

Connexité dans l'urgence

Cyril Gavaille, Pierre Halftermeyer

LaBRI, Université de Bordeaux 1

28 mai 2012

1 Le problème

- présentation

2 Approche

- Schéma d'étiquetage

3 Schéma planaire de connexité

- Courcelle et al.

4 Contribution

- Généralisation aux graphes plongeable sur une surface de genre eulérien g

5 Conclusion

6 annexes

- Schéma de connectivité avec panne
- Optimisation de l'algorithme de requête
- Requête avec réparation
- Implémentation

1 Le problème

- présentation

2 Approche

- Schéma d'étiquetage

3 Schéma planaire de connexité

- Courcelle et al.

4 Contribution

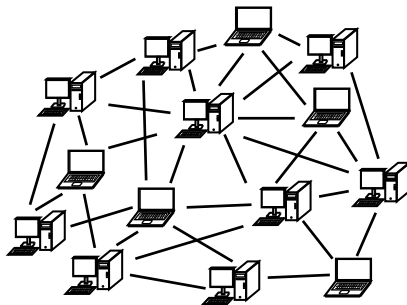
- Généralisation aux graphes plongeable sur une surface de genre eulérien g

5 Conclusion

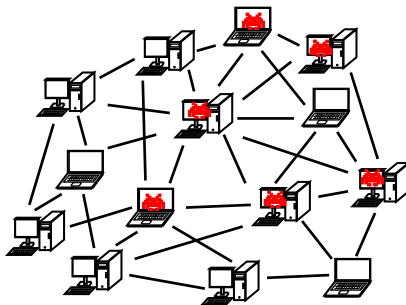
6 annexes

- Schéma de connectivité avec panne
- Optimisation de l'algorithme de requête
- Requête avec réparation
- Implémentation

Un réseau



Un réseau attaqué !



Un autre réseau



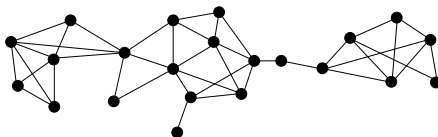
lui aussi attaqué !



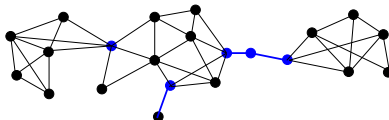
Donc, une question naturelle...



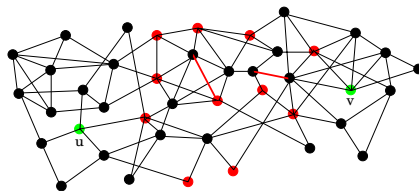
Une seule panne - Un problème simple



Points d'articulation et isthmes



Plusieurs pannes - Un problème plus difficile.



Plusieurs pannes - Un problème plus difficile.

Arêtes en panne seulement : $O(|X| \log^2 n \log \log n)$

M. Pătraşcu, and M. Thorup '07

Plusieurs sommets en panne : $O(|X| \sqrt{n})$

D. Eppstein, Z. Galil, G. F. Italiano, and A. Nissenzweig. '97

Objectifs

On veut déterminer si deux sommets sont toujours connectés, et ce d'autant plus rapidement qu'il est impératif de rétablir la stabilité post-attaque.

Objectifs

Il est envisageable de faire un précalcul polynomial en la taille de G si cela nous permet de répondre en temps polylogarithmique au moment de la crise.

Objectifs

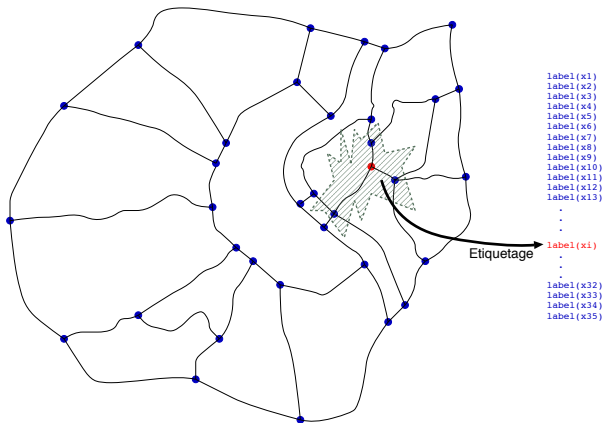
On souhaite répondre en un temps dépendant essentiellement de la taille de la panne et non celle du graphe.

Objectifs

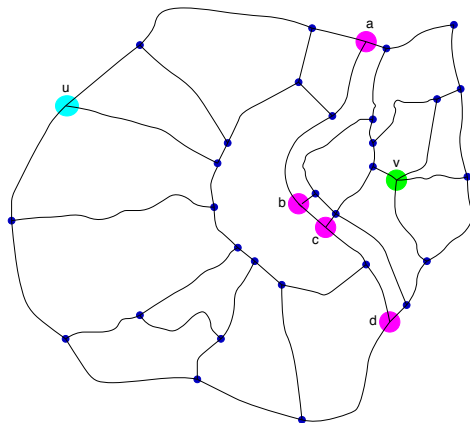
Il convient de planifier l'urgence.

- 1 Le problème
 - présentation
- 2 Approche
 - Schéma d'étiquetage
- 3 Schéma planaire de connexité
 - Courcelle et al.
- 4 Contribution
 - Généralisation aux graphes plongeable sur une surface de genre eulérien g
- 5 Conclusion
- 6 annexes
 - Schéma de connectivité avec panne
 - Optimisation de l'algorithme de requête
 - Requête avec réparation
 - Implémentation

Etiquetage



Requête



```
label(u)  
label(v)  
label(a)  
label(b)  
label(c)  
label(d)
```

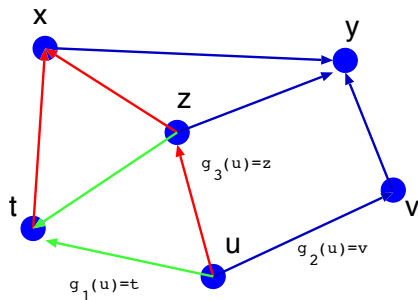
OUI / NON

Exemple d'étiquetage compact - Un schéma d'adjacence

On va présenter un schéma simple répondant, pour les graphes planaires, à la requête :

$$Adj(x, y)$$

Partition d'arête en trois forêts (Haie de Schnyder) et adjacence



Adjacence

En posant $g_i(x) = x$ si $g_i(x)$ n'est pas définie, on a :

$$\text{Adj}(x, y) \iff x \neq y \wedge \bigvee_{1 \leq i \leq 3} (x = g_i(y) \vee y = g_i(x))$$

Si chaque sommet u est identifié de façon unique $Id(u)$, un étiquetage de la forme :

$$\text{label}(x) = Id(x) \quad Id(g_1(x)) \quad Id(g_2(x)) \quad Id(g_3(x))$$

permet de résoudre les requête d'adjacence.

Idée directrice :

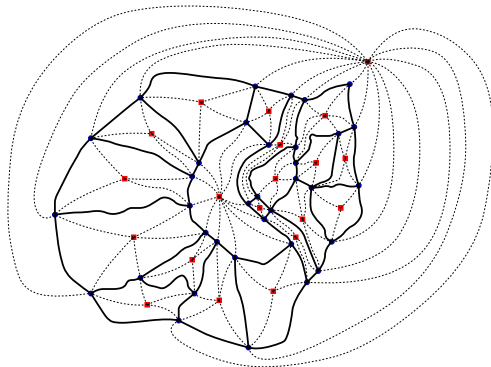
On a une formule sans quantification, combinaison booléenne de formules atomiques de formes $x = y$, $f(x) = y$, $f(x) = g(y)$ sur un nombre fini de fonctions unaires et de variables

Les formules de la forme $x = f(g(y))$ sont **proscrites**.

Les étiquettes de x et y contiennent donc toute l'information nécessaire !

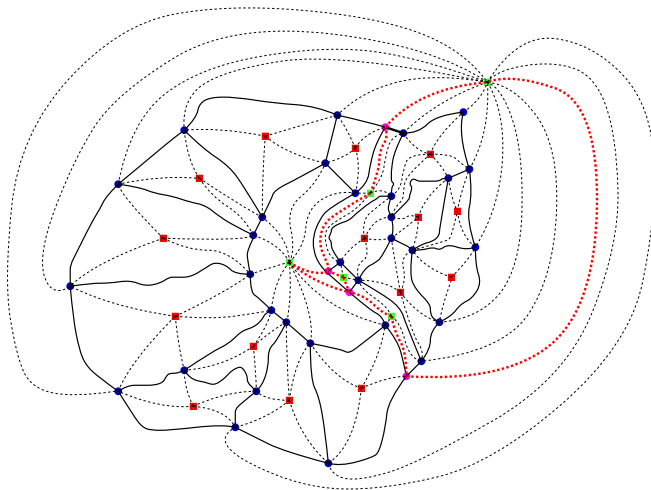
- 1 Le problème
 - présentation
- 2 Approche
 - Schéma d'étiquetage
- 3 Schéma planaire de connexité
 - Courcelle et al.
- 4 Contribution
 - Généralisation aux graphes plongeable sur une surface de genre eulérien g
- 5 Conclusion
- 6 annexes
 - Schéma de connectivité avec panne
 - Optimisation de l'algorithme de requête
 - Requête avec réparation
 - Implémentation

Graphe augmenté G^+



[B. Courcelle, C. Gavoille, M. M. Kanté, and D. A. Twigg. '08]

$\text{Bar}(X)$

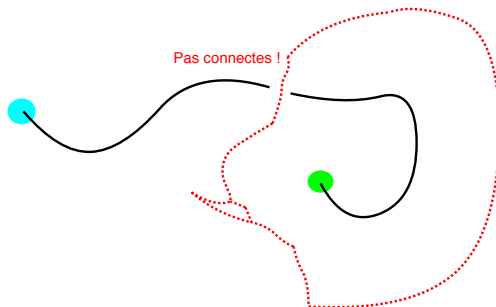


Bar(X)

Pour $X \subset V(G)$, on définit sa barrière $\text{Bar}(X)$ comme étant l'ensemble des arêtes de G^+ choisies tels que deux sommets $u, v \in V(G) \setminus X$ sont **séparés par X dans G si et seulement si ils sont séparés dans le plan par $\text{Bar}(X)$** .

$\text{Bar}(X)$ est l'ensemble des arêtes de G^+ qui relie une face f et un sommet x tel qu'il existe dans G^+ une autre arête reliant f à un sommet $y \in X$. (On aurait éventuellement $x = y$ dans le cas d'un graphe non 2-connexe.)

Requête géométrique

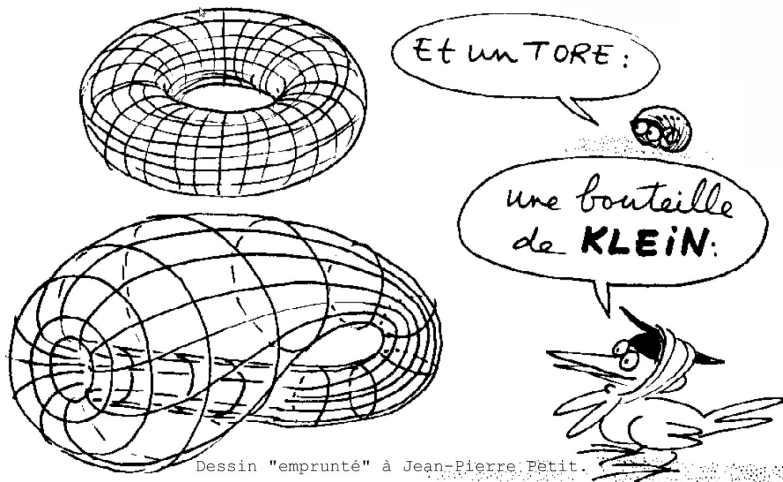


Moralité :

Les étiquettes contiennent suffisamment d'information pour **localiser** dans le plan les sommets à connecter et pour **construire** la barrière correspondant à notre panne.

- 1 Le problème
 - présentation
- 2 Approche
 - Schéma d'étiquetage
- 3 Schéma planaire de connexité
 - Courcelle et al.
- 4 Contribution**
 - Généralisation aux graphes plongeable sur une surface de genre eulérien g
- 5 Conclusion
- 6 annexes
 - Schéma de connectivité avec panne
 - Optimisation de l'algorithme de requête
 - Requête avec réparation
 - Implémentation

Genre Eulerien



Genre Eulerien

Genre Eulerien :

$$\min(2g, \tilde{g})$$

Théorème :

La classe des graphes à n sommets de genre Eulérien g admet un schéma d'étiquetage pour les requêtes de connexité avec obstacle produisant des étiquettes de $O(g \log n)$ bits. Les n étiquettes de chaque graphe peuvent être calculées en temps

$$O((n + g)(g + \log n))$$

étant donné un plongement de genre Eulérien g du graphe. Le temps de réponse à une requête $\text{CONN}(u, v, X)$ est

$$O(g|X| \cdot \log(g|X|))$$

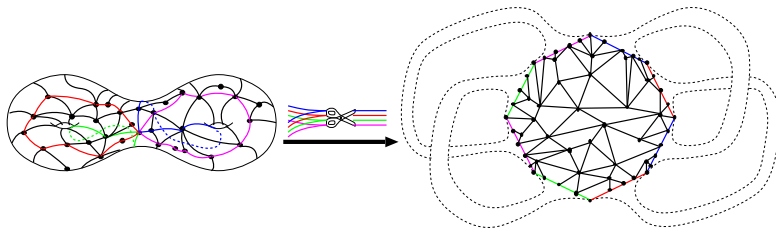
.

Idée

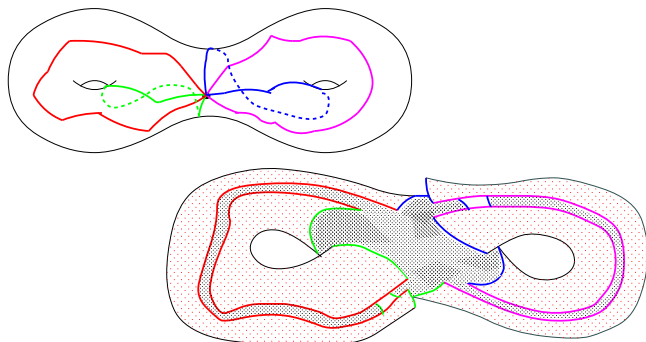
On réduit le problème au cas planaire en utilisant un schéma polygonal. (De taille $2g$)

On obtient des étiquettes de $O(g \cdot \log n)$ bits.

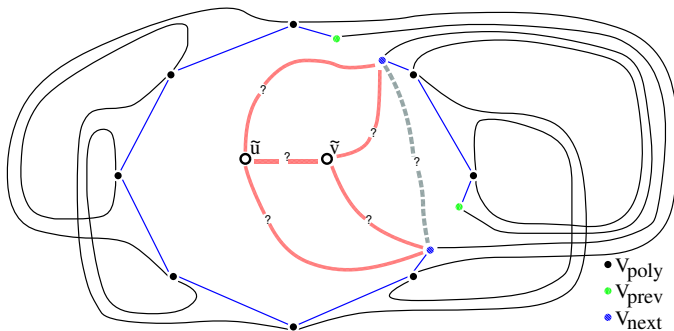
Réduction - étiquetage



Réduction - étiquetage



Réduction - requête



On peut construire G^+ en $O(g|X| \cdot \log(g|X|))$

- 1 Le problème
 - présentation
- 2 Approche
 - Schéma d'étiquetage
- 3 Schéma planaire de connexité
 - Courcelle et al.
- 4 Contribution
 - Généralisation aux graphes plongeable sur une surface de genre eulérien g
- 5 Conclusion**
- 6 annexes
 - Schéma de connectivité avec panne
 - Optimisation de l'algorithme de requête
 - Requête avec réparation
 - Implémentation

Conclusion

La requête $\text{CONN}(u, v, X)$ peut donc être résolue en temps $O(g|X|)$ une fois le graphe G^+ construit. Cependant, on peut raffiner le résultat en considérant le sous-graphe auxiliaire $G^+ \setminus \tilde{u}, \tilde{v}$, ne dépendant que de X , à partir duquel il est possible de construire (en temps linéaire) une structure de données surportant les requêtes de connexité (u, v) en temps $O(\log(g|X|))$, pour tout $u, v \notin X$. En utilisant des structures de données transdichotomiques [T. M. Chan and M. Pătraşcu. '09], il est même possible d'améliorer le temps de réponse à $\min\{\log(g|X|)/\log \log(g|X|), \sqrt{\log n / \log \log n}\}$ sans détériorer le temps de précalcul.

Merci !

(Tout est dans le titre du slide.)

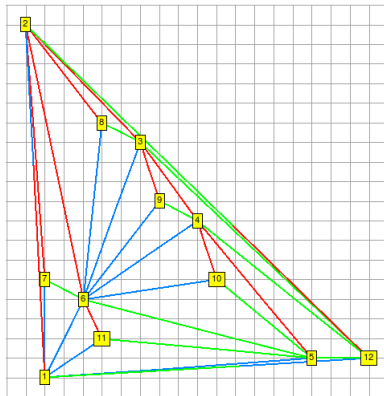
- 1 Le problème
 - présentation
- 2 Approche
 - Schéma d'étiquetage
- 3 Schéma planaire de connexité
 - Courcelle et al.
- 4 Contribution
 - Généralisation aux graphes plongeable sur une surface de genre eulérien g
- 5 Conclusion
- 6 annexes
 - Schéma de connectivité avec panne
 - Optimisation de l'algorithme de requête
 - Requête avec réparation
 - Implémentation

Question :

Dans quel plongement ?

Plongement en ligne droite dans la grille $n \times n$

[Schnyder][Bonichon, Felsner, Mosbah]



Plongement en ligne droite dans la grille - Avantages :

- Coordonnées entières représentables en $2 \cdot \log(n) + O(1)$
- Requête géométrique de type `point location` dans une subdivision du plan par des segments.

Soit Y une subdivision du plan composée de m ségments. On peut construire en un temps $O(m \cdot \log(m))$ une structure de donnée de taille $O(m)$ à partir de laquelle on peut tester en temps $O(\log(m))$ si deux éléments de $\mathbb{R}^2 \setminus Y$ sont séparés par Y . [De Berg, Cheong, Van Kreveld, Overmars]

Face commune

On a G^+ et les trois fonctions g_i^+ de Schnyder.

Face-Commune(x, y) $\iff$

$$\bigvee_{1 \leq i, j \leq 3} g_i^+(x) = g_j^+(y) \in F \quad (1a)$$

$$\vee \bigvee_{1 \leq i, j \leq 3} g_i^+(x) \in F \wedge g_j^+(g_i^+(x)) = y \quad (1b)$$

$$\vee \bigvee_{1 \leq i, j \leq 3} g_i^+(x) \in F \wedge g_j^+(g_i^+(y)) = x \quad (1c)$$

$$\vee \exists f \in F \left(\bigvee_{1 \leq i, j \leq 3} g_i^+(f) = x \wedge g_j^+(f) = y \right) \quad (1d)$$

Types de faces

$$\bigvee_{1 \leq i, j \leq 3} g_i^+(x) = g_j^+(y) \in F$$

$$\vee \bigvee_{1 \leq i, j \leq 3} g_i^+(x) \in F \wedge g_j^+(g_i^+(x)) = y$$

$$\vee \bigvee_{1 \leq i, j \leq 3} g_i^+(x) \in F \wedge g_j^+(g_i^+(y)) = x$$

$$\vee \exists f \in F \left(\bigvee_{1 \leq i, j \leq 3} g_i^+(f) = x \wedge g_j^+(f) = y \right)$$

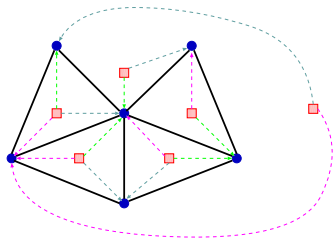
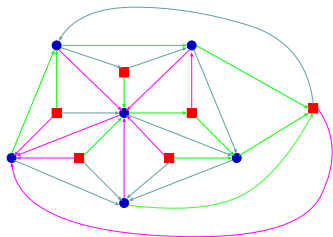


Graphe auxiliaire H

$$V(H) = V(G)$$

$x - y$ est une arête $\iff \exists i, j \in \{1, 2, 3\}$ et $f \in F$ tels qu'on ait $g_i^+(f) = x$ et $g_j^+(f) = y$. H admet un haie de Schnyder, et donc des fonctions h_i définissant l'adjacence.

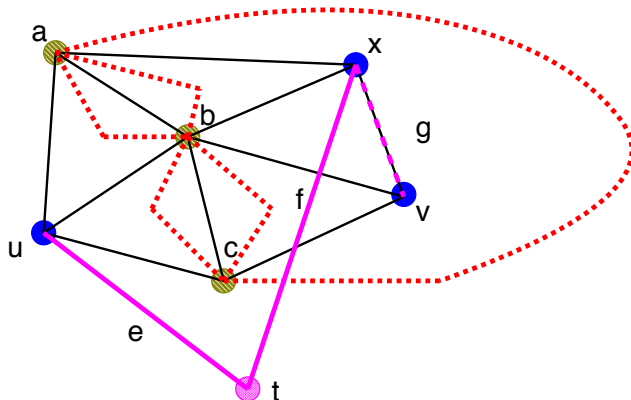
Graphe auxiliaire H



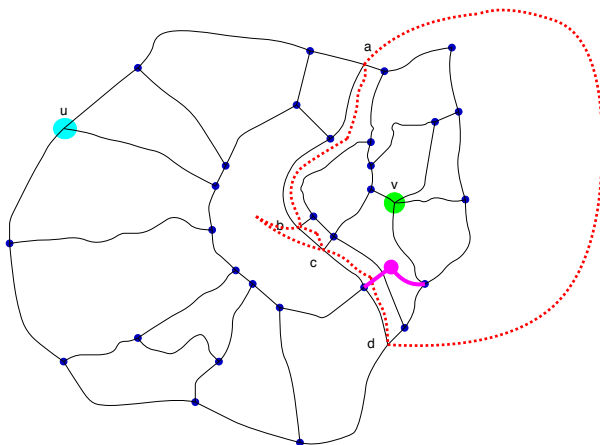


Optimisation de l'algorithme de requête

On n'utilise plus un algorithme généraliste de point-location mais un algorithme spécialisé pour notre problème. On passe d'une complexité quadratique à une complexité linéaire en fonction de la taille de la panne.



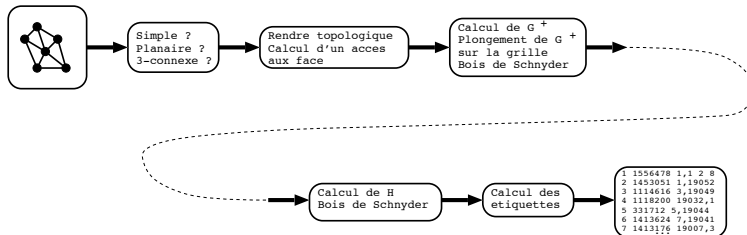
Réparation = pont



Requête avec réparation :

```
#Sommets à connecter
3 1855480 3,10 2 1 4 1 4 1,-3 -1,0 1,-3
6 1957824 7,4 2,0 1 5 2 3 5 4 -1,0
#Sommets en panne
1 507902 1,1 2 6 5,3 7,1
2 1117179 1,13 12,-12 1 1 3 5 1,-3 8,-9 1,-3 8,-9
#Sommets de la réparation (vn = nouveau sommet)
vn
v4 1994688 5,7 4,-3 1 6 2 3 5 1 1,-3 -1,0
v3 1855480 3,10 2 1 4 1 4 1,-3 -1,0 1,-3
#Arêtes (nouveau sommet connecté à 2 sommets du graphe de départ)
e0-1
e0-2
```

Etiquetage



Requete

```
1 1556478 1,1 2 8 76
78 1453051 1,19052 75
#panne
35 1114616 3,19049 87
894 1118200 19032,1 9
55 331712 5,19044 98
65 1413624 7,19041 75
7 1413176 19007,3 75
#reparation
#somet 0
v75 335412 5,19674 64
#somet 1
v65 654754 756,970 79
#arete
e0-1
```

Analyse syntaxique
de la requete

Calcul des segments
de la barriere dans
le plan.

Calcul de localisation

OUI /NON

Calcul du graphe
de reparation

Connexion dans
reparation

OUI /NON