Complétions d'intervalles minimales et largeur linéaire (pathwidth)

Ioan Todinca

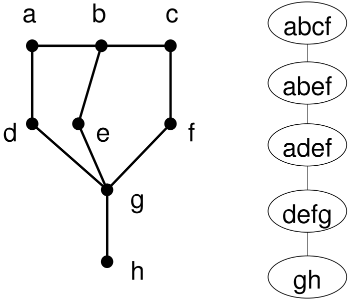
LIFO - Université d'Orléans
LRI - Université Paris XI

Décompositions linéaires (path decompositions)

[Robertson, Seymour 83]

$G \to (P, \mathcal{X})$.

- Chaque sommet et chaque arête de G sont couverts par un sac.
- Pour tout sommet x de G, les sacs contenant x induisent une sous-chaîne *connexe* de P.

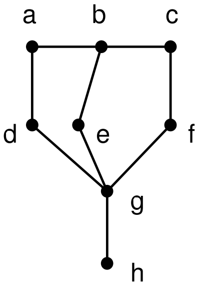


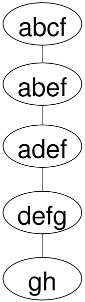
Largeur linéaire

[Robertson, Seymour 83]

$G \to (P, \mathcal{X})$.

- Largeur de la décomposition : taille du plus grand sac, moins un.
- Largeur linéaire pw(G) : largeur minimum parmi toutes les décompositions linéaires de G.





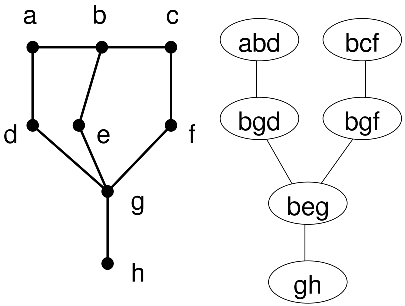
Largeur linéaire de G : écart entre G et la classe des chaînes.

Décomposition arborescente et largeur arborescents

[Robertson, Seymour 84/86]

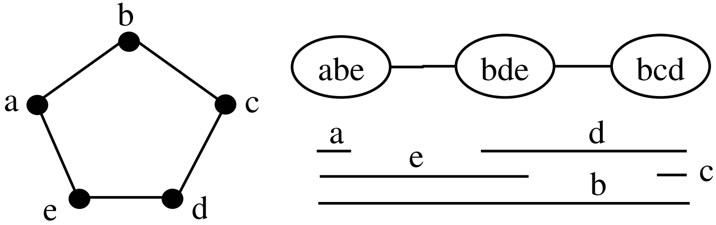
$G \to (T, \mathcal{X})$

- Arbre de décomposition.
- Largeur arborescente $\mathrm{tw}(G)$: largeur minimum parmi toutes les décompositions arborescentes de G.



Pour tout graphe G, $\mathrm{tw}(G) \leq \mathrm{pw}(G)$.

Largeur linéaire et complétions d'intervalles

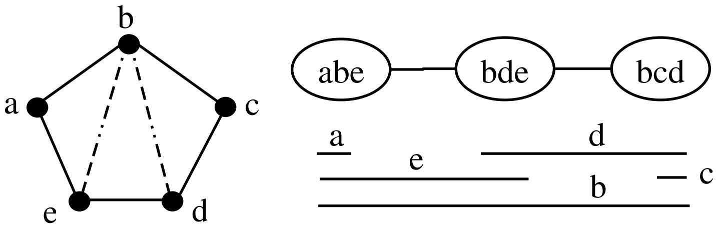


Complétion d'intervalles de $G = (V, E)$: graphe d'intervalles $H = (V, F)$ tel que $E \subseteq F$.

$$\mathrm{pw}(G) = \min\{\omega(H) \mid H \text{ complétion d'intervalles de } G\} - 1$$

Notation $\omega(H)$: taille de la clique de cardinal maximum de H.

Largeur linéaire et complétions d'intervalles



Complétion d'intervalles de $G = (V, E)$: graphe d'intervalles $H = (V, F)$ tel que $E \subseteq F$.

$$\mathrm{pw}(G) = \min\{\omega(H) \mid H \text{ complétion d'intervalles de } G\} - 1$$

Notation $\omega(H)$: taille de la clique de cardinal maximum de H.

Complétions de graphes et paramètres associés

$G = (V, E) \rightarrow H = (V, F)$

	$\min \omega(H) - 1$	$\min \lvert F \setminus E \rvert$
H triangulé	largeur arborescente	remplissage min.
H d'intervalles	largeur linéaire	profil
H d'intervalles propres	largeur de bande	MPIC

Complétions de graphes et paramètres associés

$G = (V, E) \rightarrow H = (V, F)$

	$\min \omega(H) - 1$	$\min \lvert F \setminus E \rvert$
H triangulé	largeur arborescente	remplissage min.
H d'intervalles	largeur linéaire	profil
H d'intervalles propres	largeur de bande	MPIC

Tous ces problèmes sont NP-difficiles.

Complétions de graphes et paramètres associés

$G = (V, E) \rightarrow H = (V, F)$

	$\min \omega(H) - 1$	$\min \lvert F \setminus E \rvert$
H triangulé	largeur arborescente	remplissage min.
H d'intervalles	largeur linéaire	profil
H d'intervalles propres	largeur de bande	MPIC

Tous ces problèmes sont NP-difficiles.

Les solutions optimales se trouvent parmi les complétions minimales.

Calcul d'une complétion d'intervalles minimale

Définition

Complétion d'intervalles minimale : aucun sous-graphe strict de H n'est une complétion d'intervalles de G.

1. Algorithme incrémental [Heggernes, Suchan, T, Villanger 05]
2. Caractérisation des graphes d'intervalles par un ordre spécial sur les sommets [Suchan, T. 06]
   - $\mathcal{O}(nm)$;
   - approche similaire pour les complétions d'intervalles propres minimales [Suchan, Rapaport, T. 06].
3. Extraction d'une complétion minimale à partir d'une complétion d'intervalles quelconque

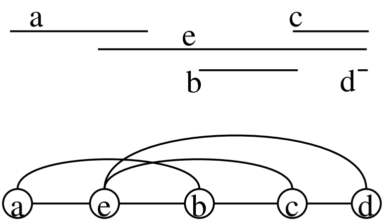
2. Calcul d'une complétion d'intervalles minimale par ordonnancement des sommets

Graphes d'intervalles et ordonnancement des sommets

Théorème ([Olariu 91])

Un graphe $H = (V, F)$ est d'intervalles si et seulement si il existe un ordonnancement d'intervalles $\sigma = (v_1, \ldots, v_n)$ de ses sommets :

$$\text{pour tout } i < j \leq k, \text{ si } v_i v_k \in F \text{ alors } v_i v_j \in F$$



- Modèle d'intervalles → ordonnancement d'intervalles : ordonner les sommets par la borne gauche des intervalles respectifs.

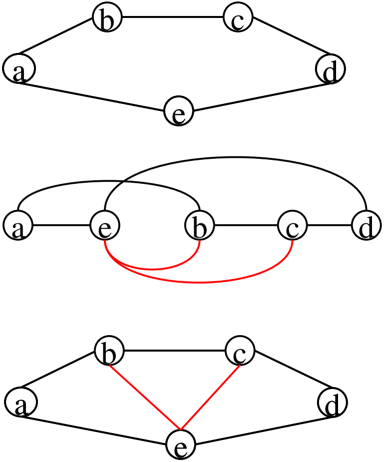
Forcer un ordonnancement d'intervalles

Lemme

Soit $\sigma = (v_1, \ldots, v_n)$ *un ordonnancement des sommets du graphe quelconque* $G = (V, E)$. *Définissons le sur-graphe* $G(\sigma) = (V, F')$ *de* G *par :*

$$\text{pour tout } i < j \leq k, \text{ si } v_i v_k \in E \text{ alors } v_i v_j \in F'$$

Le graphe $G(\sigma)$ *est d'intervalles.*



Bons ordonnancements

Remarque : pour toute complétion d'intervalles minimale H de G, il existe un ordonnancement des sommets tel que $H = G(\sigma)$.

- Etant donné $G = (V, E)$, trouver un bon ordonnancement, i.e. un ordonnancement σ tel que $G(\sigma)$ soit une complétion d'intervalles minimale de G.
- Bon préfixe $\rho = (v_1, \ldots, v_k)$: un préfixe qui peut être étendu en un bon ordonnancement.

Choisir le premier sommet

L'outil magique : **LexBFS** – Lexicographic Breadth-First Search [Rose, Tarjan, Leuker 76].

- Le dernier sommet rencontré par **LexBFS** est une "extrémité" du graphe en entrée.
- Cf. [Berry, Bordat 98,00] pour des propriétés spécifiques de ce dernier sommet.

Théorème

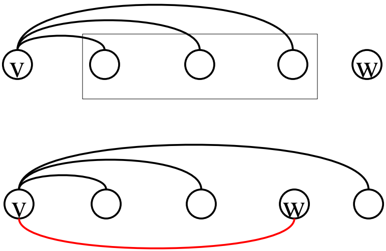
Le dernier sommet rencontré par **LexBFS** *est un bon préfixe.*

Choisir le prochain sommet – un exemple facile

Comment choisir le deuxième sommet ?

- S'assurer que $N(v_1)$ vient juste après v_1 dans l'ordonnancement σ.
- Tout autre choix σ' engendrerait une arête qui n'apparaît pas dans $G(\sigma)$.

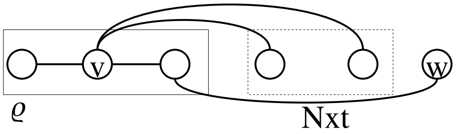
Le second sommet sera dans $N(v_1)$.

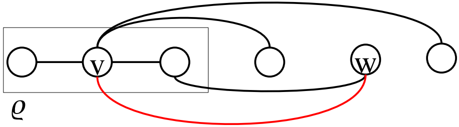


Choisir le prochain sommet – cas général

Préfixe courant ρ ; soit $R = V \setminus \rho$.

- Choisir $v_i \in \rho$ tel que $Nxt = N(v_i) \cap R$ soit non-vide et minimal par inclusion parmi toutes les possibilités.
- S'assurer que Nxt vient juste après ρ dans l'ordonnancement σ.
- Tout autre choix σ' produit une arête qui n'existe pas dans $G(\sigma)$.



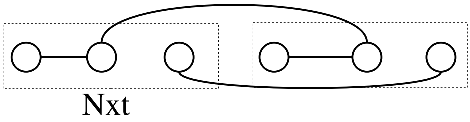


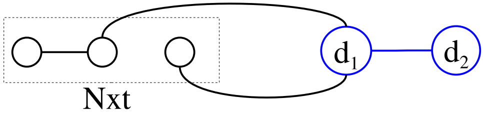
Le prochain sommet sera dans Nxt.

Choisir dans *Nxt* – **LexBFS** à nouveau

- Presque la même tâche que de choisir un premier sommet pour $G[Nxt]$.
- Tenir compte des sommets de Nxt qui ont des voisins après Nxt.

Lancer **LexBFS** sur le graphe auxiliaire $G^+[Nxt]$, à partir du sommet d_2. Le dernier sommet rencontré par **LexBFS** est ajouté au préfixe courant ρ.





L'algorithme par ordonnancement des sommets

Théorème

Il existe un algorithme de complexité $\mathcal{O}(nm)$ qui calcule une complétion d'intervalles minimale d'un graphe quelconque.

- Algorithme simple, bassé sur l'ordonnancement des sommets.
- La sortie est un ordre BFS particulier.
- Utilise n fois **LexBFS**.

3. Extraction d'une complétion d'intervalles minimale

Le problème de l'extraction

- **Entrée :** $G = (V, E)$ quelconque et $H = (V, F)$ une complétion d'intervalles, pas nécessairement minimale, de G.
- **Sortie :** une complétion d'intervalles *minimale* H' telle que $G \subseteq H' \subset H$.

Difficulté : les complétions quasi-minimales

Lorsque pour toute arête $e \in F \setminus E$, le graphe $H - e$ n'est pas d'intervalles. On dit que H est une complétion *quasi-minimale* ; elle n'est pas forcément minimale...

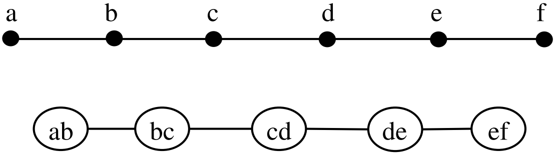
Complétions quasi-minimales et pliage

- H une complétion quasi-minimale non minimale de G.
- Il existe une complétion minimale $H' \subset H$.
- Quel lien entre H et H' ?

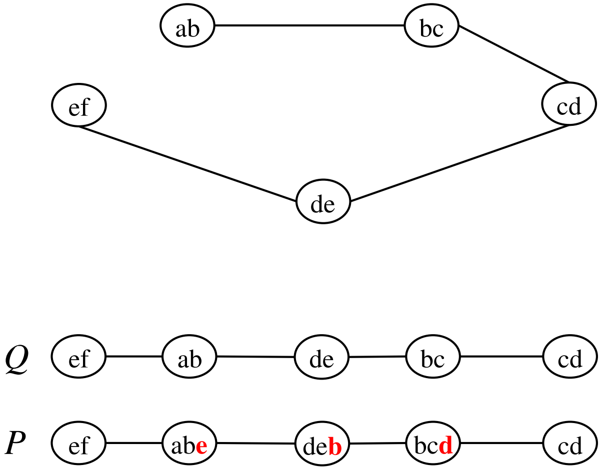
Théorème ([Heggernes, Suchan, T., Villanger 07])

Soit H une complétion quasi-minimale de G et H' une complétion telle que $G \subseteq H' \subset H$. Il existe un pliage $(G, \mathcal{P}, \mathcal{Q})$ de H' tel que $H = \mathtt{Pliage}(H', \mathcal{Q})$.

Pliage



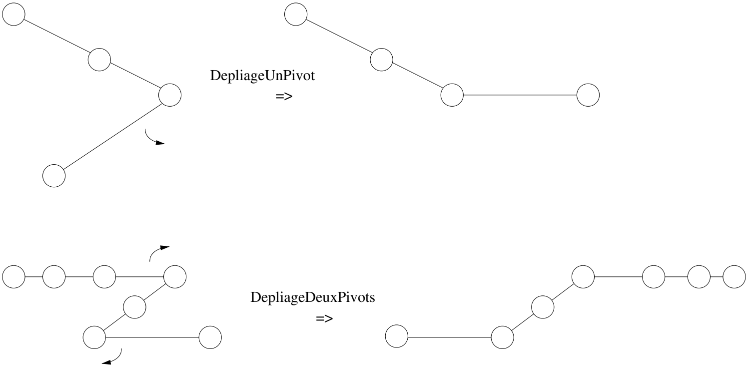
Pliage



1-pliage et 2-pliage

Théorème ([Heggernes, Suchan, T., Villanger 07])

Si H est une complétion quasi-minimale mais non minimale de G, il existe une complétion $H'' \subset H$ de G telle que H corresponde à un 1 ou 2-pliage de H''.



Extraction d'une complétion d'intervalles minimale : l'algorithme

répéter

- élaguer des arêtes de H jusqu'à l'obtention d'une complétion quasi-minimale
- appliquer la procédure de dépliage sur H

jusqu'à ce que H reste inchangé
retourner H

Théorème

Il existe un algorithme polynomial qui, à partir d'une complétion d'intervalles H de G, extrait une complétion d'intervalles minimale de G.

Calcul de la largeur linéaire ?

Très peu de choses connues !

- Algorithmes linéaires pour les arbres, algorithmes polynomiaux pour les graphes de largeur arborescente bornée.
- Algorithmes polynomiaux pour certaines classes où largeur linéaire = largeur arborescente.
- Approximation à partir de la largeur arborescente, en utilisant $\mathrm{pw}(G) \leq c \cdot \mathrm{tw}(G) \log n$.

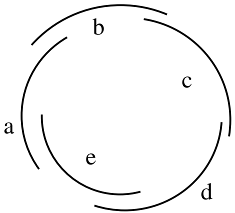
Calcul de la largeur linéaire ?

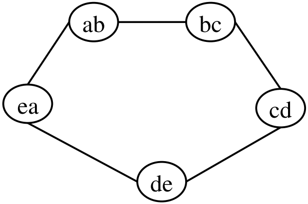
Très peu de choses connues !

- Algorithmes linéaires pour les arbres, algorithmes polynomiaux pour les graphes de largeur arborescente bornée.
- Algorithmes polynomiaux pour certaines classes où largeur linéaire = largeur arborescente.
- Approximation à partir de la largeur arborescente, en utilisant $\mathrm{pw}(G) \leq c \cdot \mathrm{tw}(G) \log n$.

Algorithme polynomial pour les graphes d'intervalles circulaires.

Graphes d'intervalles circulaires



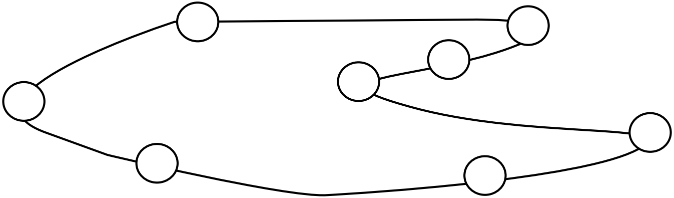


Cycle de cliques : permet de définir la même notion de pliage.

Largeur linéaire des graphes d'intervalles circulaires

Théorème ([Suchan, T. rr06])

Il existe un pliage $(G, \mathcal{C}, \mathcal{Q})$ *ayant exactement deux pivots, tel que* $H = \mathtt{Pliage}(G, \mathcal{Q})$ *satisfait* $\mathrm{pw}(H) = \mathrm{pw}(G)$.

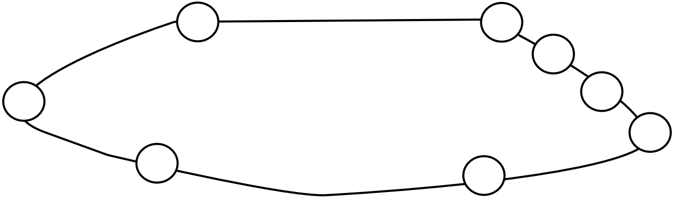


Calcul de la largeur linéaire en temps quadratique.

Largeur linéaire des graphes d'intervalles circulaires

Théorème ([Suchan, T. rr06])

Il existe un pliage $(G, \mathcal{C}, \mathcal{Q})$ ayant exactement deux pivots, tel que $H = \mathtt{Pliage}(G, \mathcal{Q})$ satisfait $\text{pw}(H) = \text{pw}(G)$.



Calcul de la largeur linéaire en temps quadratique.

Pour résumer

- Complétions d'intervalles minimales et largeur linéaire.
- Approche par ordonnancement des sommets.
- Pliage
  - Extraction d'une complétion d'intervalles minimale.
  - Calcul de la largeur linéaire pour les graphes d'intervalles circulaires.

Autour des complétions d'intervalles minimales

Travaux connexes : complétions minimales en graphes split [Heggernes, Mancini 06] et en graphes de comparabilité [Heggernes, Mancini, Papadopoulos 06].

Questions ouvertes :

- Caractérisation simple des complétions d'intervalles minimales (comme les ordres d'élimination parfaits pour les triangulations minimales) ?
- Evaluation des heuristiques ?
- Calcul, approximation de la largeur linéaire ?