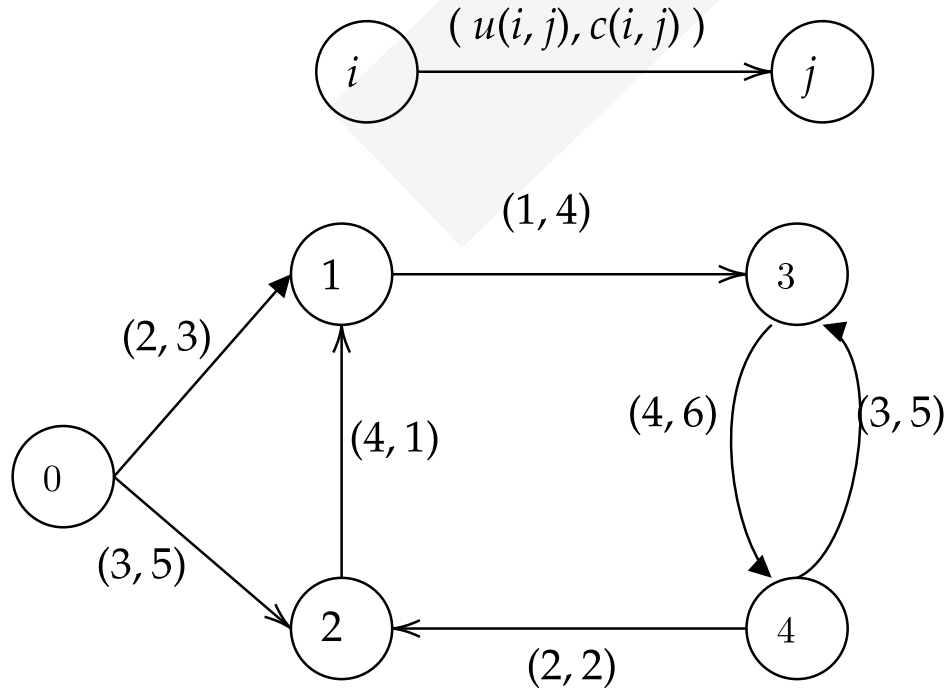


Identification de sous-structures optimales pour les graphes de flot

TIPE 2022
JOACHIM Julien
14160


Graphe de flot



Exemple de Graphe de flot a 5 sommets

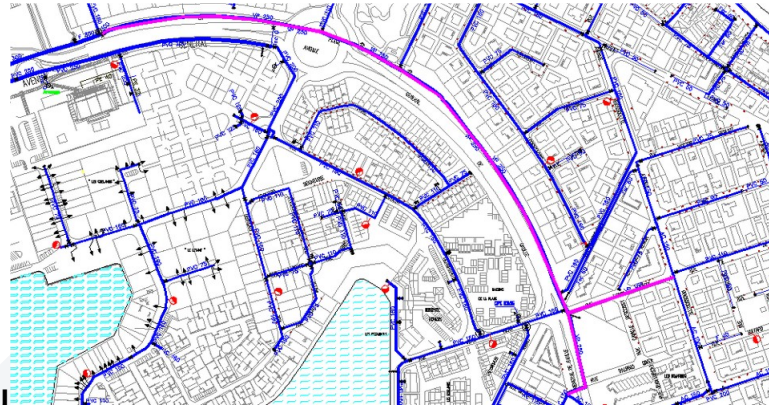
Motivations




Feuille de  Geranium Rouge



 Angiographie

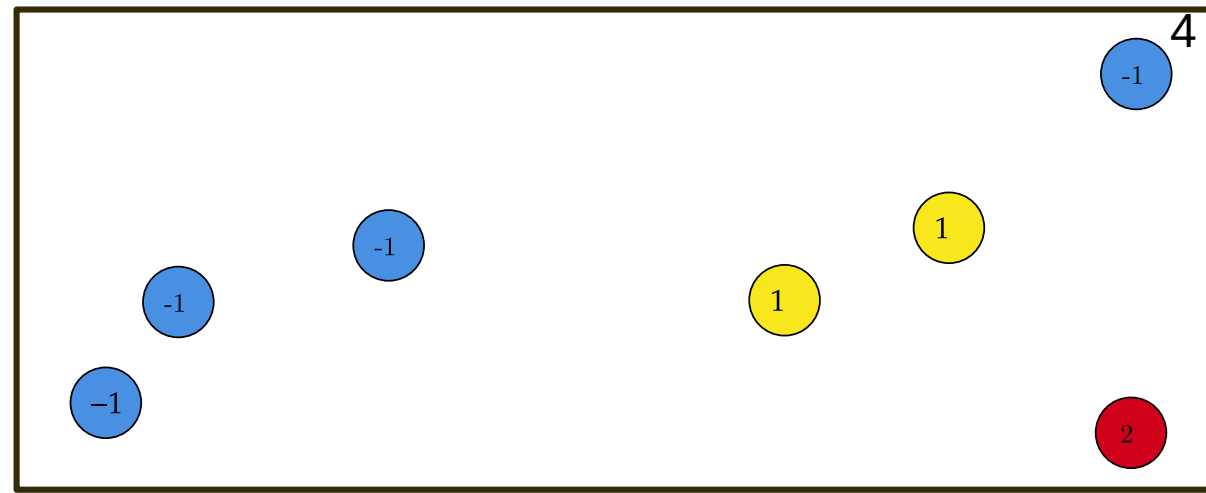


Réseau urbain
de distribution d'eau 

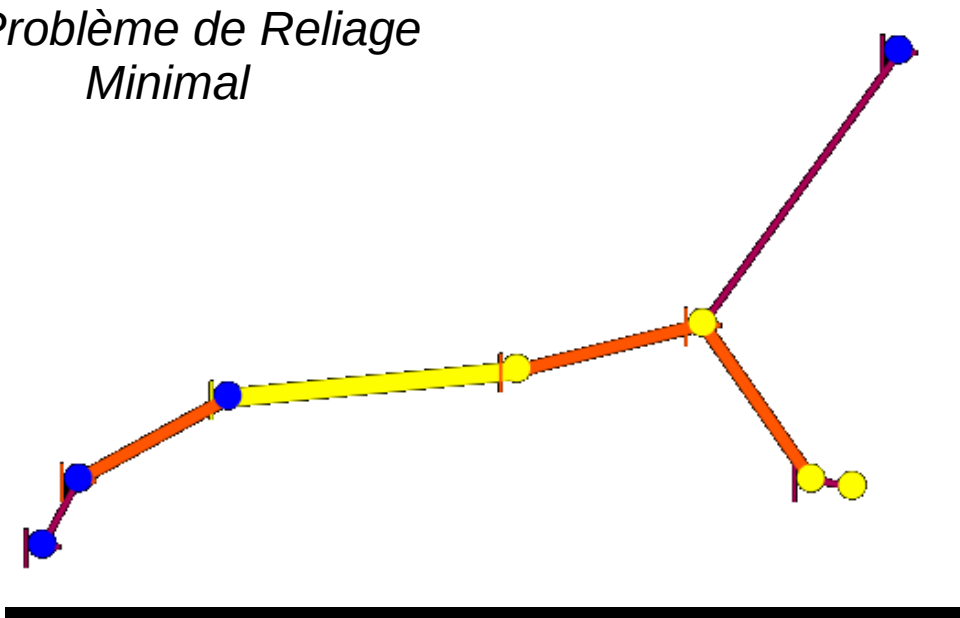


Arbre mort

Problématique



→ *Problème de Reliage Minimal*



Problématique

“

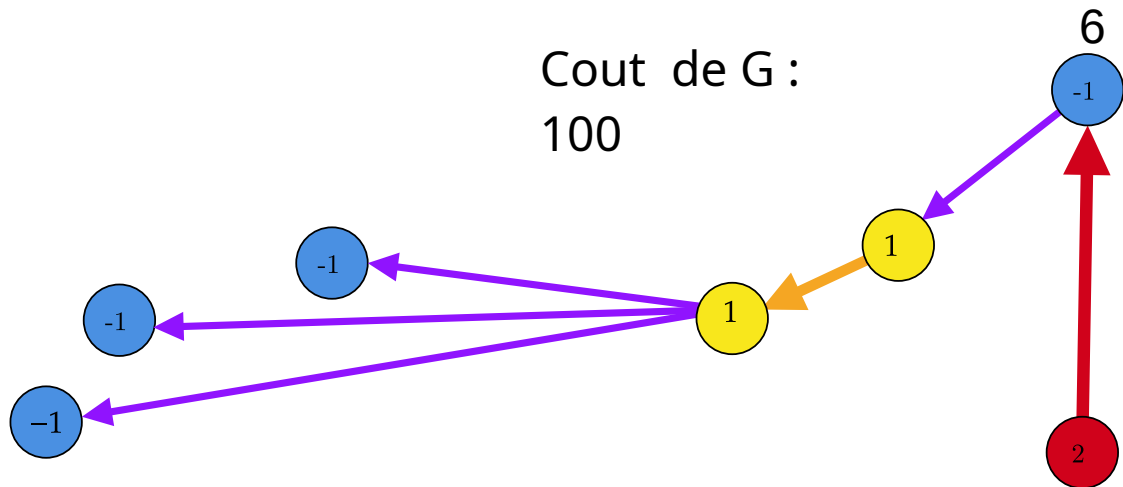
Les solutions au problème de reliage
minimal possèdent-elles des
sous-structures optimales?

”

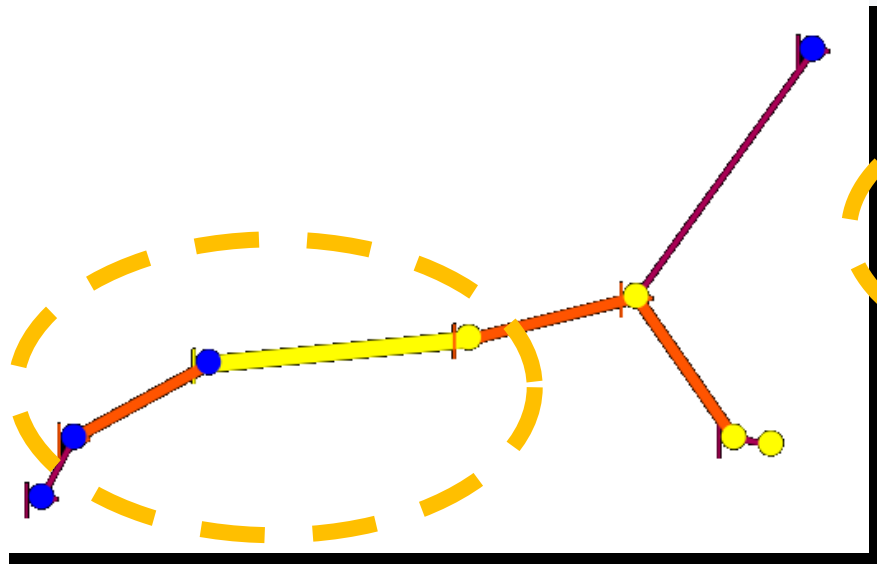
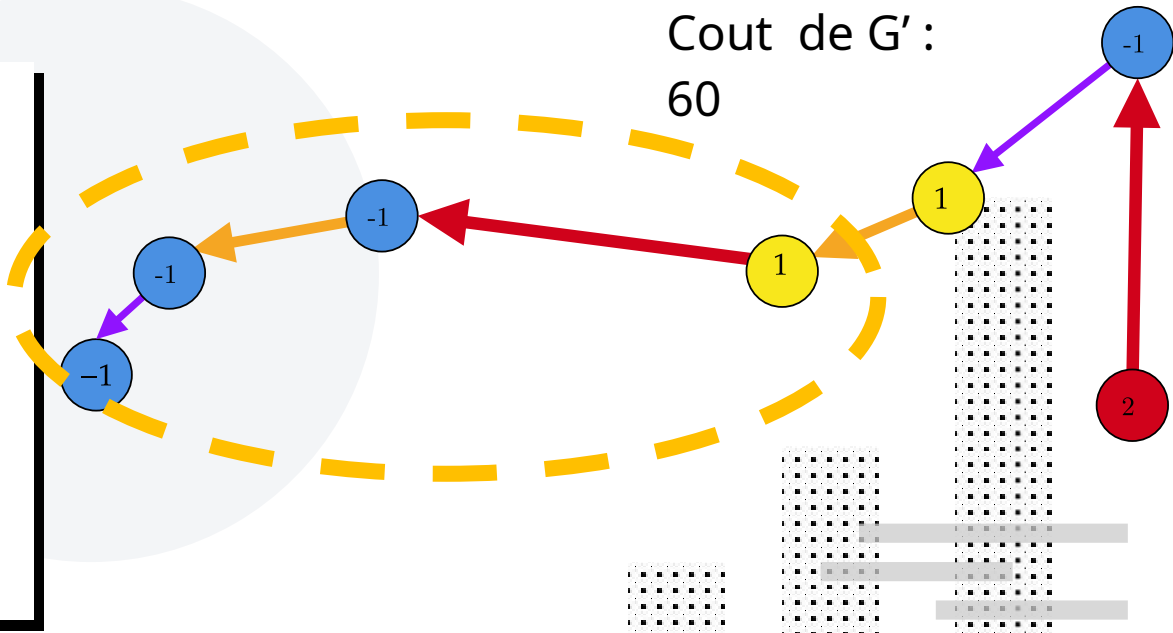
Problématique

Sous-structures optimales

Cout de G :
100



Cout de G' :
60



Sommaire

1 Expérimentations

Méthode

Structure de graphe efficace

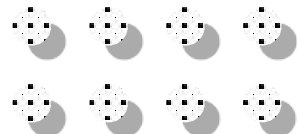
Exemples de graphes optimaux

2 Description de graphes de flots optimaux

Premières Propriétés

Passage n à $n-1$

Passage n à $n+1$

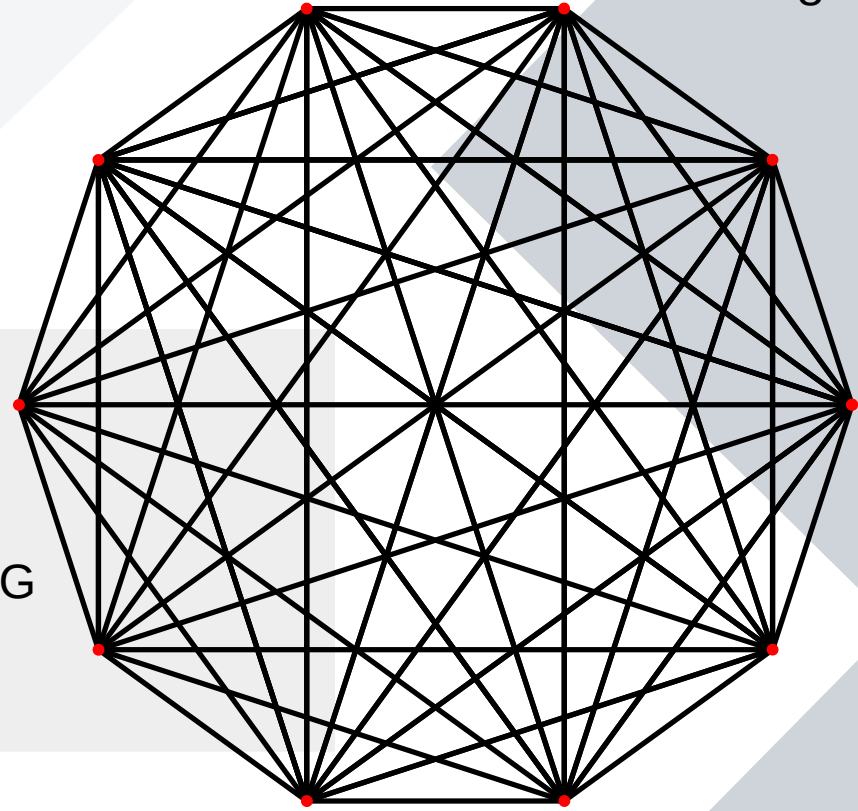


I - Expérimentations - Resoude le probleme de R.M.

Algorithme de résolution

Entree : Ensemble de sommets V

- 1 - Construire graphe complet G
- 2 - Résoudre problème min-cost flow sur G
- 3 - Emonder



Exemple de graphe complet

2 : Cycle Canceling | Capacity/Cost Scaling

Implémentation de la structure de graphe de flot efficace

	Matrice d'adjacence	Liste d'Adjacence	Foward and reverse star	(Dans ce rapport)
En memoire	$\Theta(n^2)$	$\Theta(n + m)$	$\Theta(n + m)$	$\Theta(n + m)$
Liste des predecesseurs et des successeurs	$\Theta(n)$	$\Theta(1)$	$\Theta(1)$	$\Theta(1)$ (sous forme d'iterateur)
Existence d'un arc	$\Theta(1)$	$O(a_{max})$ ou a_{max} est l'arité maximale des sommets $O(n)$ pour une borne polynomiale	$\Theta(1)$	$\Theta(1)$
Ajout d'un arc	$\Theta(1)$	$O(a_{max})$ ou $O(n)$	—	$\Theta(1)$



Comparaison complexités temps et espace des structures de données classiques sur les graphes



Expérimentations

Structures auxilliaires

`dynamic_array`

`linked_array`

Structure classique de
tableau redimensionnable

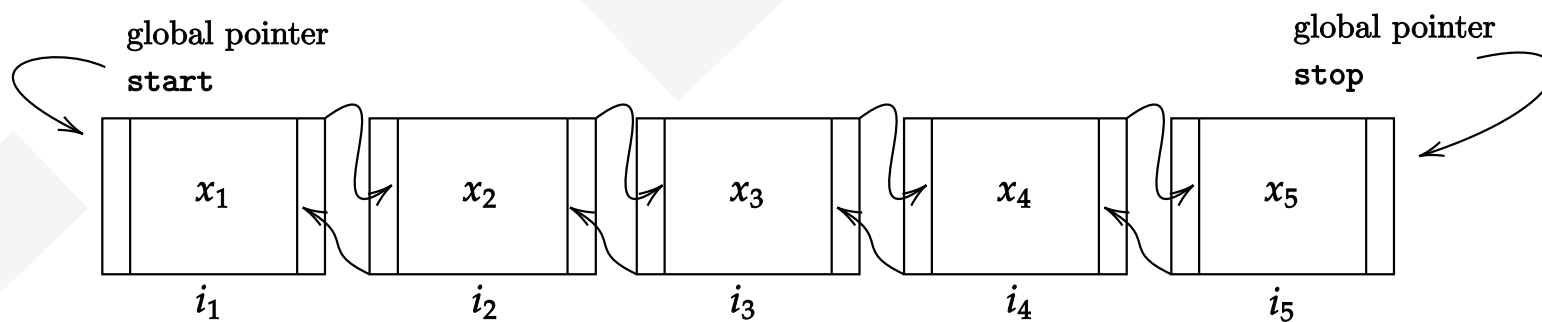
Liste chaînée "indiquée"

$\Theta(n)$ espace

$\Theta(n)$ space



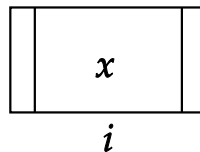
Structure `Linked_array`



Legende

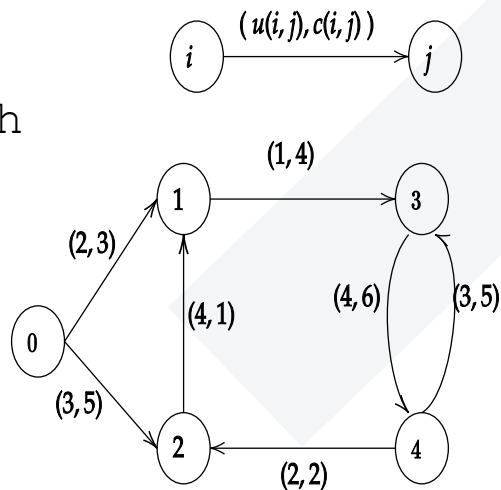


Pointeur vers la cellule
suivante (ou precedente)
dans la chaine



Une cellule du tableau chaine
avec son contenu x et son indice
fixe i

Structure flow_graph



A) Exemple de graphe

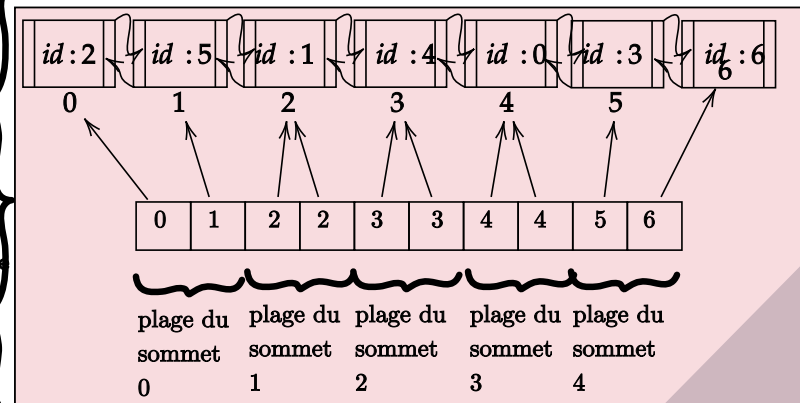
id de l'arc (i, j)	Queue i	Tete j	Capacite $u(i, j)$	Coût $c(i, j)$	Autres attributs...
id : 0	3	4	4	6	...
id : 1	1	3	1	4	
id : 2	0	2	3	5	
id : 3	4	2	2	2	
id : 4	2	1	4	1	
id : 5	0	1	2	3	
id : 6	4	3	3	5	

B) Tableau des arcs

Schématisation de la structure flow_graph

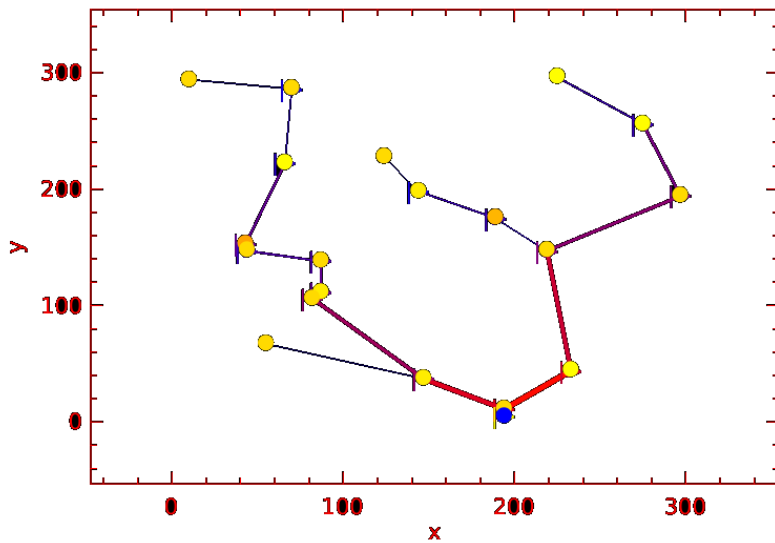
Le tableau chaîné **forward_chain** où les successeurs de chaque sommets sont enchainés consécutivement

Le tableau classique **forward_range** indiquant la plage des successeurs d'un sommet dans **forward_chain**

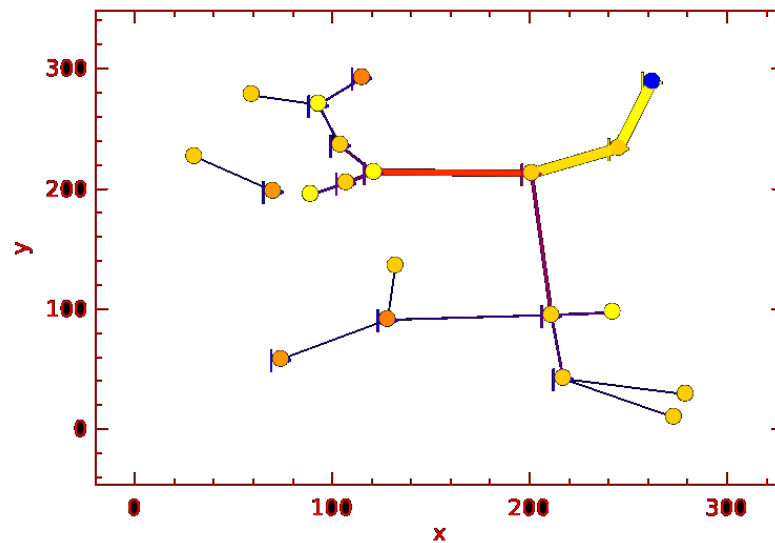


C) forward_chain et forward_range associe

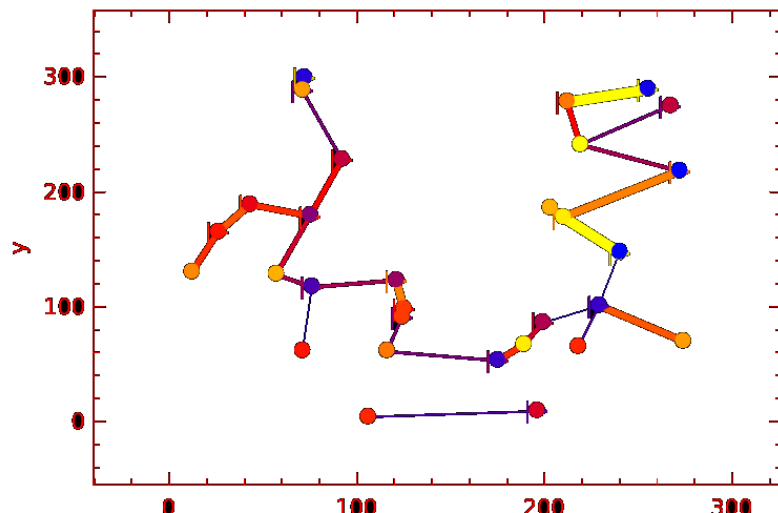
Graph representation



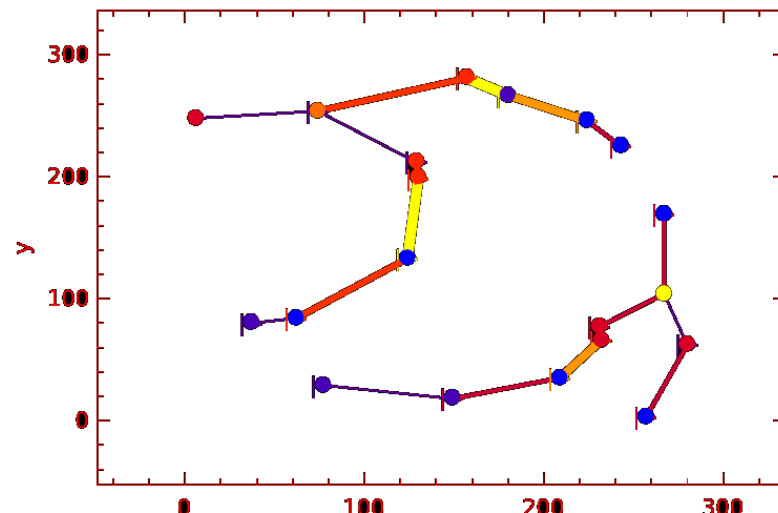
Graph representation



Graph representation



Graph representation

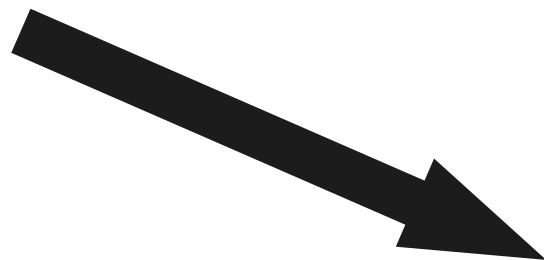




Description des solutions

Premières propriétés

Absence de cycle



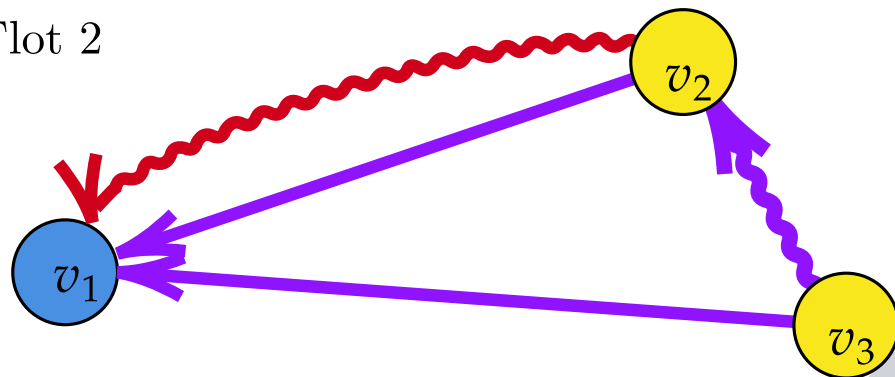
Structure du Squelette

Premières propriétés

Propriétés de la fonction de cout

Pour les couts munis de l'inégalité triangulaire:

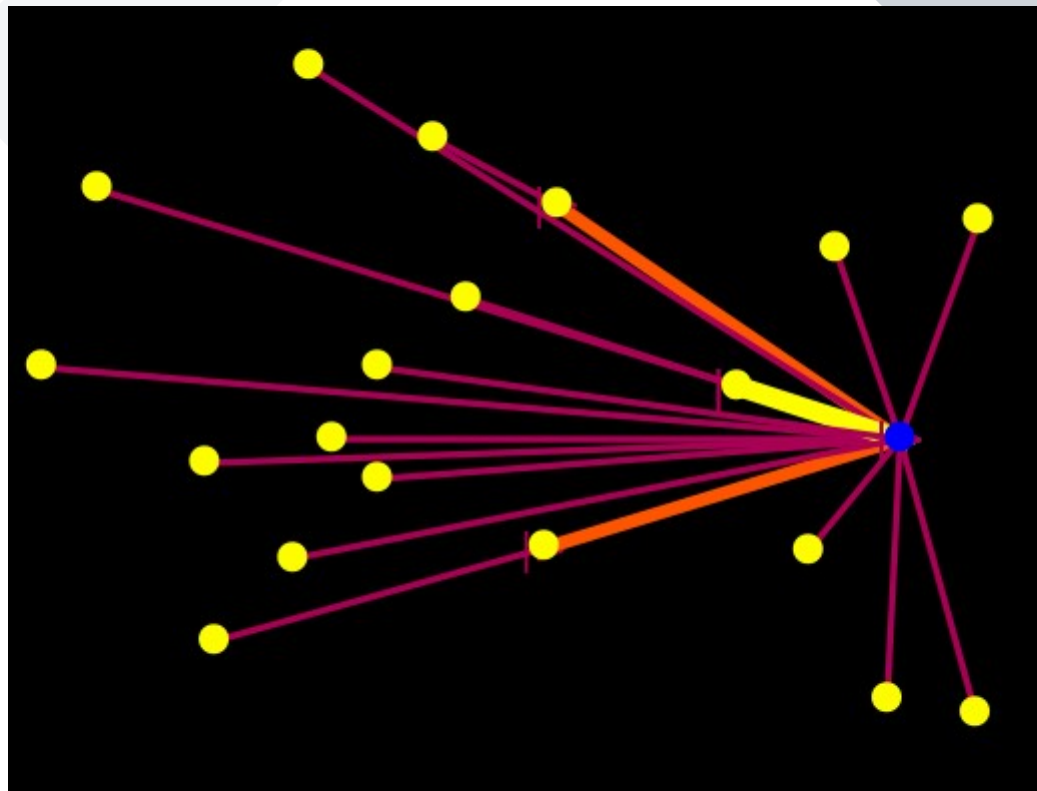
Flot 1
Flot 2



Premières propriétés

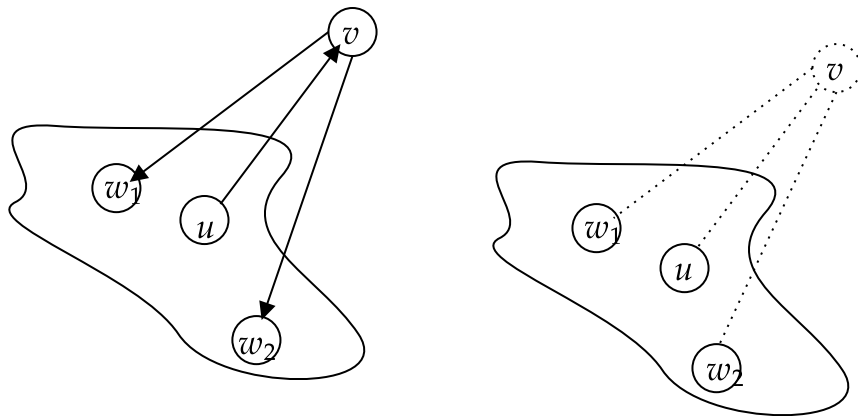
Propriétés de la fonction de cout

Graphe optimal
Obtenu pour la norme N4



Passage de n à $n-1$

Algorithme de descente



Initialisation

Soit $G' = (V', E') := (V', E \setminus \{\text{ensemble des arcs impliquants } v\})$

Pour tout $e \in E'$,

$$f(e) \leftarrow f^\star(e)$$

Execution

Pour tout $u \in \text{Adj}^+[v]$:

$$b(u) \leftarrow b(u) + f^\star(u, v)$$

Pour tout $w \in \text{Adj}^-[v]$:

$$b(w) \leftarrow b(w) - f^\star(v, w)$$

Complexité en $O(m+n)$



FIN