

Algorithmes approchés

ESP (Ecole Supérieure Polytechnique) de Dakar

Optimisation en informatique

Alain Faye

7 – Algorithmes approchés avec garantie

Algorithmes approchés avec garantie

- Dans le cadre de l'optimisation discrète
- Algorithmes spécifiques à chaque problème
 - Max 3 SAT, Min de sommets pour couvrir les arêtes d'un graphe, sac-à-dos
- Algorithmes polynomiaux donc rapides
- Performance donnée a priori et garantie quelque soit l'instance du problème

Algorithmes approchés - définition

z^* valeur de la solution optimale du problème

z valeur de la solution donnée par un algorithme A

- Problème de type minimiser

A est un algorithme ρ -approché ssi

$$z \leq \rho z^* \text{ avec } \rho > 1$$

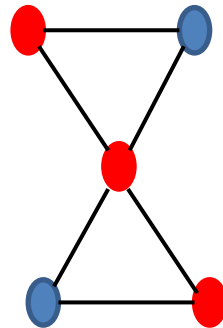
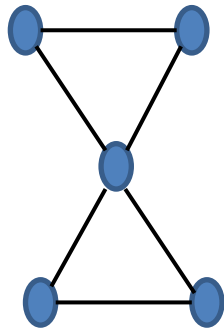
- Problème de type maximiser

A est un algorithme ρ -approché ssi

$$z \geq \rho z^* \text{ avec } 0 < \rho < 1$$

Exemple – min vertex cover

- Un graphe $G=(X,E)$ non orienté
- Trouver un ensemble de sommets de taille minimum qui couvre toutes les arêtes de G

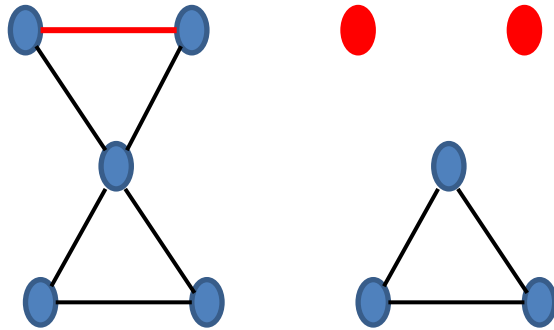


3 sommets pour couvrir les arêtes

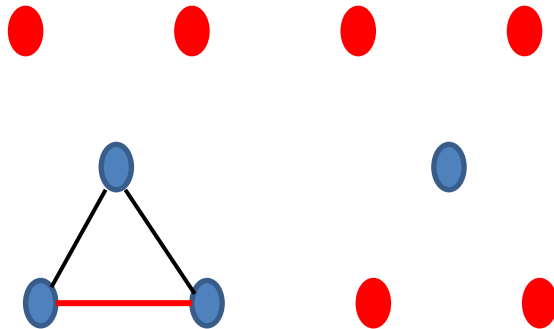
Exemple – min vertex cover

- Algorithmme
 - $F := E, Y := \emptyset$
 - Tant que F est non vide
 - choisir une arête dans F
 - Ajouter ses deux extrémités dans Y
 - supprimer de F toutes les arêtes incidentes à ces deux sommets
 - Fin tant que

Exemple – déroulement de l'algorithme



Itération 1 je choisis l'arête rouge
on met dans Y les 2 sommets rouges
on enlève les arêtes incidentes à ces deux sommets



Itération 2 je choisis l'arête rouge
on met dans Y deux nouveaux sommets rouges
on enlève les arêtes incidentes à ces deux sommets

Il n'y a plus d'arêtes
On a trouvé 4 sommets couvrants

Min vertex cover - Performance de l'algorithme

- Soit k le nombre d'arêtes sélectionnées
- Le nombre de sommets donnés par l'algo est $z=2k$
- Maintenant les k arêtes forment un couplage (elles ne sont pas incidentes les unes aux autres), donc pour les couvrir il faudra au minimum k sommets (un par arête)
- On a donc $k \leq z^* \Rightarrow 2k \leq 2z^* \Rightarrow z \leq 2z^*$
- L'algorithme est 2-approché

Exemple : TSP - voyageur commerce

- TSP symétrique

poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20

Algorithme

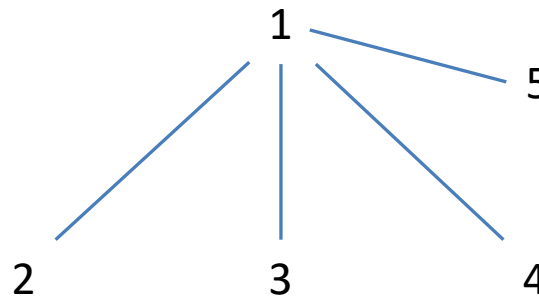
- 1- Calculer T un MST-arbre couvrant poids min
- 2- Doubler les arêtes de T : on obtient graphe eulérien (d° pairs)
- 3- Calculer C_{Euler} un cycle eulérien (qui passe par toutes les arêtes)
- 4- « Shunter » les sommets traversés plus d'une fois dans C_{Euler} .
- 5- On obtient un cycle C passant une fois et une seule par chaque sommet.

Garantie pour les graphes métriques

- Dans le cas de poids vérifiant l'inégalité triangulaire, cet algorithme est **2-approché** i.e. fournit une solution de poids ne dépassant jamais le double de la solution optimale

1- Calculer T un MST-arbre couvrant poids min

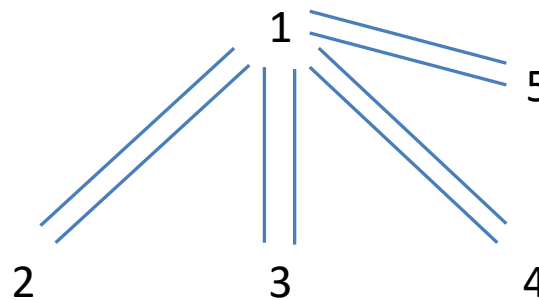
poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



MST poids = 40

2- Doubler les arêtes de T : on obtient graphe eulérien (d° pairs)

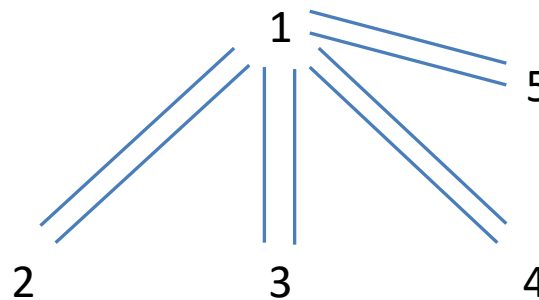
poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



Les d° sont pairs

3- Calculer C_{Euler} un cycle eulérien (qui passe par toutes les arêtes)

poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



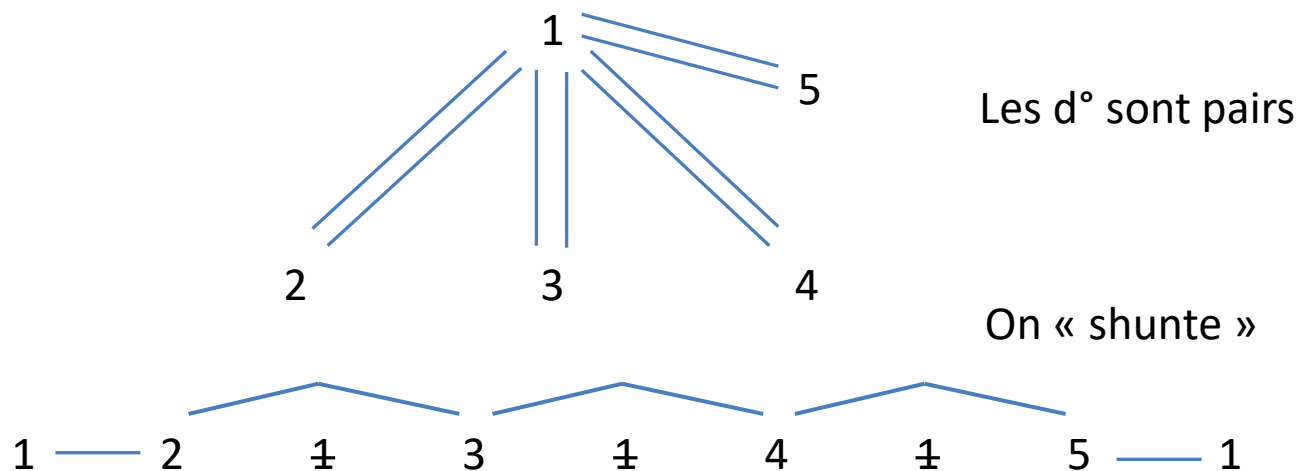
Les d° sont pairs

Cycle eulérien
poids = 80

1 — 2 — 1 — 3 — 1 — 4 — 1 — 5 — 1

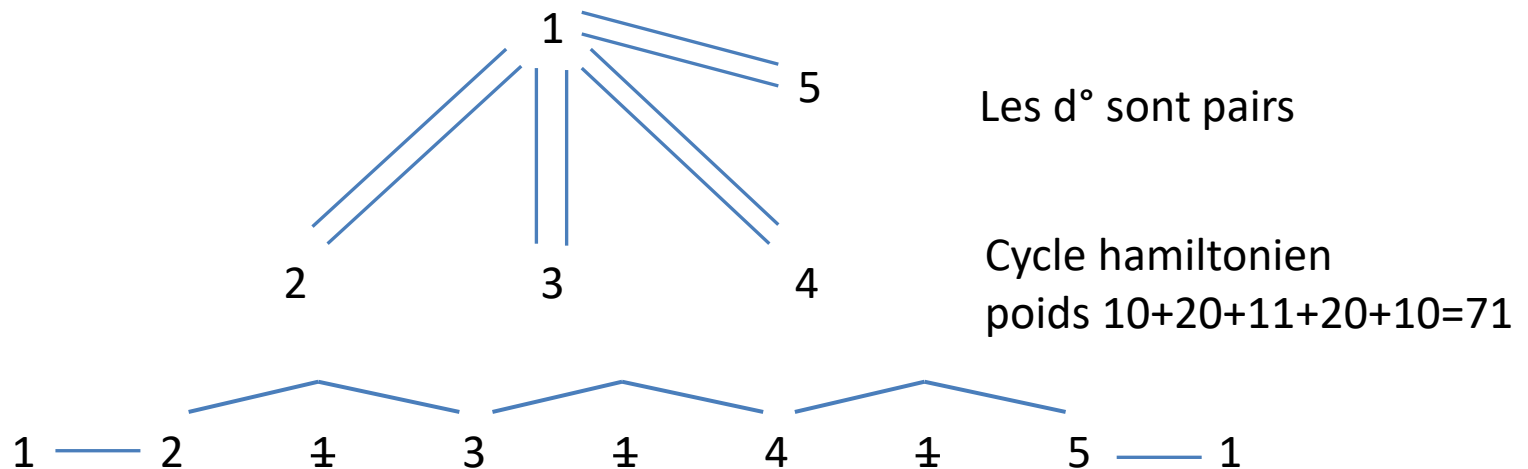
4- « Shunter » les sommets traversés plus d'une fois dans C_{Euler}

poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



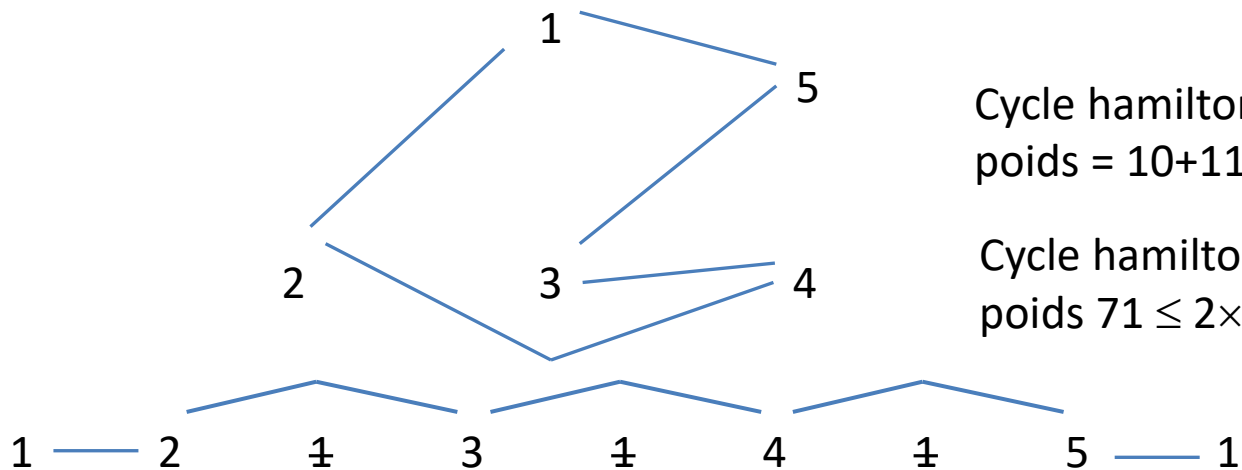
5- On obtient un cycle C passant une fois et une seule par chaque sommet

poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



La garantie est respectée

poids	1	2	3	4	5
1	x	10	10	10	10
2		x	20	11	11
3			x	11	11
4				x	20



Cycle hamiltonien optimal
poids = $10+11+11+11+10=53$

Cycle hamiltonien
poids $71 \leq 2 \times 53$

Preuve de la garantie

- Algorithme 2-approché pour graphe métrique: donne un cycle hamiltonien de poids z qui ne dépasse jamais 2 fois le poids de la solution optimale

Démonstration:

- poids MST $\leq$ poids cycle optimum
car si on retire une arête au cycle on obtient un arbre
- Donc $2 \times (\text{poids MST}) \leq 2 \times (\text{poids cycle optimum})$
- Par l'inégalité triangulaire, quand on « shunte » on n'augmente pas le poids donc $z \leq 2 \times (\text{poids MST})$
- Finalement $z \leq 2 \times (\text{poids MST}) \leq 2 \times (\text{poids cycle optimum})$

Algorithmes approchés - Conclusion

- Algorithmes spécifiques
- Garantie a priori sur la solution fournie
- Garantie dans le pire des cas, on peut espérer qu'en moyenne l'algorithme donne de bien meilleurs résultats que la garantie
- Tous les problèmes ne sont pas ρ -approximables (ρ fixé), par exemple max stable
- Il existe des algorithmes avec garantie ρ dépendant de l'instance, par exemple de la taille