un cycle eulérien, tous les sommets sont de degré pair) est facile.

Pour la condition suffisante, la preuve est plus compliquée, et algorithmique :

- on décompose l'ensemble des arêtes en un nombre fini de cycles ;
- on recolle ces cycles un par un.

Décomposition en cycles

- on part d'un sommet quelconque, disons A. Il y a au moins une arête issue de A. On la prend et on arrive en B.

Décomposition en cycles

- on part d'un sommet quelconque, disons A. Il y a au moins une arête issue de A. On la prend et on arrive en B.
- comme B est de degré pair, il y a une arête qui repart de B. On la prend et on arrive en C.

Décomposition en cycles

- on part d'un sommet quelconque, disons A. Il y a au moins une arête issue de A. On la prend et on arrive en B.
- comme B est de degré pair, il y a une arête qui repart de B. On la prend et on arrive en C.
- on continue ainsi, en ne prenant pas les arêtes déjà utilisées, jusqu'à être bloqué.

Décomposition en cycles

- on part d'un sommet quelconque, disons A. Il y a au moins une arête issue de A. On la prend et on arrive en B.
- comme B est de degré pair, il y a une arête qui repart de B. On la prend et on arrive en C.
- on continue ainsi, en ne prenant pas les arêtes déjà utilisées, jusqu'à être bloqué.
- si l'on est bloqué c'est nécessairement en A.

Décomposition en cycles

- on part d'un sommet quelconque, disons A. Il y a au moins une arête issue de A. On la prend et on arrive en B.
- comme B est de degré pair, il y a une arête qui repart de B. On la prend et on arrive en C.
- on continue ainsi, en ne prenant pas les arêtes déjà utilisées, jusqu'à être bloqué.
- si l'on est bloqué c'est nécessairement en A.
- on vient de parcourir un cycle. On efface les arêtes utilisées, et s'il reste des arêtes inutilisées on recommence à la première étape.

Recollement des cycles

- s'il y a au moins deux cycles : par connexité, on peut trouver deux cycles qui partagent un sommet.

Recollement des cycles

- s'il y a au moins deux cycles : par connexité, on peut trouver deux cycles qui partagent un sommet.
- on ouvre les deux cycles en ce sommet, et on les recolle.

Recollement des cycles

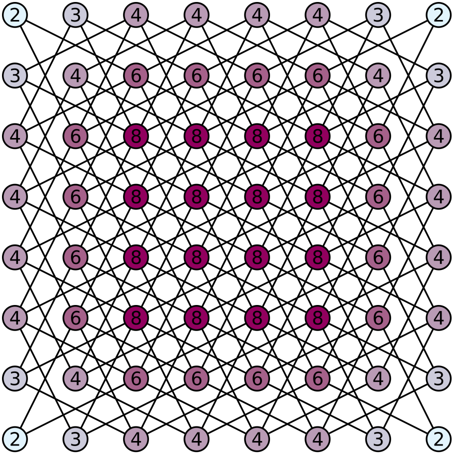
- s'il y a au moins deux cycles : par connexité, on peut trouver deux cycles qui partagent un sommet.
- on ouvre les deux cycles en ce sommet, et on les recolle.
- s'il ne reste plus qu'un cycle : ce cycle est eulérien, et on a terminé. Sinon, on recommence.

Chemins eulériens

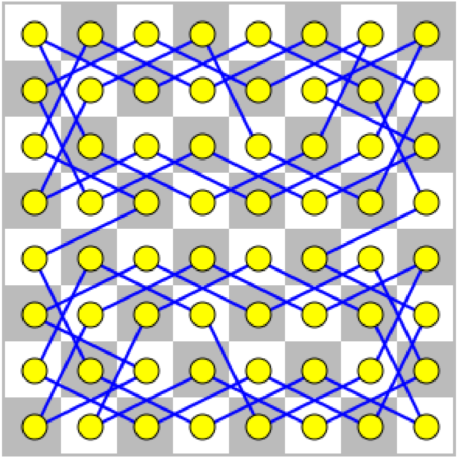
Théorème

Soit G un graphe connexe. Il existe un chemin eulérien dans G si et seulement si G a au plus deux sommets de degré impair.
Si G est connexe et a deux sommets de degré impair, alors tout chemin eulérien part d'un sommet de degré impair et arrive à l'autre.

Le tour du cavalier



Le tour du cavalier : solution (Euler)



Graphe hamiltonien

Soit $G = (V, E, \varphi)$ un graphe orienté ou non. Un chemin est dit **hamiltonien** si il emprunte une fois et une seule chaque sommet. Un cycle est dit **hamiltonien** s'il est élémentaire et passe par tous les sommets.

Graphe hamiltonien

Soit $G = (V, E, \varphi)$ un graphe orienté ou non. Un chemin est dit **hamiltonien** si il emprunte une fois et une seule chaque sommet. Un cycle est dit **hamiltonien** s'il est élémentaire et passe par tous les sommets.

Un graphe est dit **hamiltonien** s'il admet un cycle hamiltonien.

Quelques graphes hamiltoniens

Sont hamiltoniens :

- C_n pour $n \geq 3$;

Quelques graphes hamiltoniens

Sont hamiltoniens :

- C_n pour $n \geq 3$;
- K_n pour $n \geq 3$;

Quelques graphes hamiltoniens

Sont hamiltoniens :

- C_n pour $n \geq 3$;
- K_n pour $n \geq 3$;
- $K_{n,n}$ pour $n \geq 2$;

Quelques graphes hamiltoniens

Sont hamiltoniens :

- C_n pour $n \geq 3$;
- K_n pour $n \geq 3$;
- $K_{n,n}$ pour $n \geq 2$;
- Le graphe correspondant au tour du cavalier sur un échiquier 8×8...

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;
- Le graphe de Petersen ;

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;
- Le graphe de Petersen ;
- Le graphe correspondant au tour du cavalier sur un échiquier 4×4

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;
- Le graphe de Petersen ;
- Le graphe correspondant au tour du cavalier sur un échiquier 4×4
- Le graphe correspondant au tour du cavalier sur un échiquier $n \times m$, où n et m sont impairs...

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;
- Le graphe de Petersen ;
- Le graphe correspondant au tour du cavalier sur un échiquier 4×4
- Le graphe correspondant au tour du cavalier sur un échiquier $n \times m$, où n et m sont impairs...

Quelques graphes non hamiltoniens

Ne sont pas hamiltoniens :

- $K_{n,m}$ pour $n \neq m$;
- Le graphe de Petersen ;
- Le graphe correspondant au tour du cavalier sur un échiquier 4×4
- Le graphe correspondant au tour du cavalier sur un échiquier $n \times m$, où n et m sont impairs...

Remarque : si $|n - m| = 1$, alors $K_{n,m}$ n'est pas hamiltonien, mais admet un chemin hamiltonien.

Théorème de Dirac

Un graphe ayant suffisamment d'arêtes est hamiltonien :

Théorème (G.A. Dirac)

Soit G un graphe simple à $n \geq 3$ sommets. Supposons que tous les sommets sont de degré au moins $n/2$. Alors G est hamiltonien.

Démonstration du théorème de Dirac

La preuve est encore une fois algorithmique :

- on part d'un sommet quelconque.

Démonstration du théorème de Dirac

La preuve est encore une fois algorithmique :

- on part d'un sommet quelconque.
- on engendre un chemin élémentaire, jusqu'à ce que l'on ne puisse plus le rallonger (ni au début, ni à la fin).

Démonstration du théorème de Dirac

La preuve est encore une fois algorithmique :

- on part d'un sommet quelconque.
- on engendre un chemin élémentaire, jusqu'à ce que l'on ne puisse plus le rallonger (ni au début, ni à la fin).
- alors, on peut modifier ce chemin pour en faire un cycle élémentaire.

Démonstration du théorème de Dirac

La preuve est encore une fois algorithmique :

- on part d'un sommet quelconque.
- on engendre un chemin élémentaire, jusqu'à ce que l'on ne puisse plus le rallonger (ni au début, ni à la fin).
- alors, on peut modifier ce chemin pour en faire un cycle élémentaire.
- si ce cycle passe par tous les sommets, il est hamiltonien.

Démonstration du théorème de Dirac

La preuve est encore une fois algorithmique :

- on part d'un sommet quelconque.
- on engendre un chemin élémentaire, jusqu'à ce que l'on ne puisse plus le rallonger (ni au début, ni à la fin).
- alors, on peut modifier ce chemin pour en faire un cycle élémentaire.
- si ce cycle passe par tous les sommets, il est hamiltonien.
- sinon, prendre un sommet qui n'appartient pas à ce cycle, et le rajouter au cycle. On obtient un chemin. Revenir à la seconde étape.

Théorème de Schwenk

En général, déterminer si un graphe est hamiltonien est difficile. Même pour la marche du cavalier sur un échiquier $n \times m$, la réponse est subtile.

Théorème de Schwenk

En général, déterminer si un graphe est hamiltonien est difficile. Même pour la marche du cavalier sur un échiquier $n \times m$, la réponse est subtile.

Théorème (A. Schwenk)

Le graphe correspondant à la marche du cavalier sur un échiquier $n \times m$, avec $n \leq m$, est hamiltonien, sauf dans les cas suivants :

- n *et* m *sont impairs ;*
- $n \in \{1, 2, 4\}$ *(sauf si* $n = m = 1$*) ;*
- $n = 3$ *et* $m \in \{4, 6, 8\}$*.*

Matrice d'adjacence

Définition

Soit $G = (V, E, \varphi_+, \varphi_-)$ un graphe orienté à $r = |V|$ sommets. La **matrice d'adjacence** M de G est la matrice $r \times r$ dont le terme m_{ij} est le nombre d'arêtes partant de i et arrivant en j.

Définition

Soit $G = (V, E, \varphi_+, \varphi_-)$ un graphe orienté à $r = |V|$ sommets. La **matrice d'adjacence** M de G est la matrice $r \times r$ dont le terme m_{ij} est le nombre d'arêtes partant de i et arrivant en j.

On définit de même la matrice d'adjacence d'un graphe simple. Dans ce cas, la matrice d'adjacence est symétrique. Si le graphe est sans boucle, les termes diagonaux sont nuls.

Définition

Soit $G = (V, E, \varphi_+, \varphi_-)$ un graphe orienté à $r = |V|$ sommets. La **matrice d'adjacence** M de G est la matrice $r \times r$ dont le terme m_{ij} est le nombre d'arêtes partant de i et arrivant en j.

On définit de même la matrice d'adjacence d'un graphe simple. Dans ce cas, la matrice d'adjacence est symétrique. Si le graphe est sans boucle, les termes diagonaux sont nuls.

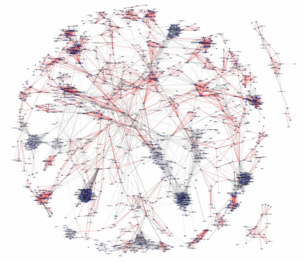
Pour des graphes généraux, les boucles peuvent compter pour 1 ou 2, suivant les conventions.

Quelques remarques

On peut faire l'opération inverse : à partir d'une matrice carrée à coefficients positifs, obtenir un graphe. C'est utile pour visualiser des données.

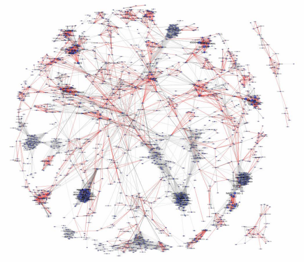
Quelques remarques

On peut faire l'opération inverse : à partir d'une matrice carrée à coefficients positifs, obtenir un graphe. C'est utile pour visualiser des données.



Quelques remarques

On peut faire l'opération inverse : à partir d'une matrice carrée à coefficients positifs, obtenir un graphe. C'est utile pour visualiser des données.



En pratique, les matrices d'adjacence peuvent être très grandes et avec beaucoup de zéros, ce qui est très cher en termes de mémoire et d'opérations.

Résultat principal

Théorème

Soit G un graphe simple ou orienté, de matrice d'adjacence M. Alors le nombre de chemins de longueur n allant de i à j est $(M^n)_{ij}$.

Résultat principal

Théorème

Soit G un graphe simple ou orienté, de matrice d'adjacence M. Alors le nombre de chemins de longueur n allant de i à j est $(M^n)_{ij}$.

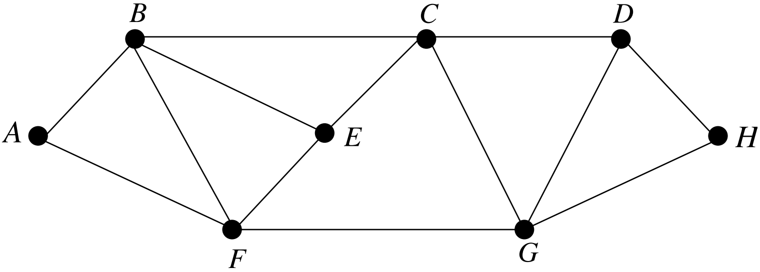
Résultat principal

Théorème

Soit G *un graphe simple ou orienté, de matrice d'adjacence* M*. Alors le nombre de chemins de longueur* n *allant de* i *à* j *est* $(M^n)_{ij}$*.*

Démonstration : récurrence sur n.

Un exemple : graphe



Un exemple : matrice d'adjacence

La matrice d'adjacence du graphe précédent, avec l'ordre A, B, C, D, E, F, G, H :

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

Un exemple : matrice d'adjacence

La matrice d'adjacence du graphe précédent, avec l'ordre A, B, C, D, E, F, G, H :

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

Combien y a-t-il de chemins de longueur 6 de A à H ? Combien y a-t-il de chemins de longueur 8 ?

Critère de positivité

Théorème

Soit G un graphe fortement connexe d'ordre $r \geq 2$, de matrice d'adjacence M. Il existe un entier N tel que M^n soit strictement positive pour tout $n \geq N$ si et seulement si G contient un cycle de longueur impaire.

Critère de positivité

Théorème

Soit G un graphe fortement connexe d'ordre $r \geq 2$, de matrice d'adjacence M. Il existe un entier N tel que M^n soit strictement positive pour tout $n \geq N$ si et seulement si G contient un cycle de longueur impaire.

Critère de positivité

Théorème

Soit G un graphe fortement connexe d'ordre $r \geq 2$, de matrice d'adjacence M. Il existe un entier N tel que M^n soit strictement positive pour tout $n \geq N$ si et seulement si G contient un cycle de longueur impaire.

Démonstration : exercice.

Théorème de Perron-Frobenius

Théorème (Corollaire du théorème de Perron-Frobenius)

Soit G un graphe fortement connexe d'ordre $r \geq 2$, admettant un cycle de longueur impaire.
Alors il existe $\lambda > 1$, $\theta \in [0,1)$ et une projection de rang 1 strictement positive C tels que, pour tous sommets i et j :

$$|\{\textit{chemins de longueur } n \textit{ de } i \textit{ à } j\}| = c_{ij}\lambda^n + O(\lambda^{\theta n}).$$

Théorème de Perron-Frobenius : application

On part du graphe à 2 sommets, une arête, une boucle.

Théorème de Perron-Frobenius : application

On part du graphe à 2 sommets, une arête, une boucle.
Sa matrice d'adjacence est :

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

Théorème de Perron-Frobenius : application

On part du graphe à 2 sommets, une arête, une boucle.
Sa matrice d'adjacence est :

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

On obtient :

$$C = \frac{1}{\sqrt{5}} \begin{pmatrix} \phi & 1 \\ 1 & \phi^{-1} \end{pmatrix} \text{ et } \lambda = \phi.$$

Matrice stochastique

Une **matrice stochastique** est une matrice carrée, à coefficients positifs, tel que la somme des coefficients de chaque ligne vaut 1.

Matrice stochastique

Une **matrice stochastique** est une matrice carrée, à coefficients positifs, tel que la somme des coefficients de chaque ligne vaut 1.

Un **graphe probabiliste** est un graphe orienté, tel que :

- chaque arête e est munie d'un nombre $p(e) \in [0, 1]$;
- pour tout sommet, la somme des poids des arêtes sortantes vaut 1.

On y associe naturellement un grahe probabiliste P.

Matrice stochastique

Une **matrice stochastique** est une matrice carrée, à coefficients positifs, tel que la somme des coefficients de chaque ligne vaut 1.

Un **graphe probabiliste** est un graphe orienté, tel que :

- chaque arête e est munie d'un nombre $p(e) \in [0, 1]$;
- pour tout sommet, la somme des poids des arêtes sortantes vaut 1.

On y associe naturellement un grahe probabiliste P.

Un graphe stochastique est **ergodique** si entre deux points il existe toujours un chemin dont les poids sont strictement positifs, et **aperiodique** si le PGCD des longueurs de ces chemins est 1.

Processus stochastique sur le graphe

Soit $G = (V, E, \varphi_-, \varphi_+, p)$ un graphe probabiliste. On définit un processus stochastique $(X_n)_{n\geq 0}$ sur V en choisissant X_0 au hasard, et pour chaque n,

$$\mathbb{P}(X_0, \ldots, X_n) = \mathbb{P}(X_0)p(X_0X_1)\ldots p(X_{n-1}X_n).$$

Processus stochastique sur le graphe

Soit $G = (V, E, \varphi_-, \varphi_+, p)$ un graphe probabiliste. On définit un processus stochastique $(X_n)_{n \geq 0}$ sur V en choisissant X_0 au hasard, et pour chaque n,

$$\mathbb{P}(X_0, \ldots, X_n) = \mathbb{P}(X_0)p(X_0X_1)\ldots p(X_{n-1}X_n).$$

A chaque étape, en X_n, on choisit le sommet suivant avec probabilité $p(X_n\cdot)$, indépendamment du passé.

Résultat principal pour les graphes stochastiques

Théorème

Soit G *un graphe probabiliste, et* $(X_n)_{n\geq 0}$ *le processus stochastique associé tel que* $X_0 = i$*.*
Alors $\mathbb{P}(X_n = j) = (P^n)_{ij}$.

Perron-Frobenius pour les graphes stochastiques

Théorème

Soit G un graphe stochastique ergodique. Alors il existe une unique mesure de probabilité stationnaire sur le graphe.

Perron-Frobenius pour les graphes stochastiques

Théorème

Soit G un graphe stochastique ergodique. Alors il existe une unique mesure de probabilité stationnaire sur le graphe.

Théorème

Soit G un graphe stochastique ergodique et apériodique. Soit $(X_n)_{n\geq 0}$ un processus stochastique associé. Alors la loi de X_n converge exponentiellement vite vers la mesure de probabilité stationnaire.

Algorithme de Dijkstra

Un exemple

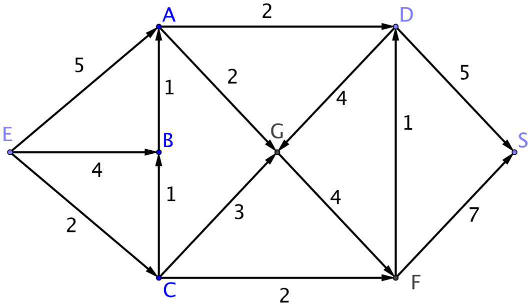


FIGURE : Un graphe pondéré

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ?
L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

Son algorithme est polynomial en V et E, quoique trop lent pour certaines applications pratiques (GPS...).

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

Son algorithme est polynomial en V et E, quoique trop lent pour certaines applications pratiques (GPS...).

On peut supposer le graphe simple, éventuellement orienté. On se donne à chaque étape n :

- un ensemble de sommets visités X, de cardinal n ;

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

Son algorithme est polynomial en V et E, quoique trop lent pour certaines applications pratiques (GPS...).

On peut supposer le graphe simple, éventuellement orienté. On se donne à chaque étape n :

- un ensemble de sommets visités X, de cardinal n ;
- pour tout $v \in V$, une fonction $\pi(v)$, le **coût provisoire**.

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

Son algorithme est polynomial en V et E, quoique trop lent pour certaines applications pratiques (GPS...).

On peut supposer le graphe simple, éventuellement orienté. On se donne à chaque étape n :

- un ensemble de sommets visités X, de cardinal n ;
- pour tout $v \in V$, une fonction $\pi(v)$, le **coût provisoire**.

L'algorithme

Quel est le chemin le plus court entre deux sommets a et s ? L'algorithme de Dijkstra (E.W. Dijkstra, 1956) permet de le déterminer.

Idée : pour tout $n \geq 1$, déterminer les n sommets les plus proches de a, et leur distance à a. En faisant croître n, on finit par englober s.

Son algorithme est polynomial en V et E, quoique trop lent pour certaines applications pratiques (GPS...).

On peut supposer le graphe simple, éventuellement orienté. On se donne à chaque étape n :

- un ensemble de sommets visités X, de cardinal n ;
- pour tout $v \in V$, une fonction $\pi(v)$, le **coût provisoire**.

Si $v \in X$, alors $\pi(v) = d(a, v)$; sinon, $\pi(v) \geq d(a, v)$.

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

À l'étape n :

- on cherche $v_0 \in \partial X$ minimisant π ;

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

À l'étape n :

- on cherche $v_0 \in \partial X$ minimisant π ;
- on pose $X \leftarrow X \cup \{v_0\}$;

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

À l'étape n :

- on cherche $v_0 \in \partial X$ minimisant π ;
- on pose $X \leftarrow X \cup \{v_0\}$;
- pour tout $v \in \partial X \cap \partial\{v_0\}$, on pose

$$\pi(v) \leftarrow \min\{\pi(v), \pi(v_0) + p(v_0 v)\}.$$

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

À l'étape n :

- on cherche $v_0 \in \partial X$ minimisant π ;
- on pose $X \leftarrow X \cup \{v_0\}$;
- pour tout $v \in \partial X \cap \partial\{v_0\}$, on pose

$$\pi(v) \leftarrow \min\{\pi(v), \pi(v_0) + p(v_0v)\}.$$

L'algorithme : initialisation et récurrence

À la première étape : $X = \{a\}$, $\pi(a) = 0$, $\pi(bv) = p(av)$ si $\{a, v\}$ est une arête, et $\pi(v) = +\infty$ sinon.

À l'étape n :

- on cherche $v_0 \in \partial X$ minimisant π ;
- on pose $X \leftarrow X \cup \{v_0\}$;
- pour tout $v \in \partial X \cap \partial\{v_0\}$, on pose

$$\pi(v) \leftarrow \min\{\pi(v), \pi(v_0) + p(v_0v)\}.$$

On s'arête quand $v_0 = b$ (auquel cas $d(a, b) = f(b)$) ou sinon quand $\partial X = \emptyset$ (auquel cas b n'est pas dans la même composante connexe du graphe que a).

L'algorithme : exemple

	A	B	C	D	E	F	G	S
π_0	∞	∞	∞	∞	0	∞	∞	∞
π_1	5	4	2	∞		∞	∞	∞
π_2	5	3		∞		4	5	∞
π_3	4			∞		4	5	∞
π_4				6		4	5	∞
π_5				5			5	11
π_6							5	11
π_7								10

Distance entre E et S : 10.

Chemin le plus court

Pour trouver le chemin le plus court, on garde en plus une information $\gamma : X \cup \partial X \to X$: là d'où provient le plus court chemin. On l'initialise en posant $\gamma(b) = a$ pour $b \in \{a\} \cup \partial\{a\}$

À chaque étape, pour tout $v \in \partial X \cap \partial\{v_0\}$, on considère la quantité $\min\{\pi(v), \pi(v_0) + p(v_0 v)\}$.

- si le minimum est réalisé par $\pi(v)$: $\gamma(v) \leftarrow \gamma(v)$.

Chemin le plus court

Pour trouver le chemin le plus court, on garde en plus une information $\gamma : X \cup \partial X \to X$: là d'où provient le plus court chemin. On l'initialise en posant $\gamma(b) = a$ pour $b \in \{a\} \cup \partial\{a\}$

À chaque étape, pour tout $v \in \partial X \cap \partial\{v_0\}$, on considère la quantité $\min\{\pi(v), \pi(v_0) + p(v_0 v)\}$.

- si le minimum est réalisé par $\pi(v)$: $\gamma(v) \leftarrow \gamma(v)$.
- si le minimum est réalisé par $\pi(v_0) + p(v_0 v)$: $\gamma(v) \leftarrow v_0$.

Chemin le plus court

Pour trouver le chemin le plus court, on garde en plus une information $\gamma : X \cup \partial X \to X$: là d'où provient le plus court chemin. On l'initialise en posant $\gamma(b) = a$ pour $b \in \{a\} \cup \partial\{a\}$

À chaque étape, pour tout $v \in \partial X \cap \partial\{v_0\}$, on considère la quantité $\min\{\pi(v), \pi(v_0) + p(v_0v)\}$.

- si le minimum est réalisé par $\pi(v)$: $\gamma(v) \leftarrow \gamma(v)$.
- si le minimum est réalisé par $\pi(v_0) + p(v_0v)$: $\gamma(v) \leftarrow v_0$.

Chemin le plus court

Pour trouver le chemin le plus court, on garde en plus une information $\gamma : X \cup \partial X \to X$: là d'où provient le plus court chemin. On l'initialise en posant $\gamma(b) = a$ pour $b \in \{a\} \cup \partial\{a\}$

À chaque étape, pour tout $v \in \partial X \cap \partial\{v_0\}$, on considère la quantité $\min\{\pi(v), \pi(v_0) + p(v_0 v)\}$.

- si le minimum est réalisé par $\pi(v)$: $\gamma(v) \leftarrow \gamma(v)$.
- si le minimum est réalisé par $\pi(v_0) + p(v_0 v)$: $\gamma(v) \leftarrow v_0$.

Pour retrouver le plus court chemin, il suffit de partir du point d'arrivée b et de suivre γ jusqu'au point de départ, puis d'inverser le chemin obtenu.

Chemin le plus court : exemple

	A	B	C	D	E	F	G	S
π_0, γ_0	∞	∞	∞	∞	$0, E$	∞	∞	∞
π_1, γ_1	$5, E$	$4, E$	$2, E$	∞		∞	∞	∞
π_2, γ_2	$5, E$	$3, C$		∞		$4, C$	$5, C$	∞
π_3, γ_3	$4, B$			∞		$4, C$	$5, C$	∞
π_4, γ_4				$6, A$		$4, C$	$5, C$	∞
π_5, γ_5				$5, F$			$5, C$	$11, F$
π_6, γ_6							$5, C$	$11, F$
π_7, γ_7								$10, D$

Chemin minimal : $ECFDS$.

Démonstration de l'algorithme

On veut montrer que l'algorithme fournit bien le plus court chemin de a à b. On procède par récurrence : à chaque étape n,

- pour $v \in X$, on a $\pi(v) = d(a, v)$;

Démonstration de l'algorithme

On veut montrer que l'algorithme fournit bien le plus court chemin de a à b. On procède par récurrence : à chaque étape n,

- pour $v \in X$, on a $\pi(v) = d(a, v)$;
- pour $v \in V \setminus X$, la fonction $\pi(v)$ est la longueur du plus court chemin de a à v dans X.

Démonstration de l'algorithme

On veut montrer que l'algorithme fournit bien le plus court chemin de a à b. On procède par récurrence : à chaque étape n,

- pour $v \in X$, on a $\pi(v) = d(a, v)$;
- pour $v \in V \setminus X$, la fonction $\pi(v)$ est la longueur du plus court chemin de a à v dans X.
- X est un ensemble des $|X|$ sommets les plus proches de a.

Coloration des graphes

Nombre chromatique

Soit G un graphe simple. On appelle **coloriage** de G la donnée d'un ensemble C et d'une application $f : V \to C$ tels que $f(x) \neq f(y)$ si x et y sont adjacents.

Nombre chromatique

Soit G un graphe simple. On appelle **coloriage** de G la donnée d'un ensemble C et d'une application $f : V \to C$ tels que $f(x) \neq f(y)$ si x et y sont adjacents.

Les éléments de C sont appelés les **couleurs**.

Nombre chromatique

Soit G un graphe simple. On appelle **coloriage** de G la donnée d'un ensemble C et d'une application $f : V \to C$ tels que $f(x) \neq f(y)$ si x et y sont adjacents.

Les éléments de C sont appelés les **couleurs**.

Le **nombre chromatique** de G, noté $\chi(G)$, est le nombre minimal de couleurs nécessaires pour colorier G.

Graphes usuels

- $\chi(K_n) = n$
- $\chi(K_{n,m}) = 2$
- $\chi(C_n)$ vaut 2 si n est pair, et 3 si n est impair.

Graphes usuels

- $\chi(K_n) = n$
- $\chi(K_{n,m}) = 2$
- $\chi(C_n)$ vaut 2 si n est pair, et 3 si n est impair.

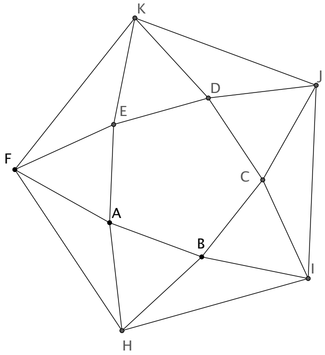
Exercice : Les graphes complets sont les seuls à vérifier $\chi(G) = |V|$.

Sous-graphes

Si H est un sous-graphe de G, alors $\chi(H) \leq \chi(G)$. C'est le plus souvent ainsi qu'on minore $\chi(G)$. $\chi(G)$ est supérieur ou égal à l'ordre du plus grand sous-graphe complet de G.

Sous-graphes

Si H est un sous-graphe de G, alors $\chi(H) \leq \chi(G)$. C'est le plus souvent ainsi qu'on minore $\chi(G)$. $\chi(G)$ est supérieur ou égal à l'ordre du plus grand sous-graphe complet de G.



Majoration du nombre chromatique

On note $r(G)$ le maximum des degrés de G. Dans ce qui suit, G est un graphe simple fini.

Théorème

$\chi(G) \leq r(G) + 1.$

Majoration du nombre chromatique

On note $r(G)$ le maximum des degrés de G. Dans ce qui suit, G est un graphe simple fini.

Théorème

$\chi(G) \leq r(G) + 1$.

Théorème (R.L. Brooks, 1941)

$\chi(G) \leq r(G)$, *sauf si* G *est un graphe complet ou un cycle d'ordre impair.*

Majoration du nombre chromatique

On note $r(G)$ le maximum des degrés de G. Dans ce qui suit, G est un graphe simple fini.

Théorème

$\chi(G) \leq r(G) + 1$.

Théorème (R.L. Brooks, 1941)

$\chi(G) \leq r(G)$, *sauf si* G *est un graphe complet ou un cycle d'ordre impair.*

Majoration du nombre chromatique

On note $r(G)$ le maximum des degrés de G. Dans ce qui suit, G est un graphe simple fini.

Théorème

$\chi(G) \leq r(G) + 1$.

Théorème (R.L. Brooks, 1941)

$\chi(G) \leq r(G)$, sauf si G est un graphe complet ou un cycle d'ordre impair.

En général, on peut faire beaucoup mieux, mais on ne connaît pas d'algorithme efficace qui donne le nombre chromatique exact.

Cas particulier du théorème de Brooks

On peut montrer assez facilement le théorème de Brooks pour des graphes connexes non réguliers.

- récurrence forte sur le nombre de sommet. Pour $n = 3$: il y a un seul graphe connexe non régulier, qui vérifie le théorème.

Cas particulier du théorème de Brooks

On peut montrer assez facilement le théorème de Brooks pour des graphes connexes non réguliers.

- récurrence forte sur le nombre de sommet. Pour $n = 3$: il y a un seul graphe connexe non régulier, qui vérifie le théorème.
- pour $n \geq 4$: on enlève un sommet de degré $< r(G)$. Il suffit de montrer le résultat pour chaque composante connexe obtenue.

Cas particulier du théorème de Brooks

On peut montrer assez facilement le théorème de Brooks pour des graphes connexes non réguliers.

- récurrence forte sur le nombre de sommet. Pour $n = 3$: il y a un seul graphe connexe non régulier, qui vérifie le théorème.
- pour $n \geq 4$: on enlève un sommet de degré $< r(G)$. Il suffit de montrer le résultat pour chaque composante connexe obtenue.
- chaque composante connexe est ou bien d-régulière pour $d < r(G)$, ou bien non régulière.

Algorithme glouton

Les algorithme exacts connus ont un temps d'exécution exponentiels en la taille du graphe. On voudrait un algorithme moins exact, mais plus rapide.

Algorithme glouton

Les algorithme exacts connus ont un temps d'exécution exponentiels en la taille du graphe. On voudrait un algorithme moins exact, mais plus rapide.

L'**algorithme glouton** procède en coloriant les sommets successivement.

Algorithme glouton

Les algorithme exacts connus ont un temps d'exécution exponentiels en la taille du graphe. On voudrait un algorithme moins exact, mais plus rapide.

L'**algorithme glouton** procède en coloriant les sommets successivement.

On se donne pour ensemble de couleurs $\mathbb{N}$.

- on ordonne les sommets arbitrairement.

Algorithme glouton

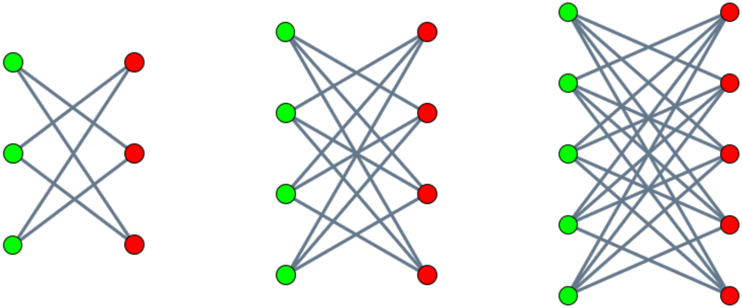
Les algorithme exacts connus ont un temps d'exécution exponentiels en la taille du graphe. On voudrait un algorithme moins exact, mais plus rapide.

L'**algorithme glouton** procède en coloriant les sommets successivement.

On se donne pour ensemble de couleurs $\mathbb{N}$.

- on ordonne les sommets arbitrairement.
- à l'étape n, on colore le sommet n par la plus petite couleur non utilisée par ses voisins déjà coloriés.

Algorithme glouton : échec



Algorithme de Welsh-Powell

L'**algorithme de Welsh-Powell** procède en utilisant chaque couleur successivement.

Algorithme de Welsh-Powell

L'**algorithme de Welsh-Powell** procède en utilisant chaque couleur successivement.
On peut le rendre plus efficace en ordonnant les sommets intelligemment.

Algorithme de Welsh-Powell

L'**algorithme de Welsh-Powell** procède en utilisant chaque couleur successivement.
On peut le rendre plus efficace en ordonnant les sommets intelligemment.

- on ordonne les sommets par degré décroissant.

Algorithme de Welsh-Powell

L'**algorithme de Welsh-Powell** procède en utilisant chaque couleur successivement.
On peut le rendre plus efficace en ordonnant les sommets intelligemment.

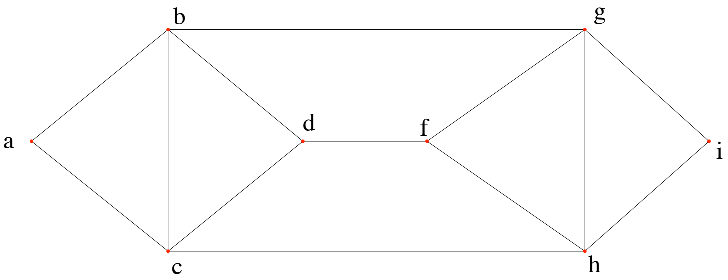
- on ordonne les sommets par degré décroissant.
- on procède par couleur croissante.

Algorithme de Welsh-Powell

L'**algorithme de Welsh-Powell** procède en utilisant chaque couleur successivement.
On peut le rendre plus efficace en ordonnant les sommets intelligemment.

- on ordonne les sommets par degré décroissant.
- on procède par couleur croissante.
- pour chaque couleur, on parcourt la liste des sommets, en coloriant les sommets dès que possible et en respectant les incompatibilités.

Algorithme de Welsh-Powell : exemple



Algorithme de Welsh-Powell : raffinements

Pour améliorer l'efficacité de l'algorithme de Welsh-Powell, on peut :

- mieux ordonner les sommets : en cas d'égalité sur les degrés, regarder les degrés des voisins (e.g. leur somme).

Algorithme de Welsh-Powell : raffinements

Pour améliorer l'efficacité de l'algorithme de Welsh-Powell, on peut :

- mieux ordonner les sommets : en cas d'égalité sur les degrés, regarder les degrés des voisins (e.g. leur somme).
- avoir une version dynamique : à chaque nouvelle couleur, effacer les sommets déjà coloriés et réordonner le graphe.

Algorithme de Welsh-Powell : raffinements

Pour améliorer l'efficacité de l'algorithme de Welsh-Powell, on peut :

- mieux ordonner les sommets : en cas d'égalité sur les degrés, regarder les degrés des voisins (e.g. leur somme).
- avoir une version dynamique : à chaque nouvelle couleur, effacer les sommets déjà coloriés et réordonner le graphe.

Algorithme de Welsh-Powell : raffinements

Pour améliorer l'efficacité de l'algorithme de Welsh-Powell, on peut :

- mieux ordonner les sommets : en cas d'égalité sur les degrés, regarder les degrés des voisins (e.g. leur somme).
- avoir une version dynamique : à chaque nouvelle couleur, effacer les sommets déjà coloriés et réordonner le graphe.

Exercice : appliquer la version dynamique de Welsh-Powell à l'exemple précédent.

Exemple 1 : carte d'Amérique du Sud

De combien de couleurs a-t-on besoin pour colorier cette carte de telle sorte que deux pays voisins aient toujours des couleurs différentes ?

Figure : Carte d'Amérique du Sud

Théorème des quatre couleurs

Théorème (Théorème des quatre couleurs (Appel, Haken, 1976))

Si G est un graphe planaire, alors $\chi(G) \leq 4$.

Théorème des quatre couleurs

Théorème (Théorème des quatre couleurs (Appel, Haken, 1976))

Si G est un graphe planaire, alors $\chi(G) \leq 4$.

Théorème des quatre couleurs

Théorème (Théorème des quatre couleurs (Appel, Haken, 1976))

Si G est un graphe planaire, alors $\chi(G) \leq 4$.

On peut colorier toute carte en utilisant au plus quatre couleurs (à condition que les pays soient d'un seul tenant).

Exemple 2 : les examens

Cinq étudiants, Alice, Benoît, Charles, Dorothée, Estelle doivent passer des oraux de rattrapage d'une heure. Précisément, Alice doit passer devant le jury 1, Benoît devant le 2, Charles devant les jurys 2 et 3, Dorothée devant les 1 et 3 et Estelle devant les trois jurys 1,2,3.

Comment organiser la session pour qu'elle dure le moins longtemps possible ?

Exemple : modélisation en termes de graphes

- Sommets : couples étudiants-jurys ($A1$, $B2$, $C2$, $C3$, $D1$, $D3$, $E1$, $E2$, $E3$) ;

Exemple : modélisation en termes de graphes

- Sommets : couples étudiants-jurys ($A1$, $B2$, $C2$, $C3$, $D1$, $D3$, $E1$, $E2$, $E3$) ;
- Arêtes : incompatibilités (couples Mi, Mj ou Pi, Qi) ;

Exemple : modélisation en termes de graphes

- Sommets : couples étudiants-jurys ($A1$, $B2$, $C2$, $C3$, $D1$, $D3$, $E1$, $E2$, $E3$) ;
- Arêtes : incompatibilités (couples Mi, Mj ou Pi, Qi) ;
- Couleurs : horaires.

Exemple : modélisation en termes de graphes

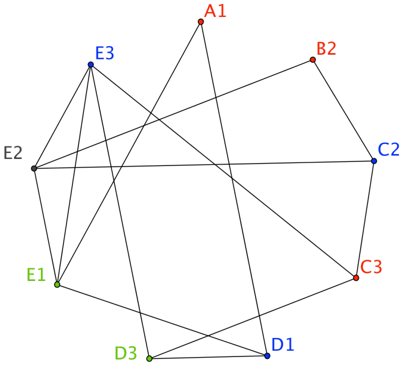
- Sommets : couples étudiants-jurys $(A1$, $B2$, $C2$, $C3$, $D1$, $D3$, $E1$, $E2$, $E3)$;
- Arêtes : incompatibilités (couples Mi, Mj ou Pi, Qi) ;
- Couleurs : horaires.

Exemple : modélisation en termes de graphes

- Sommets : couples étudiants-jurys ($A1$, $B2$, $C2$, $C3$, $D1$, $D3$, $E1$, $E2$, $E3$) ;
- Arêtes : incompatibilités (couples Mi, Mj ou Pi, Qi) ;
- Couleurs : horaires.

Un coloriage donne une heure de passage pour chaque couple, sans incompatibilités.

Exemple 2 : coloriages



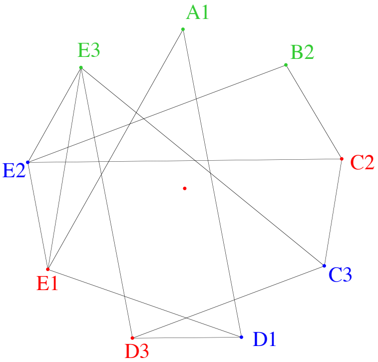


FIGURE : Coloriages du graphe des examens, à gauche par le glouton associé à l'ordre lexicographique, à droite par Welsh-Powell.