

# Soutenances d'*Algorithmes du Monde Réel*

Ludovic Brochard, Fabien Kuntz, Benoît Védrenne

Professeurs : Cyril Gavoille, Anca Muscholl, Marc Zeitoun, Alexander Zvonkin

Module AMR

Master S&T Informatique

Université Bordeaux I

22 janvier 2009 - 11h40

## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

## 4-Col

## Données

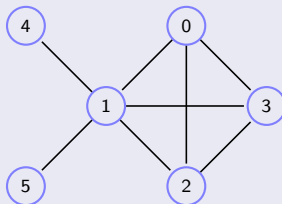


FIG. : Graphe poisson

## Question

Le graphe est-il 4-coloriable ?

## 4-Col

## Données

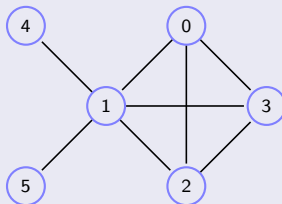


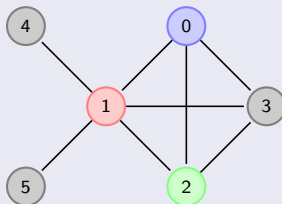
FIG. : Graphe poisson

## Question

Le graphe est-il 4-coloriable ?

## 4-Col

## Solution

FIG. : Graphe *poisson* 4-colorié

## Question/Réponse

Le graphe est-il 4-coloriable ? Oui ! 😊

## 3-Col

## Données

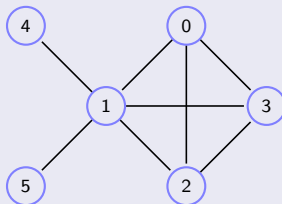


FIG. : Graphe poisson

## Question

Le graphe est-il 3-coloriable ?

## 3-Col

## Données

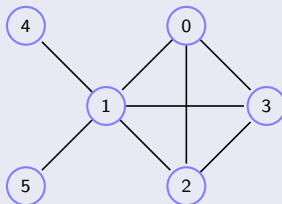


FIG. : Graphe poisson

## Question

Le graphe est-il 3-coloriable ?



## 3-Col

## Données

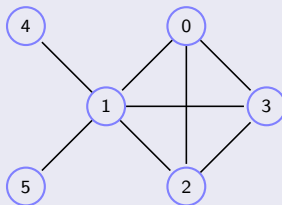


FIG. : Graphe poisson

## Question/Réponse

Le graphe est-il 3-coloriable ? **Non !** ☹️

# Clique - nombre minimum d'arêtes

Données

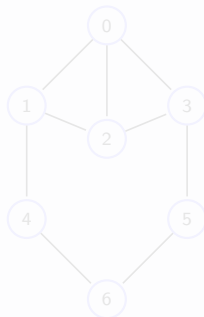


FIG. : Graphe *souris*

Question

Le graphe comporte-t-il une clique de taille 5 ?

# Clique - nombre minimum d'arêtes

## Données

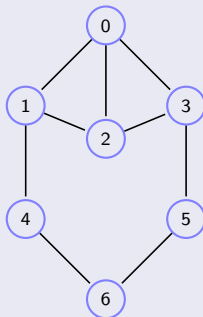


FIG. : Graphe *souris*

## Question

Le graphe comporte-t-il une clique de taille 5 ?

# Clique - nombre minimum d'arêtes

## Données

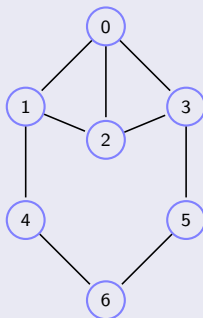


FIG. : Graphe *souris*

## Question

Le graphe comporte-t-il une clique de taille 5 ?

# Clique - nombre minimum d'arêtes

## Données

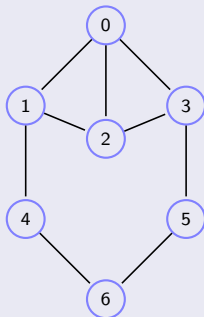


FIG. : Graphe *souris*

## Question/Réponse

Le graphe comporte-t-il une clique de taille 5 ? (Nombre d'arêtes du graphe = 9) et (Minimum d'arêtes dans une clique de taille 5 = 10)  $\Rightarrow$  Non

## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

# Algorithme

## Idée

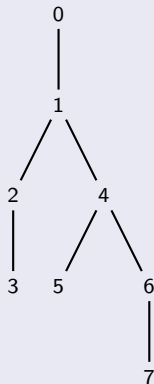
- Méthode : Parcours en profondeur
- Aller : Marquer les pères d'au moins une feuille
- Retour : Marquer les sommets pères de sommets non marqués

## Complexité

La complexité est celle du parcours en profondeur :  $O(|V| + |E|)$ .

# Exécution

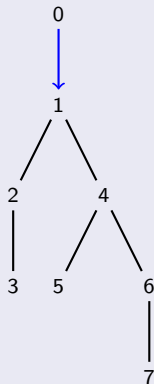
## Application de l'algorithme





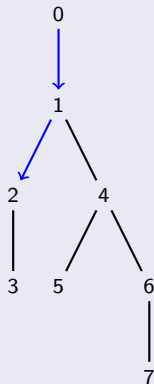
# Exécution

## Application de l'algorithme



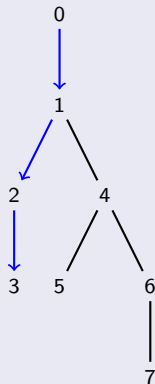
# Exécution

## Application de l'algorithme



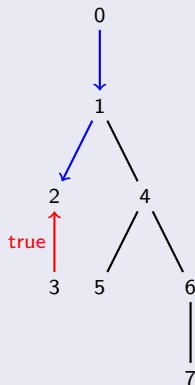
# Exécution

## Application de l'algorithme



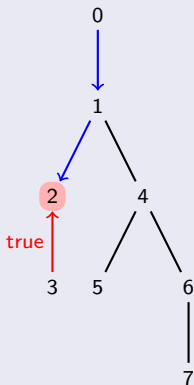
# Exécution

## Application de l'algorithme



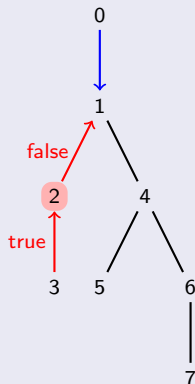
# Exécution

## Application de l'algorithme



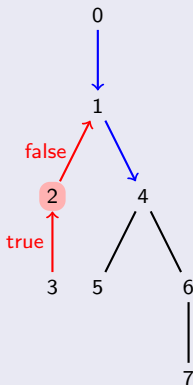
# Exécution

## Application de l'algorithme



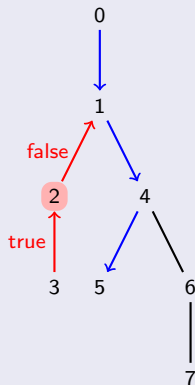
# Exécution

## Application de l'algorithme



# Exécution

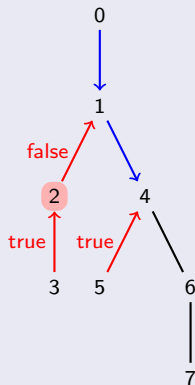
## Application de l'algorithme





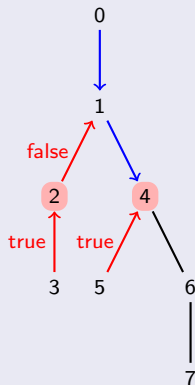
# Exécution

## Application de l'algorithme



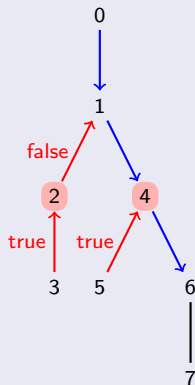
# Exécution

## Application de l'algorithme



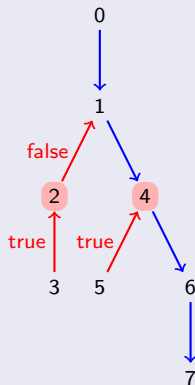
# Exécution

## Application de l'algorithme



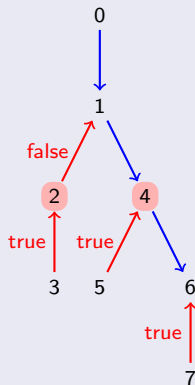
# Exécution

## Application de l'algorithme



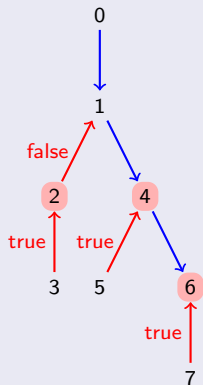
# Exécution

## Application de l'algorithme



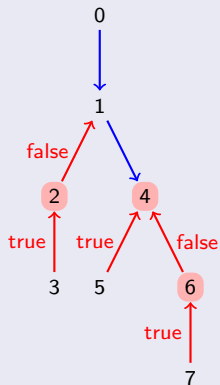
# Exécution

## Application de l'algorithme



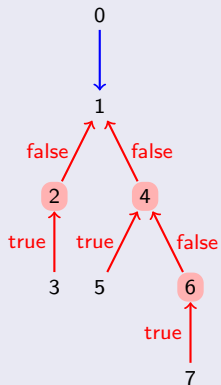
# Exécution

## Application de l'algorithme



## Exécution

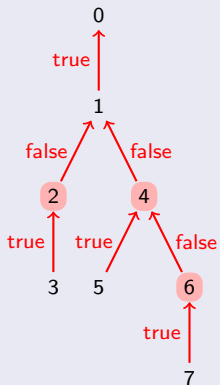
## Application de l'algorithme





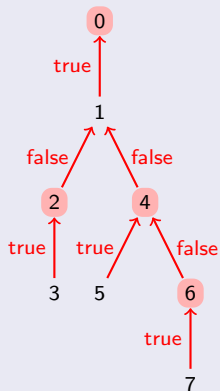
# Exécution

## Application de l'algorithme



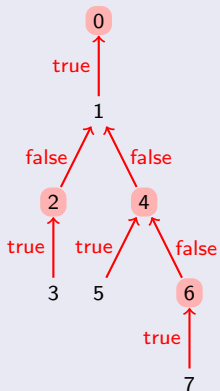
# Exécution

## Application de l'algorithme



## Exécution

## Solution



Couverture minimale =  $\{0,2,4,6\}$

## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

# Rappel

## Algorithme du projet (sommets non feuilles)

- Calcul d'un arbre couvrant par parcours en profondeur
- Couverture = ensemble des sommets non feuilles (par parcours en profondeur)

## Algorithme du cours (couplage maximal)

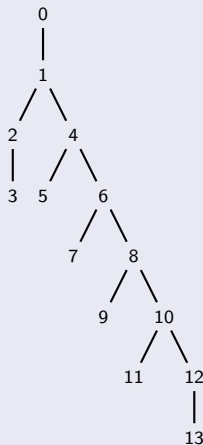
- Couverture = ensemble des sommets d'un couplage maximal

## Complexités

- Algorithme du projet : linéaire car consiste en deux parcours en profondeur l'un après l'autre.
- Avec une structure de graphe adapté : linéaire.

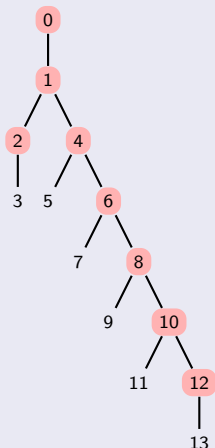
# Comparaison des algorithmes d'approximation (1/3)

Données : Graphe de comparaison *Pise*

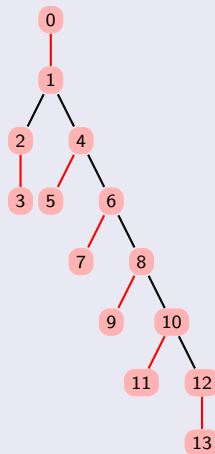


# Comparaison des algorithmes d'approximation (2/3)

## Comparaison des solutions sur le graphe *Pise*



$$C_{\text{projet}} = \{0, 1, 2, 4, 6, 8, 10, 12\}$$



$$C_{\text{cours}} = V$$

# Comparaison des algorithmes d'approximation (3/3)

## Tableau de comparaison

	jeuPise.gin		testTree100.gin		testTree1000.gin	
	algoP	algoC	algoP	algoC	algoP	algoC
Temps d'exécution(en $\mu s$ )	58.5808	1.12014	400.344	5.82175	3911.25	56.8837
Taille de la couverture	8	14	38	58	365	576
Taille couverture optimale	7		31		325	

FIG. : Tableau comparatif des exécutions des algorithmes sur plusieurs exemples.



## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

# Graphe critique pour les algorithmes d'approximation

## Données

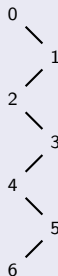
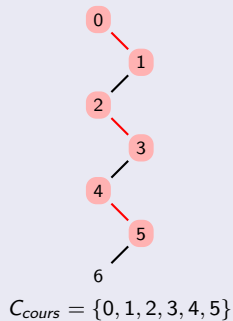
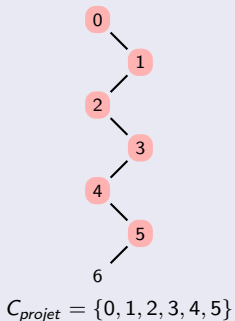


FIG. : Graphe *serpent*

# Solution pour le graphe *serpent*

## Comparaison des solutions sur le graphe *serpent*



# Graphe critique pour les algorithmes d'approximation

## Couverture minimale sur le graphe *serpent*



Couverture minimale =  $\{1, 3, 5\}$

## 1 Réductions

- Instance positive
- Instance négative
- Bonus : les “cas faciles”

## 2 Couverture par sommets

- Présentation de l'algorithme
- Exemple d'exécution

## 3 Algorithmes d'approximations

- Bref rappel des algorithmes
- Comparaison

## 4 Instances critiques

## 5 Bilan

# Bilan

## Améliorations

- Partie *Réduction* :
  - Utiliser des structures de graphes différentes selon les réductions.
  - Trouver plus de “cas faciles”
- Partie *Approximation* :
  - Utiliser une structure de graphe plus adaptée à la suppression des arêtes pour le couplage maximum.
  - Améliorer le *parser* qui est la cause d'exécutions lentes.

## Conclusion

- Nous avons remarqué l'intérêt d'utiliser des algorithmes d'approximation lorsque le calcul d'une solution exacte est trop coûteux
- Un des rares projets où la théorie est au service de la pratique

# Bilan

## Améliorations

- Partie *Réduction* :
  - Utiliser des structures de graphes différentes selon les réductions.
  - Trouver plus de “cas faciles”
- Partie *Approximation* :
  - Utiliser une structure de graphe plus adaptée à la suppression des arêtes pour le couplage maximum.
  - Améliorer le *parser* qui est la cause d'exécutions lentes.

## Conclusion

- Nous avons remarqué l'intérêt d'utiliser des algorithmes d'approximation lorsque le calcul d'une solution exacte est trop coûteux
- Un des rares projets où la théorie est au service de la pratique

# Bilan

## Améliorations

- Partie *Réduction* :
  - Utiliser des structures de graphes différentes selon les réductions.
  - Trouver plus de “cas faciles”
- Partie *Approximation* :
  - Utiliser une structure de graphe plus adaptée à la suppression des arêtes pour le couplage maximum.
  - Améliorer le *parser* qui est la cause d'exécutions lentes.

## Conclusion

- Nous avons remarqué l'intérêt d'utiliser des algorithmes d'approximation lorsque le calcul d'une solution exacte est trop coûteux
- Un des rares projets où la théorie est au service de la pratique