Parcours de graphes

Michel Habib
MPRI 2009

28 mars 2011

Plan

Le problème

Étant donné un graphe $G = (V, E)$ (non-orienté),
explorer l'ensemble des sommets G en "traversant" les arêtes du graphe.

Résultat

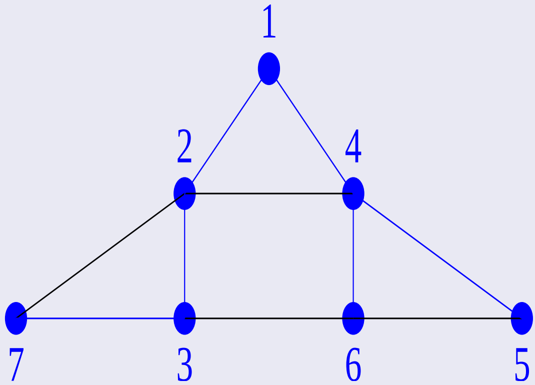
- Un arbre de parcours
- Un ordre total sur les sommets du graphe

Questions

- À quelle condition un ordre σ sur les sommets correspond à un parcours ?
- Quelles sont les propriétés de ces ordres ?

Référence principale :

Ces questions simples n'ont été posées que très récemment :
D.G. Corneil et R. M. Krueger, A unified view of graph searching, SIAM J. Discrete Math, 22, N°4 (2008) 1259-1276



Invariant

À chaque étape, on sélectionne une arête entre un sommet visité et un non-visité

Parcours générique

$S \leftarrow \{s\}$

pour $i \leftarrow 1$ **à** n **faire**

  Extraire un sommet non-numéroté v de S

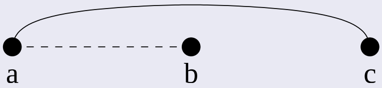
  $\sigma(i) \leftarrow v$

  pour chaque *sommet non-numéroté* $w \in N(v)$ **faire**

    $S \leftarrow S \cup \{w\}$

Question ?

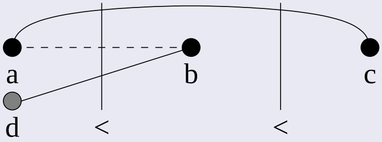
Soient a, b et c trois sommets tels que $ab \notin E$ et $ac \in E$.



À quelle condition est-il possible de visiter a puis b puis c ?

Propriété (Gen)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $d < b$ et $db \in E$



Théorème

Pour un graphe $G = (V, E)$, un ordre σ sur V est un parcours de G ssi σ vérifie la propriété (Gen).

Parcours en largeur (BFS)

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

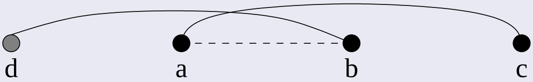
Initialiser la file S à s

pour $i \leftarrow 1$ **à** n **faire**

- Extraire le sommet v de la tête de la file S
- $\sigma(i) \leftarrow v$
- **pour chaque** *sommet non-numéroté* $w \in N(v)$ **faire**
  - **si** w *n'est pas dans* S **alors**
    - Ajouter w en fin de la file S

Propriété (B)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $d < a$ et $db \in E$



Théorème

Pour un graphe $G = (V, E)$, un ordre σ sur V est un parcours BFS de G ssi σ vérifie la propriété (B).

Applications

Distances à la source

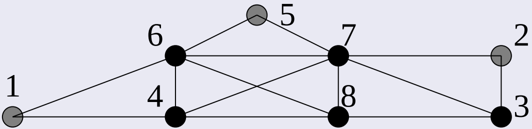
Soient σ un BFS de G à partir de s et x, y 2 sommets tels que $d(s, x) < d(s, y)$, alors $\sigma(x) < \sigma(y)$.

Evaluation du diamètre [CDK'03]

Si G ne contient pas de cycle sans corde de longueur supérieure ou égale à k, alors BFS permet de trouver un sommet x tel que $ecc(x) \geq Diam(G) - \lfloor k/2 \rfloor$.

Graphe triangulé

Un graphe est triangulé ssi tout cycle de longueur ≥ 4 possède une corde.

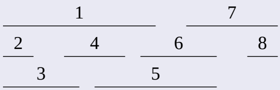


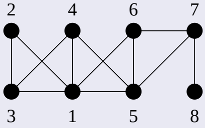
Diamètre [CDK'03]

Si v est le dernier sommet d'un ordre BFS sur un graphe triangulé G, alors $ecc(v) \geq Diam(G) - 1$.

Graphes d'intervalles

Un graphe est un graphe d'intervalles ssi c'est le graphe d'intersection d'une famille d'intervalles sur la droite réelle.





Diamètre [Corneil, Dragan, Khoëler 2003]

- Si v est le dernier sommet d'un ordre BFS sur un graphe d'intervalles G, alors $ecc(v) \geq Diam(G) - 1$.
- Il est possible de trouver dans la dernière couche, un sommet v tel que $ecc(v) = Diam(G)$

Parcours en largeur lexicographique (LexBFS)

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

Affecter l'étiquette $\emptyset$ à chaque sommet

$label(s) \leftarrow \{n\}$

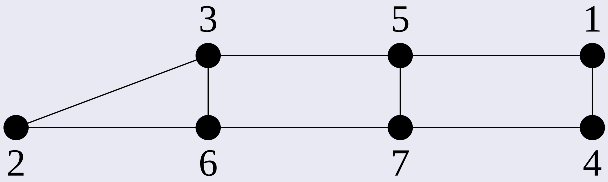
pour $i \leftarrow n$ **à** 1 **faire**

  Choisir un sommet v d'étiquette lexicographique max.

  $\sigma(i) \leftarrow v$

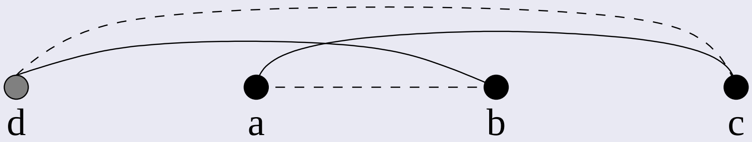
  pour chaque *sommet non-numéroté* $w \in N(v)$ **faire**

    $label(w) \leftarrow label(w).\{i\}$


3
5
1
2
6
7
4

Propriété (LexB)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $d < a$ et $db \in E$ et $dc \notin E$.



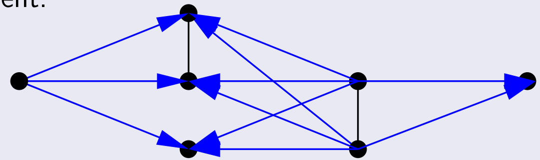
Théorème

Pour un graph $G = (V, E)$, un ordre σ sur V est un parcours LexBFS de G ssi σ vérifie la propriété (LexB).

Orientation transitive

Graphe de comparabilité et orientation transitive

Un graphe est de comparabilité ssi ses arêtes peuvent être orientées transitivement.

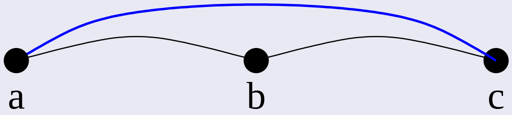


LexBFS [Habib, McConnel, Paul, Viennot 1998

Le dernier sommet d'un LexBFS réalisé sur le complémentaire d'un graphe de comparabilité G peut-être pris comme une source d'une orientation transitive de G.

Extension linéaire

Un graphe G est de comparabilité ssi il existe un ordre sur V tel que pout tout triplet $a < b < c$ tel que $ab \in E$ et $bc \in E$, alors $ac \in E$



Théorème : D.G. Corneil

Si G est de comparabilité, alors il existe un ordre LexBFS qui est une extension linéaire.

Problème

Comment calculer efficacement cet ordre LexBFS particulier ?

Parcours en profondeur (DFS)

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

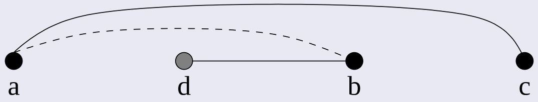
Initialiser la pile S à s

pour $i \leftarrow 1$ **à** n **faire**

- Extraire le sommet v du haut de la pile S
- $\sigma(i) \leftarrow v$
- **pour chaque** *sommet non-numéroté* $w \in N(v)$ **faire**
  - **si** w *n'est pas dans* S **alors**
    - Ajouter w en haut de la pile S

Propriété (D)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $a < d < b$ et $db \in E$.



Théorème

Pour un graphe $G = (V, E)$, un ordre σ sur V est un parcours DFS de G ssi σ vérifie la propriété (D).

Quelques exemples d'application

- Test de planarité (utilisation d'un DFS pour placer les arêtes autour de l'arbre associé au parcours).
- Composantes 2-connexes (resp. composantes fortement connexes dans le cas des graphes orientés).
- Tri topologique (extension linéaire) des graphes sans circuits, applications aux méchanismes d'héritage. . . .

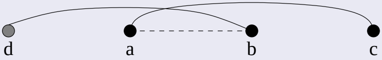
LexDFS

Peut-on définir un parcours en profondeur lexicographique ?

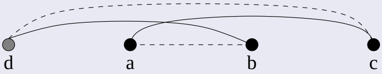
1. Parcours Générique
2. Parcours en largeur
   - BFS classique
   - LexBFS
3. Parcours en profondeur
   - DFS classique
   - DFS lexicographique
4. Autres parcours
   - Parcours selon le voisinage maximal MNS
   - Maximal Cardinality Search
5. Applications aux graphes triangulés
   - Les parcours *

BFS vs LexBFS

BFS

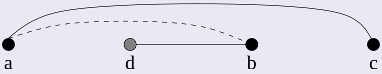


LexBFS

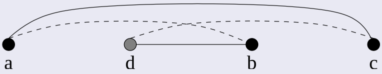


DFS vs LexDFS

DFS

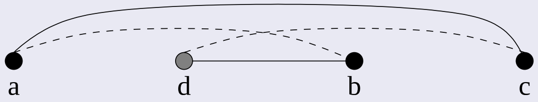


LexDFS



Propriété (LD)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $a < d < b$ et $db \in E$ et $dc \notin E$.



Théorème

Pour un graphe $G = (V, E)$, un ordre σ sur V est un parcours LexDFS de G ssi σ vérifie la propriété (LD).

LexDFS

Parcours en profondeur lexicographique (LexDFS)

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

Affecter l'étiquette $\emptyset$ à chaque sommet

$label(s) \leftarrow \{1\}$

pour $i \leftarrow 1$ **à** n **faire**

- Choisir un sommet v d'étiquette lexicographique max.
- $\sigma(i) \leftarrow v$
- **pour chaque** *sommet non-numéroté* $w \in N(v)$ **faire**
  - $label(w) \leftarrow \{i\}.label(w)$

LexDFS

Complexité

Peut-on implémenter LexDFS en $O(n+m)$?

Krueger et Spinrad 2008

Il est possible d'implémenter LexDFS en $O(n + mloglogn)$ en utilisant des arbres de Van der Boas.

Applications D. Corneil, M.H. 2009

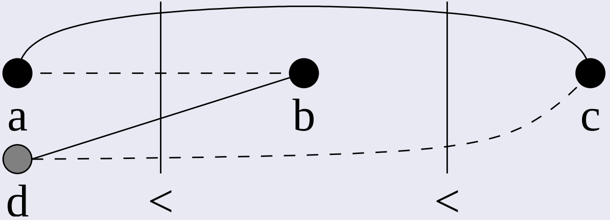
On peut calculer l'existence d'une chaîne hamiltonienne sur un graphe de cocomparabilité en utilisant LexDFS.

Pourquoi LexDFS n'est pas trivialement linéaire

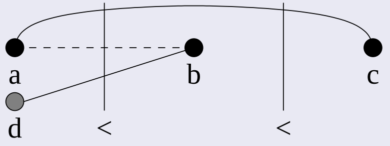
À partir de $P = \{X_1, X_2, \ldots X_k\ \}$ partition ordonnée en k parties et d'un ensemble pivot S et il faut obtenir en $O(|S|)$:
la partition ordonnée
$Q = \{S \cap X_1, S \cap X_2, \ldots, S \cap\ X_k, X_1 - S, \ldots, X_k - S\}$
Certains de ces ensembles pouvant être vides.

Propriété (MNS)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet d tel que $d < b$, $db \in E$ et $dc \notin E$.



Parcours générique



Le parcours suivant le voisinage maximal est donc le complété du parcours générique (par analogie à LexBFS (resp. LesDFS) qui sont les complétés de BFS (resp. DFS)). Ainsi MNS fut initialement appelé LexGen.

Parcours MNS

Parcours par voisinage maximal (MNS)

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

Affecter l'étiquette $\emptyset$ à chaque sommet

$label(s) \leftarrow \{n\}$

pour $i \leftarrow n$ **à** 1 **faire**

  Choisir un sommet v d'étiquette maximal pour l'inclusion.

  $\sigma(i) \leftarrow v$

  pour chaque *sommet non-numéroté* $w \in N(v)$ **faire**

    $label(w) \leftarrow \{i\} \cup label(w)$

Récapitulatif

D'après le configurations interdites nous avons les inclusions :

Inclusions

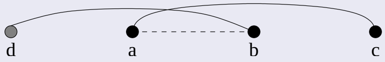
$$\begin{tikzcd}
 & \text{Gen} & \\
\text{BFS} \arrow[ur] & \text{MNS} \arrow[u] & \text{DFS} \arrow[ul] \\
\text{LexBFS} \arrow[u] \arrow[ur] & \text{MCS} \arrow[u] & \text{LexDFS} \arrow[ul] \arrow[u]
\end{tikzcd}$$

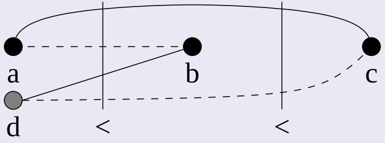
Exercices

1. Vérifier que les inclusions sont bien strictes.
2. Que peut-on dire d'un parcours à la fois BFS et MNS ?
   A-t-on BFS+MNS =LexBFS ?
3. Idem pour DFS et MNS
   A-t-on DFS + MNS = LexDFS ?

BFS + MNS

BFS

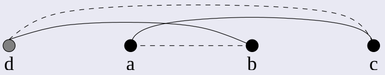




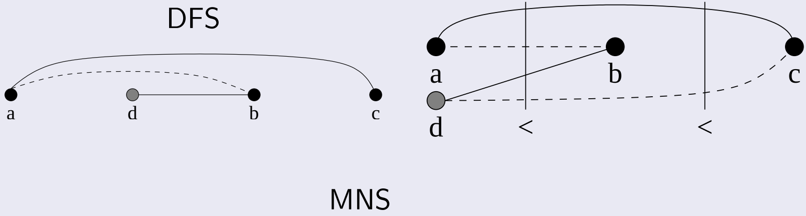
MNS

= LexBFS ?

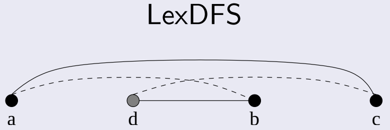
LexBFS



DFS + MNS



= LexDFS ?



A classification

Given the usual paradigm of an ordering ¡ of V, where $a < b < c, ac \in E$, $ab \notin E$, BFS, DFS and generic search all require that there is a neighbour of b that is in a certain location with respect to a and b.

The lexicographic versions of these searches (LBFS, LDFS and MNS) all require that there is a private neighbour of b (with respect to c) that is in the appropriate location with respect to a and c.

Layer ordering or distance level ordering

un ordre préservant les distances au sommet initial
$\neq$ BFS

Maximal Cardinality Search

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

Affecter l'étiquette 0 à chaque sommet

$label(s) \leftarrow \{n\}$

pour $i \leftarrow n$ **à** 1 **faire**

  Choisir un sommet v d'étiquette maximum

  $\sigma(i) \leftarrow v$

  pour chaque *sommet non-numéroté* $w \in N(v)$ **faire**

    $label(w) \leftarrow label(w) + 1$

Propriété

$MCS \subseteq MNS$

Car un voisinage maximal au sens du cardinal est nécessairement maximal au sens de l'inclusion.

Applications

Recognition

Definition

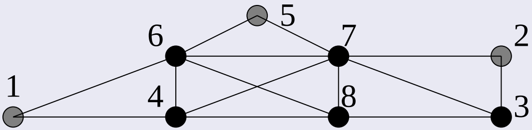
A graph is chordal iff it has no chordless cycle of length ≥ 4.

Minimal Separators

A subset of vertices S is a minimal separator if G
if there exist $a, b \in G$ such that a and b are not connected in $G - S$.
and S is minimal for inclusion with this property .

Sommets simpliciaux

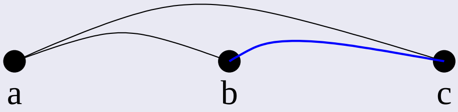
Definition



Un sommet est simplicial si son voisinage est une clique.

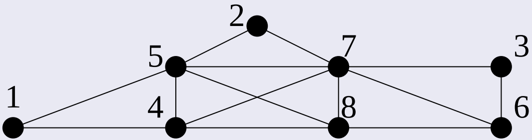
Ordre d'élimination simplicial

$\sigma = [x_1 \ldots x_i \ldots x_n]$ est un ordre d'élimination simplicial si x_i est simplicial dans le sous-graphe $G_i = G[\{x_i \ldots x_n\}]$



Caractérisation

Un graphe est triangulé ssi il possède un ordre d'élimination simplicial.



Preuve :

1. La classe des graphes triangulés est héréditaire.
2. Tout séparateur minimal est une clique
3. Si G n'est pas une clique, alors il admet au moins deux sommets simpliciaux.

Théorème [Tarjan et Yannakakis, 1984]

G est un graphe chordal ssi tout parcours MNS fournit un schéma simplicial.

Preuve :

Soit c le premier sommet non simplicial à gauche. Il existe donc $a < b \in N(c)$ avec $ab \notin E$. D'après la caractérisation des ordres MNS, il existe $d < b$ vérifiant $db \in E$ et $dc \notin E$. Comme G est chordal, nécessairement $ad \notin E$.
Soit $d < a$, dans ce cas, en considérant le triplet d, a, b, il existe nécessairement $d' < a$ tel que $d'a \in E$ et $d'b \notin E$. En outre $d'd \notin E$.
Soit $a < d$, en considérant le triplet a, d, c, il existe nécessairement $d' < d$ tel que $d'd \in E$ et $d'c \notin E$. En outre $ad' \notin E$.
Dans les deux cas une figure se propage strictement sur la gauche, d'où la contradiction.

Corollaire

G est un graphe chordal ssi tout parcours Maximal Cardinality Search (MCS) (resp LexBFS, LexDFS) fournit un schéma simplicial.

Parcours Générique *

Parcours *

Données: Un graphe $G = (V, E)$ et un sommet source s

Résultat: Un ordre total σ de V

Affecter l'étiquette $\emptyset$ à chaque sommet

$label(s) \leftarrow \{n\}$

pour $i \leftarrow n$ **à** 1 **faire**

  Choisir un sommet v suivant la règle du parcours mais appliquée à une composante connexe de G-{les sommets numérotés}.

  $\sigma(i) \leftarrow v$

  pour chaque *sommet non-numéroté* $w \in N(v)$ **faire**

    Modifier l'étiquette de w suivant la règle du parcours

Définition

Nous pouvons donc définir les parcours suivants : LexBFS*, LexDFS*, MCS*, MNS*.

Premières Propriétés

- GEN = GEN *
- LexBFS* n'est plus un parcours un largeur et LexDFS* n'est plus un parcours en profondeur

Caractérisation de MNS *

Propriété (MNS *)

Étant donné un ordre σ sur V, si $a < b < c$ et $ac \in E$ et $ab \notin E$, alors il existe un sommet $d < b$ tel que $db \in E$. En outre si $b, c \in$ la même composante connexe de $G - \{a]\}$ alors $dc \notin E$.

Caractérisations

Un ordre σ sur V provient d'un parcours MNS* ss'il vérifie la propriété ci-dessus.
On obtient des conditions analogues pour LexBFS*, LexFS*.

D'après le configurations interdites nous retrouvons les inclusions suivantes :

Inclusions

$$
\begin{tikzcd}
 & \text{Gen} & \\
 & \text{MNS*} \arrow[u] & \\
\text{LexBFS*} \arrow[ur] & \text{MCS*} \arrow[u] & \text{LexDFS*} \arrow[ul]
\end{tikzcd}
$$

Théorème Shier 1984

Sur un graphe chordal MNS* =LexBFS* = LexDFS* = MCS* = { l'ensemble de tous les schémas d'élimination simpliciaux }.

Preuve

Soit c le premier sommet non simplicial à gauche. Il existe donc $a < b \in N(c)$ avec $ab \notin E$. b et c appartiennent à la comp. connexe de $G - a]$ et donc nous pouvons utiliser le même raisonnement que celui utilisé pour MNS.
Ainsi tout ordre MNS* fournit un schéma simplicial.
Nous avons montré G chordal ssi tout parcours MNS* (resp. LexBFS*, LexDFS*, MCS*) fournit un schéma d'élimination simplicial.

lemme

On considère un schéma simplicial σ et supposons $a <_\sigma b$ et qu'il existe une chaîne de a vers b dans $G - a[$. Alors il existe une chaîne $a_0 = a <_\sigma a_1 \cdots <_\sigma a_k = b$ compatible avec σ.

MNS* $\supseteq$ tous les schémas simpliciaux.

Soit σ un schéma simplicial et supposons qu'il existe $a <_\sigma b <_\sigma c$ tels que b et c appartiennent à la même composante connexe de $G - a[$, et $ab \notin E$, $ac \in E$.
D'après le lemme il existe un chaîne de b vers c compatible avec σ, la simplicialité implique $ab \in E$ d'où la contradiction.

La preuve est identique pour LexBFS*, LexDFS*, MCS*.

Intérêts des parcours *

Cela peut permettre aussi de faire des parcours plus sophistiqués : parmi ceux qui sont éligibles dans les composantes connexes, choisir le meilleur selon un certain critère.
Par exemple : parmi les maximaux pour l'inclusion, choisir celui de cardinal minimum, sorte de Max-Min.
C'est avec une idée de ce genre que Levesque et Maffray ont proposé en 2007 un algorithme glouton pour le calcul de la coloration des graphes ...
Mais nous risquons de perdre la linéarité de l'implémentation.
Que faire de LexM de Tarjan dans ce scénario ?