



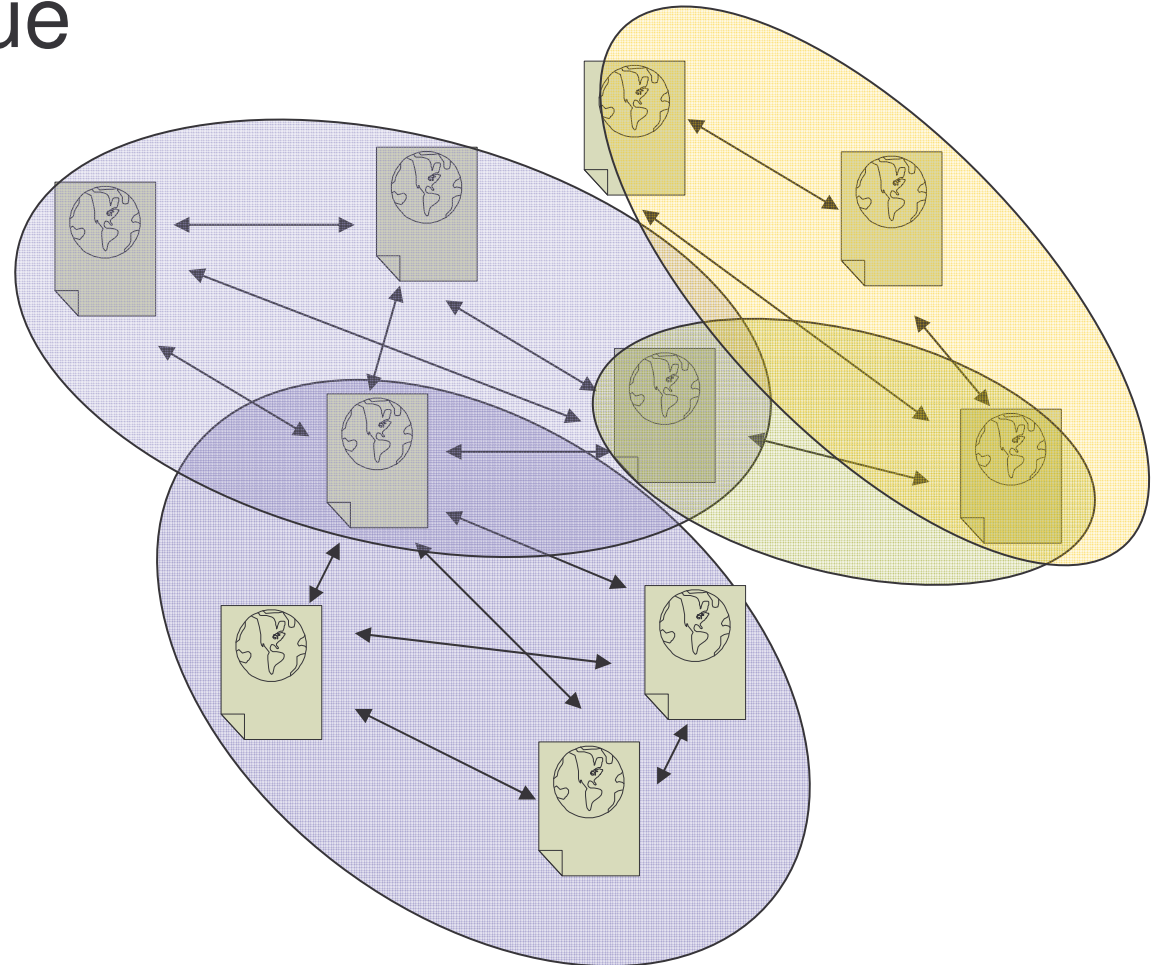
# Cliques & Bicliques Maximales

Vers une présentation unifiée des  
algorithmes d'énumération

Alain Gély – LIMOS – Clermont-Ferrand

# Cliques maximales & fouille de données

- Bio-Informatique
- Web Mining





# Bicliques maximales & fouille de données

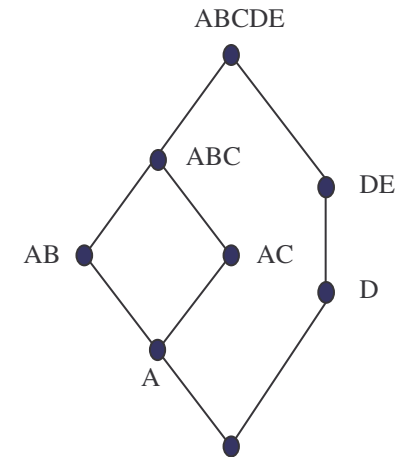
- Itemset fermés (fréquent ou non)
- Utile pour les règles d'implications...

# Énumération vs Construction

## CONSTRUCTION

- Graphe de couverture
- Taille exponentielle
- Complexité théorique  $O(n^2) \cdot |C|$

	1	2	3	4	5
A	X	X		X	
B	X			X	
C		X		X	
D			X		X
E					X



Entrée

Sortie

## GENERATION

- Énumération des fermés
- Espace polynomial
- Complexité théorique  $O(n^3) \cdot |C|$

	1	2	3	4	5
A	X	X		X	
B	X			X	
C		X		X	
D			X		X
E					X

$\emptyset, A, AB, ABC, AC,$   
 $D, DE, ABCDE$

# Generation vs Construction

Entrée

	1	2	3	4	5
A	X	X		X	
B	X			X	
C		X		X	
D			X		X
E					X

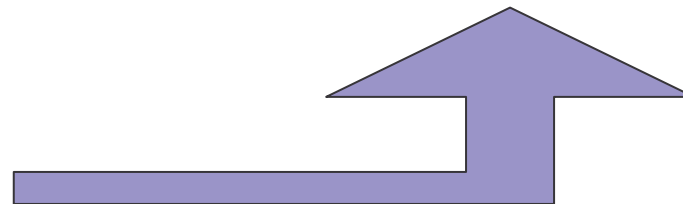
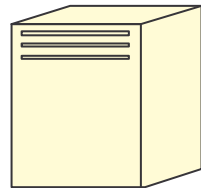
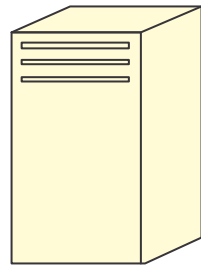
Sortie

$\emptyset$ , A, AB, ABC,  
AC, D, DE, ABCDE

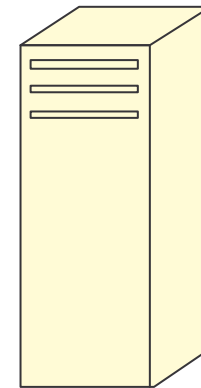
# Pourquoi juste énumérer ?

- Délai polynomial
- Utile en « pipe line »

Enumération



Construction / Traitement





# Plusieurs sources d'algorithmes

- Fouille de données
- Analyse de Concept Formel
- Algorithmique des systèmes de fermeture
- Analyse de données / problèmes matriciels
- Algorithmes de graphes



# Fouille de données

- CHARM
- CLOSET, CLOSET+
- LCM, LCM v2
- Apriori






# Analyse de concepts formel

- Lindig (Fast Concept Analysis)
- Vatchev et al.
- Godin et al.



# Algorithmique des systèmes de fermeture

- Ganter (Next-closure)
- Nourine-Raynaud



# Analyse de données / problèmes matriciels

- Algorithme de Chein
- Algorithme de Bordat

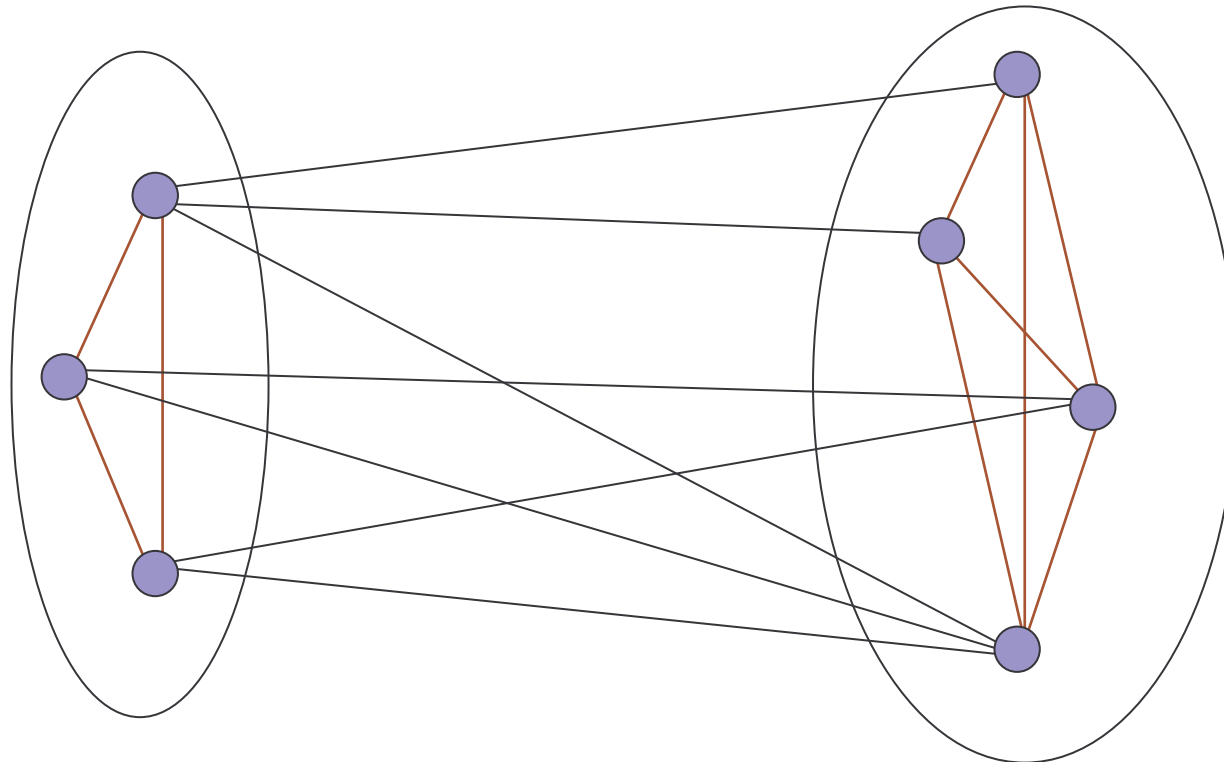


# Algorithmes de graphe

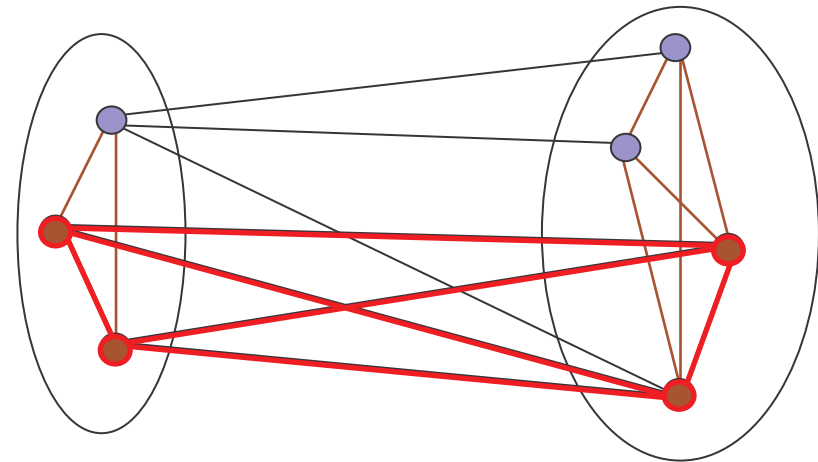
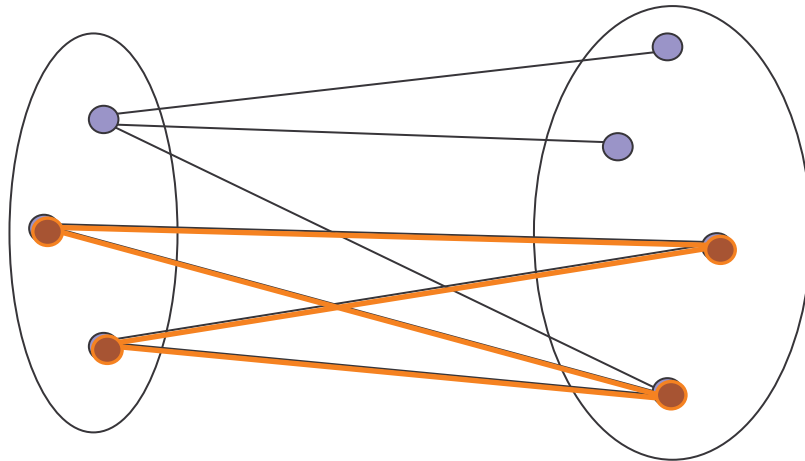
- Enumération des cliques maximales
  - Algorithme de Tsukiyama et al
  - Algorithme de Johnson et al
  - Algorithmes de Makino et Uno
  - ...

# Cliques & bicliques maximales

- Transformation classique



# Graphe de transition : Cas des bicliques maximales





# Comparaisons des algorithmes ?

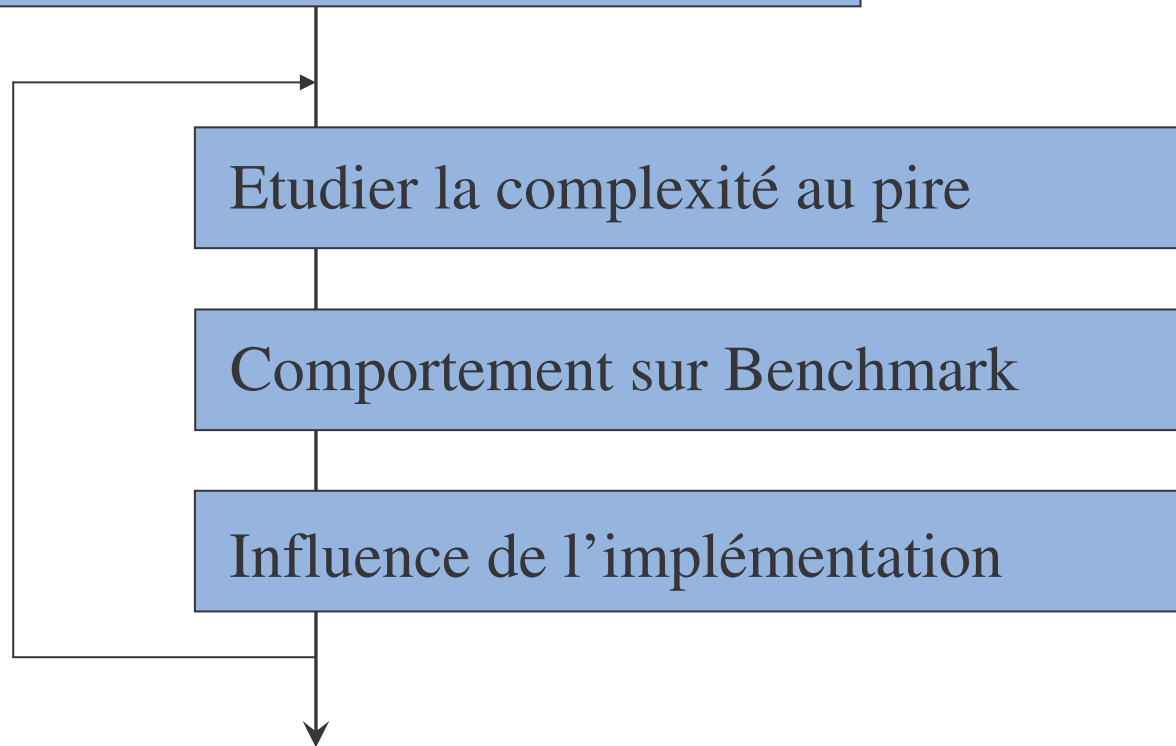
# Comparaisons des algorithmes ?

Pour un algorithme...

Etudier la complexité au pire

Comportement sur Benchmark

Influence de l'implémentation







# Comparaisons des algorithmes ?

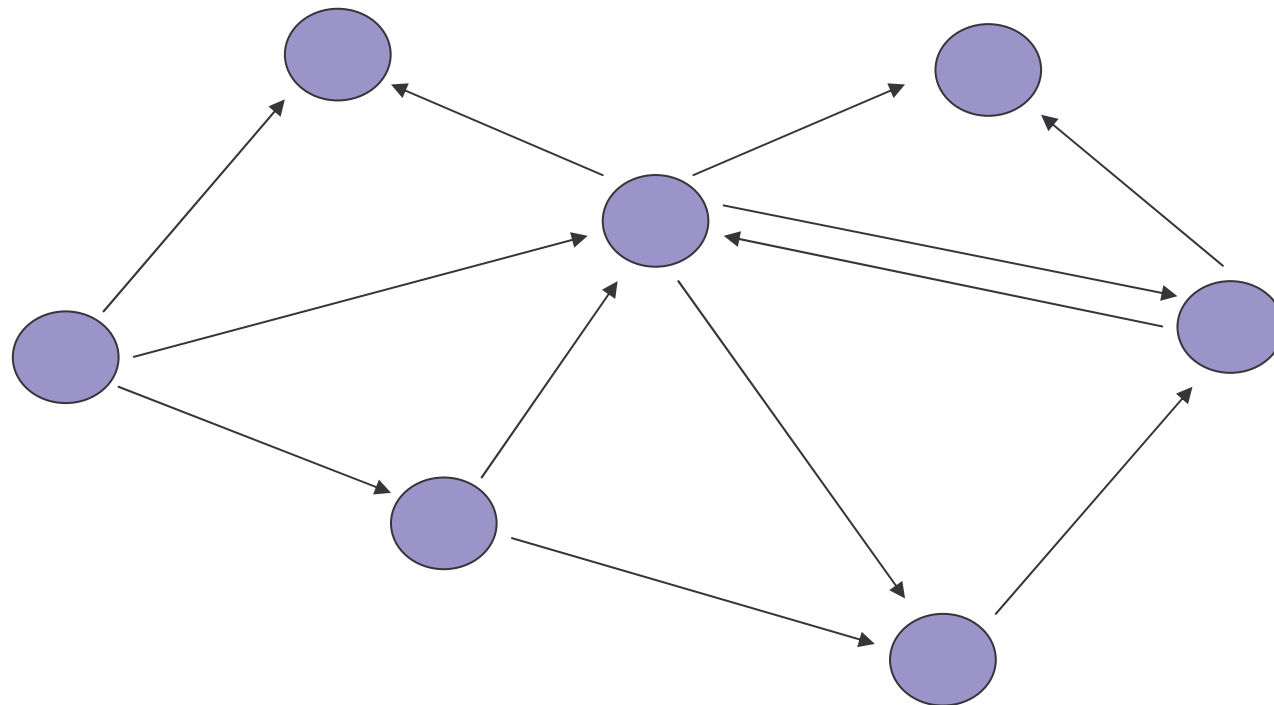
Deux questions

Peut-on expliquer les différences de comportement des algorithmes par des **Propriétés structurelles** ?

Les structures de données peuvent jouer un rôle important sur l'efficacité.

Y-a-t-il d'autres facteurs influençant les performances ?

# Graphe de transition des cliques maximales d'un graphe





# Unifier la présentation des algorithmes existants ?

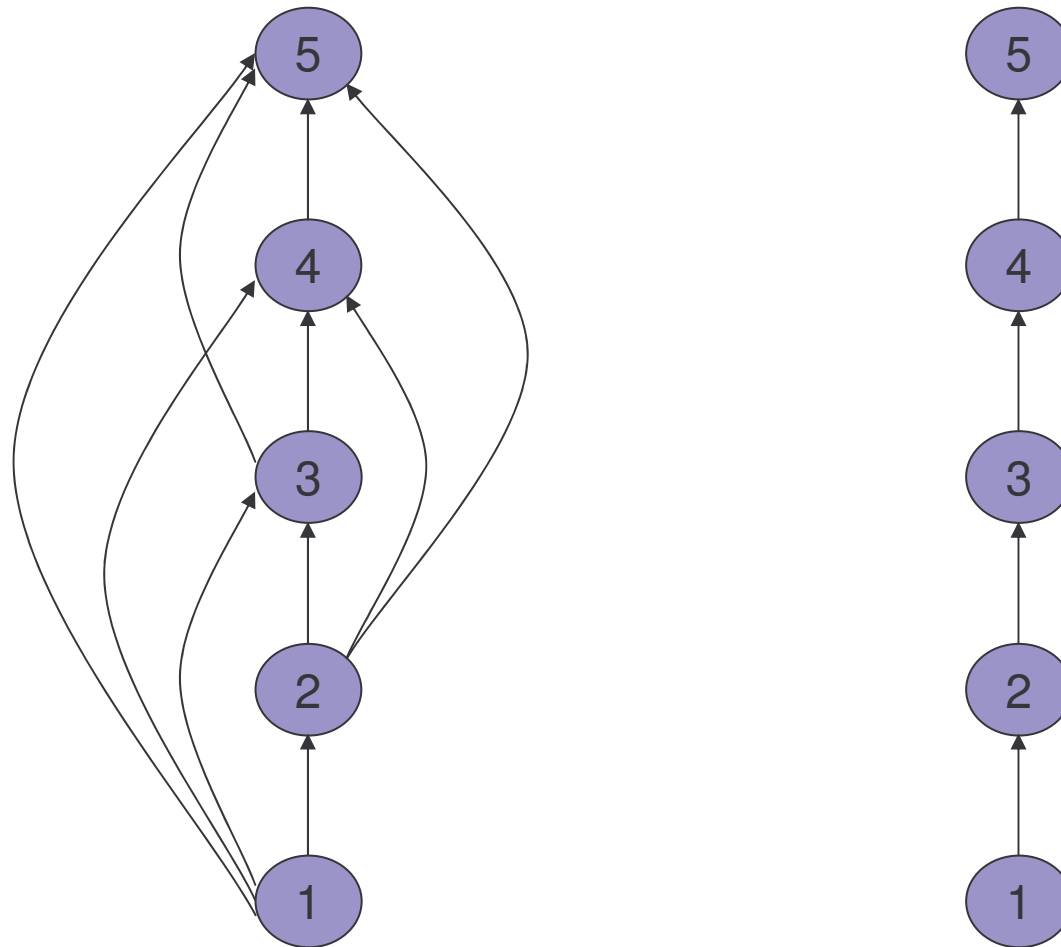
- Structure élémentaire [ Graphe ]
- Briques élémentaires sur cette structure
  - Passage d'un élément à un autre [ arêtes ]
  - Vérification d'appartenance [ arbre couvrant ]



# Graphe de transition des bicliques maximales

- Cas particulier des graphes
- Propriétés particulières de numérotation
  - $G(U, V, E)$
  - Hypothèse :  $U > V$
- Partitionnement sur les éléments de  $V$ 
  - *Transitions « lexicographique »*
  - *Graphe de couverture contenu dans le graphe de transition*

# Graphe de transition entre bicliques





## *Next-Closure & Ordre Lectique*

- Légume/Vegetable ,  $\emptyset$
- Litchi/Lychee ,  $\{A_4\}$
- Orge/Barley ,  $\{A_3\}$
- Pomme/Apple ,  $\{A_2\}$
- $\emptyset$  ,  $\{A_1\}$

Même structure de treillis

**Mais**

Un comportement différent des algorithmes

## Données « En français »

$B$	$C(B)$	$F <_i C(B)$	Réponse
• Next-Closure( $\emptyset$ )=1			
$\{\emptyset \cap 123\} \cup 4$	$C(4) = 1234$	$\emptyset <_4 1234$	Non (invalide)
$\{\emptyset \cap 123\} \cup 4$	$C(3) = 123$	$\emptyset <_3 123$	Non (invalide)
$\{\emptyset \cap 123\} \cup 4$	$C(2) = 12$	$\emptyset <_2 12$	Non (invalide)
$\{\emptyset \cap 123\} \cup 4$	$C(2) = 12$	$\emptyset <_1 1$	Oui
• Next-Closure(1)=12			
$\{1 \cap 123\} \cup 4$	$C(14) = 1234$	$1 <_4 1234$	Non (invalide)
$\{\emptyset \cap 12\} \cup 3$	$C(13) = 123$	$\emptyset <_3 123$	Non (invalide)
$\{1 \cap 1\} \cup 2$	$C(12) = 12$	$1 <_2 12$	Oui
• Next-Closure(12)=123			
$\{12 \cap 123\} \cup 4$	$C(124) = 1234$	$12 <_4 1234$	Non (invalide)
$\{12 \cap 12\} \cup 3$	$C(123) = 123$	$12 <_3 123$	Oui
• Next-Closure(123)=1234			
$\{123 \cap 123\} \cup 4$	$C(1234) = 1234$	$123 <_4 1234$	Oui

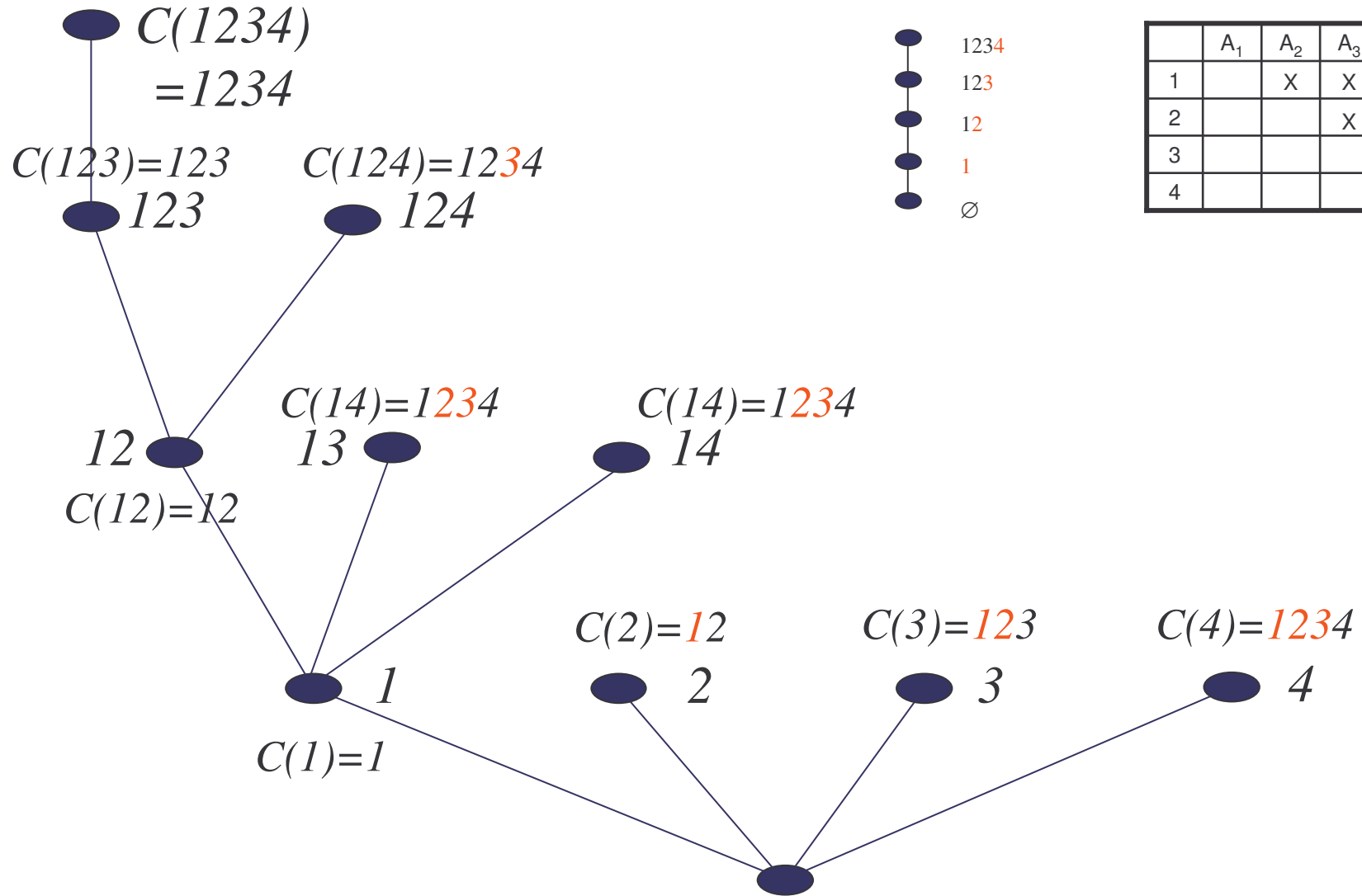
## Données « En anglais »

$B$	$C(B)$	$F <_i C(B)$	Réponse
• Next-Closure( $\emptyset$ )=4			
$\{\emptyset \cap 123\} \cup 4$	$C(4) = 4$	$\emptyset <_4 4$	Oui
• Next-Closure(4)=34			
$\{4 \cap 12\} \cup 3$	$C(3) = 34$	$4 <_3 34$	Oui
• Next-Closure(34)=234			
$\{34 \cap 1\} \cup 2$	$C(2) = 234$	$34 <_2 234$	Oui
• Next-Closure(123)=1234			
$\{234 \cap \emptyset\} \cup 1$	$C(1234) = 234$	$123 <_1 1234$	Oui

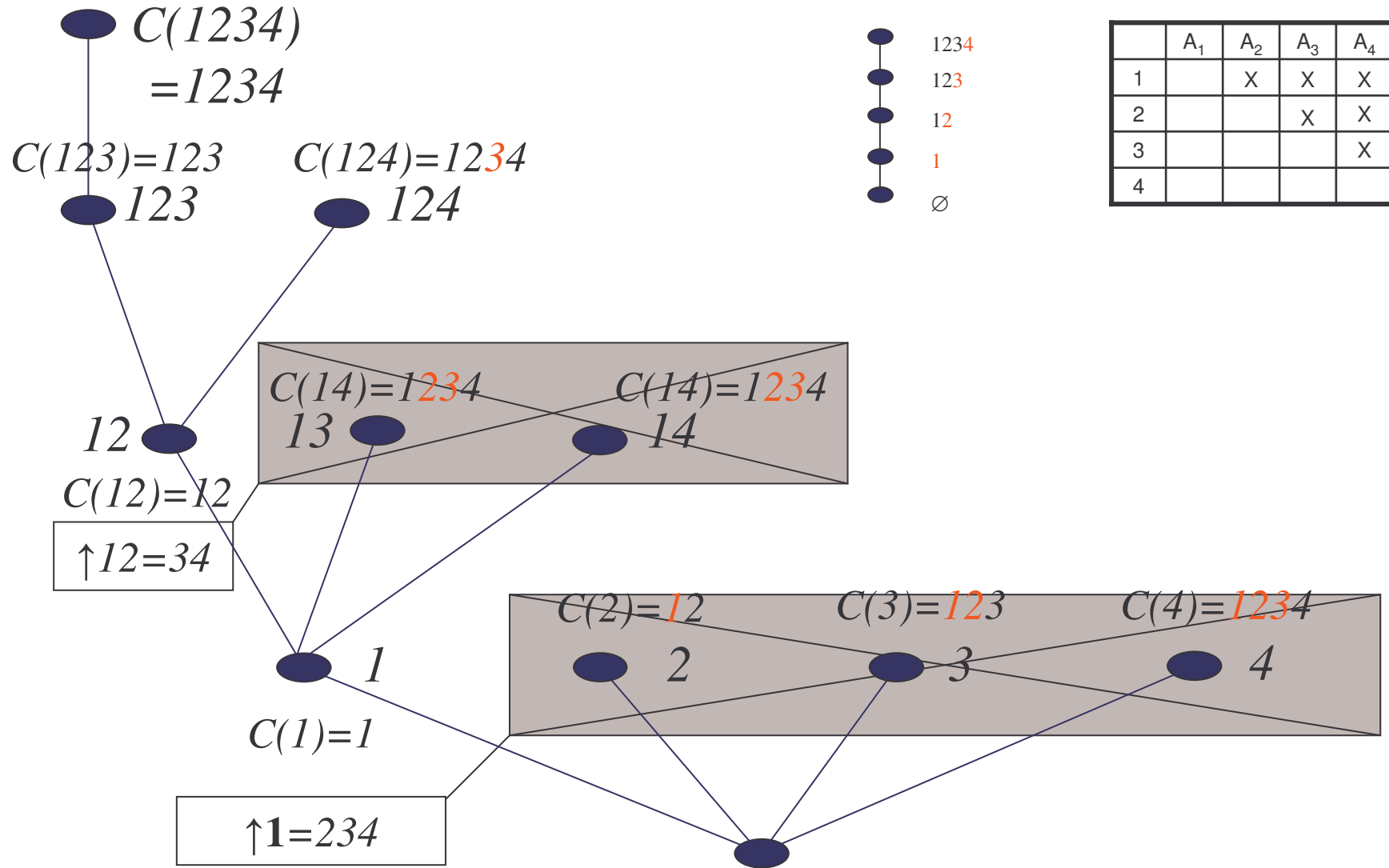
4 Calculs de fermeture



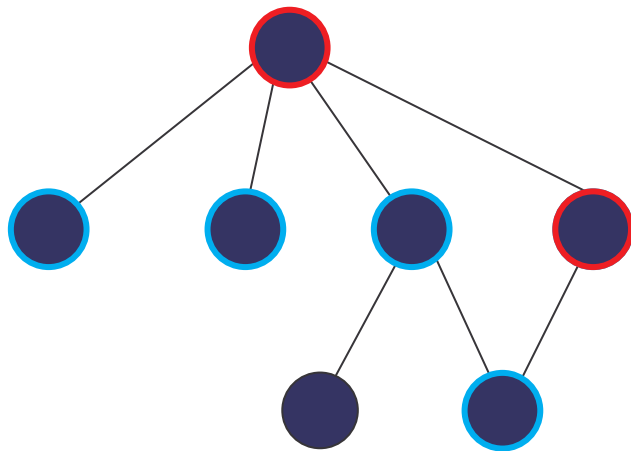
# Exécution de Next-closure



# Exécution de Next-closure - Amélioration



# *Algorithmes parcourant le graphe de couverture*

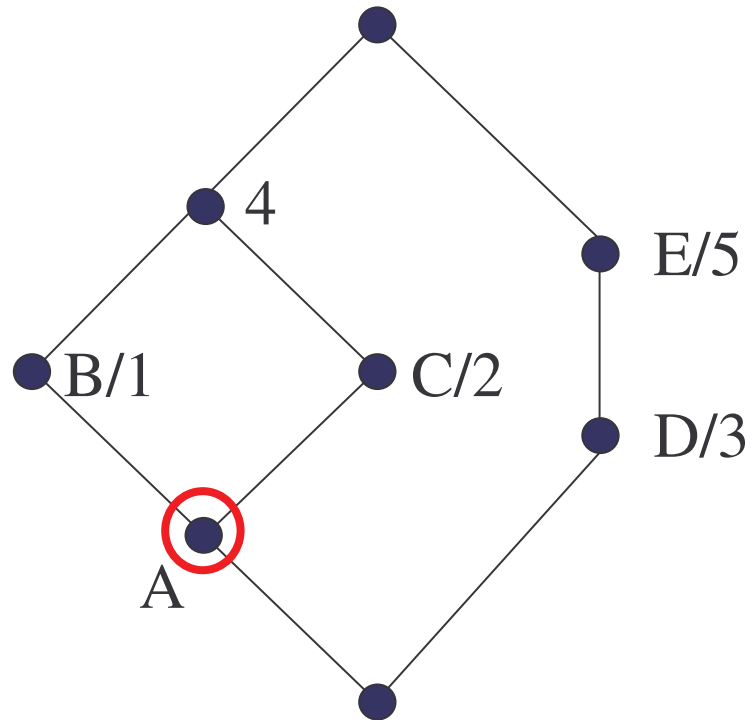


- Approche « top-down »

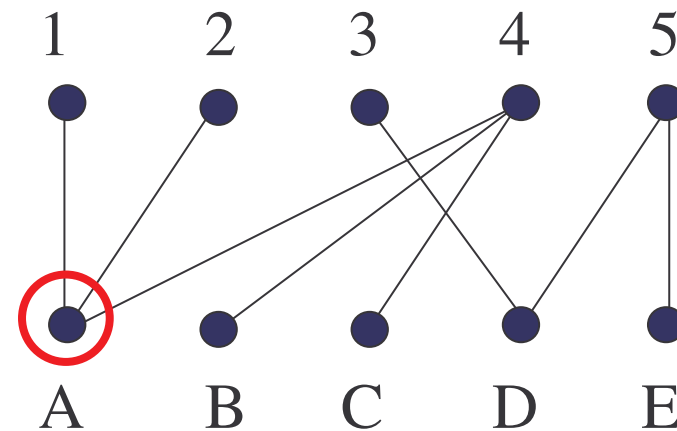
- D'un élément vers un prédécesseur

- Test d'unicité

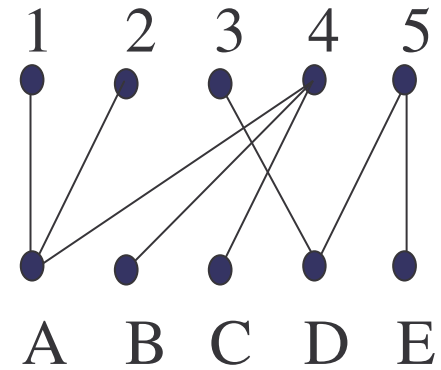
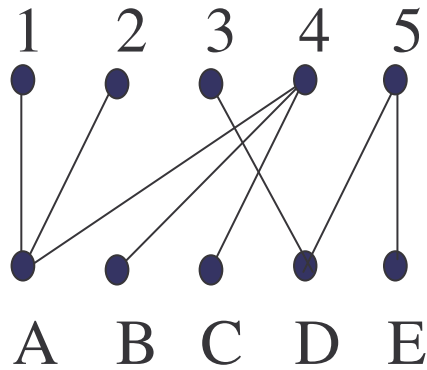
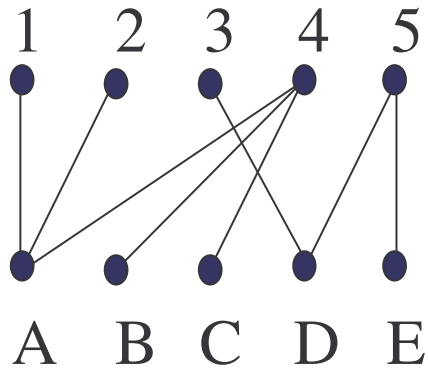
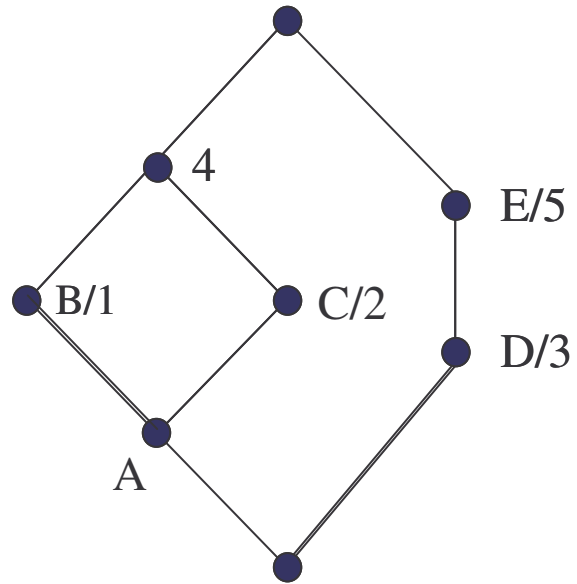
# Parcours de graphes en profondeur



	1	2	3	4	5
<b>A</b>	X	X		X	
<b>B</b>	X			X	
<b>C</b>		X		X	
<b>D</b>			X		X
<b>E</b>					X



# Connaître une arête de l'arbre couvrant ?





# *Enumérer les ensembles fermés*

GENERER (R)

**Choisir** un élément  $a$  de  $J$

Calculer  $C(a)$

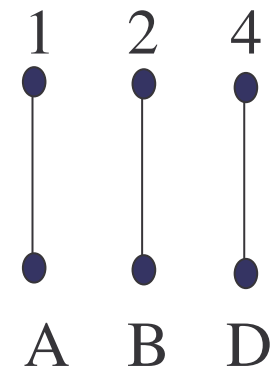
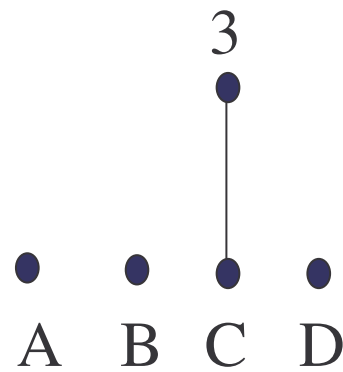
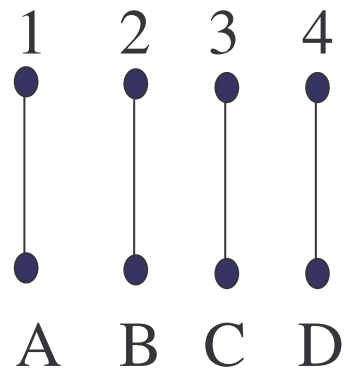
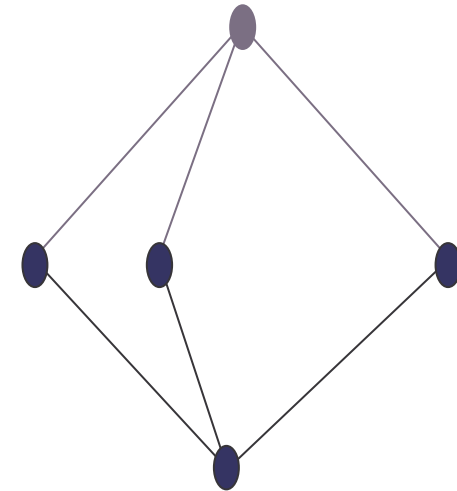
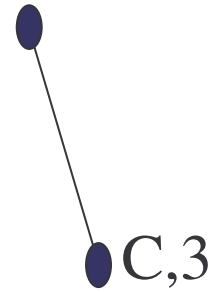
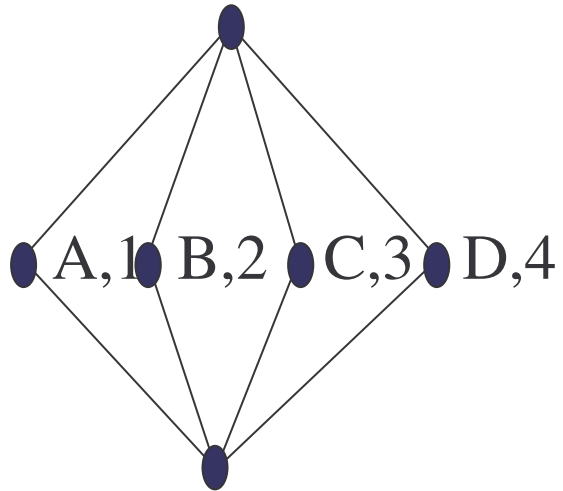
Afficher  $C(a)$

$R_1 \leftarrow (J, a', I_1) ; \text{GENERER } (R_1)$

$R_2 \leftarrow (J \setminus \uparrow a, M, I_2) ; \text{GENERER } (R_2)$

●

# *Systemes de fermeture resultant*



# *Méthode générique d'énumération*

GENERER (R)

**Choisir** un élément  $a$  de  $J$

Calculer  $C(a)$

Si  $C(a)$  **est valide**

$R_1 \leftarrow (J, a', I_1) ; \text{GENERER } (R_1)$

Afficher  $C(a)$

$R_2 \leftarrow (J \setminus \uparrow a, M, I_2) ; \text{GENERER } (R_2)$





## *Next-closure (ordre lectique)*

GENERER (R)

**Choisir**  $a$ , *élément de plus petit label* de  $J$

Calculer  $C(a)$

Si  $C(a)$  est valide

$R_1 \leftarrow (J, a', I_1) ; \text{GENERER } (R_1)$

Afficher  $C(a)$

$R_2 \leftarrow (J \setminus \uparrow a, M, I_2) ; \text{GENERER } (R_2)$



## *Next-closure (ordre lexicographique)*

GENERER (R)

**Choisir**  $j$  *élément de plus petit label* de  $J$

Calculer  $C(a)$

Si  $C(a)$  est valide

$R_1 \leftarrow (J, a', I_1) ; \text{GENERER}(R_1)$

Afficher  $C(a)$

$R_2 \leftarrow (J \setminus \uparrow a, M, I_2) ; \text{GENERER}(R_2)$



# Algorithme parcourant le graphe de couverture

GENERER (R)

**Choisir**  $a$ , *élément minimal* de  $J$

Calculer  $C(a)$

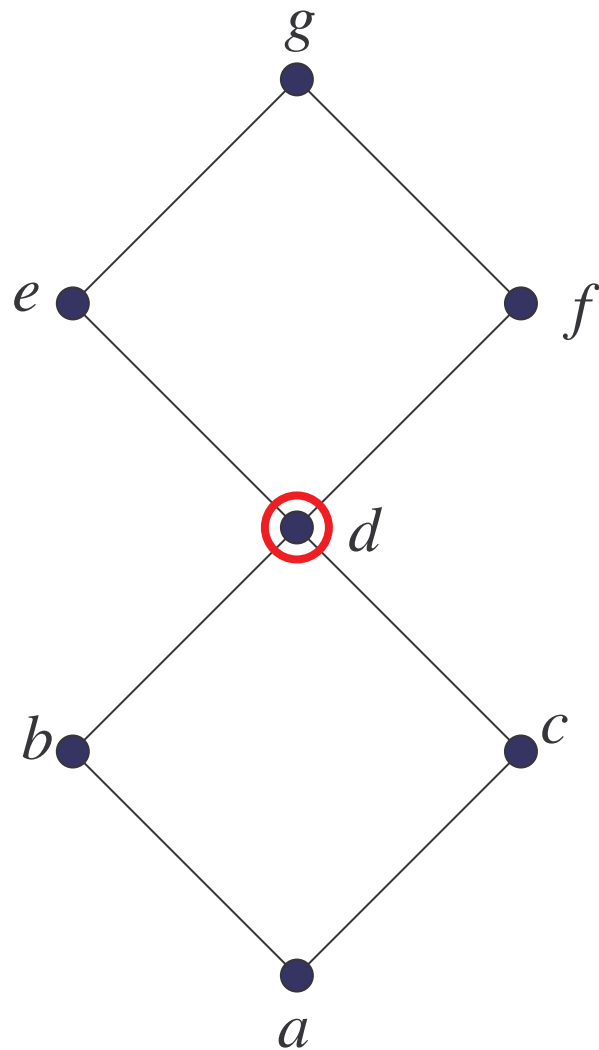
Si  $C(a)$  est valide

$R_1 \leftarrow (J, j', I_1) ; \text{GENERER}(R_1)$

$R_2 \leftarrow (J \setminus \uparrow j, M, I_2) ; \text{GENERER}(R_2)$

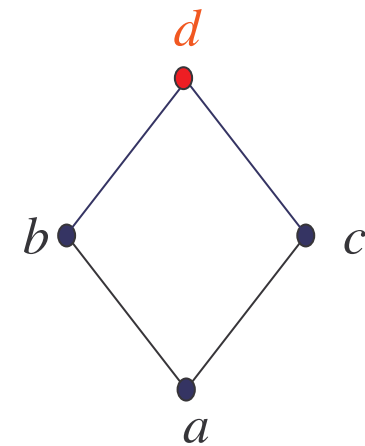
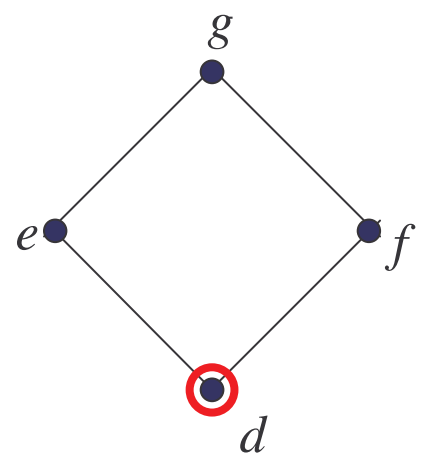


# Choix d'un élément non irréductible



Pour séparer le système de fermeture

Un élément non *Sup-irréductible* n'est pas un bon choix





## *Heuristique de choix d'élément*

- Au hasard

- Élément de plus petit label (Next Closure)

- Élément maximal (de l'ordre induit par J)

- Élément minimal (parcours du graphe de couverture)

- ...



## *Conclusion*

Trouver un compromis entre :

- La complexité de la fonction choix
- Le nombre de fermeture calculé (#invalide)

Perspectives :

- methode de comparaison d'algorithmes
- Etude de l'influence pratique de fonctions choix
- Fonctions choix spécifiques à certaines situations ?