Chemins

Modélisation

Algorithme de Dijkstra

Requêtes

Institut National des Sciences Appliquées – Rouen
Département Architecture des Systèmes d'Information
michel.mainguenaud@insa-rouen.fr

Approches

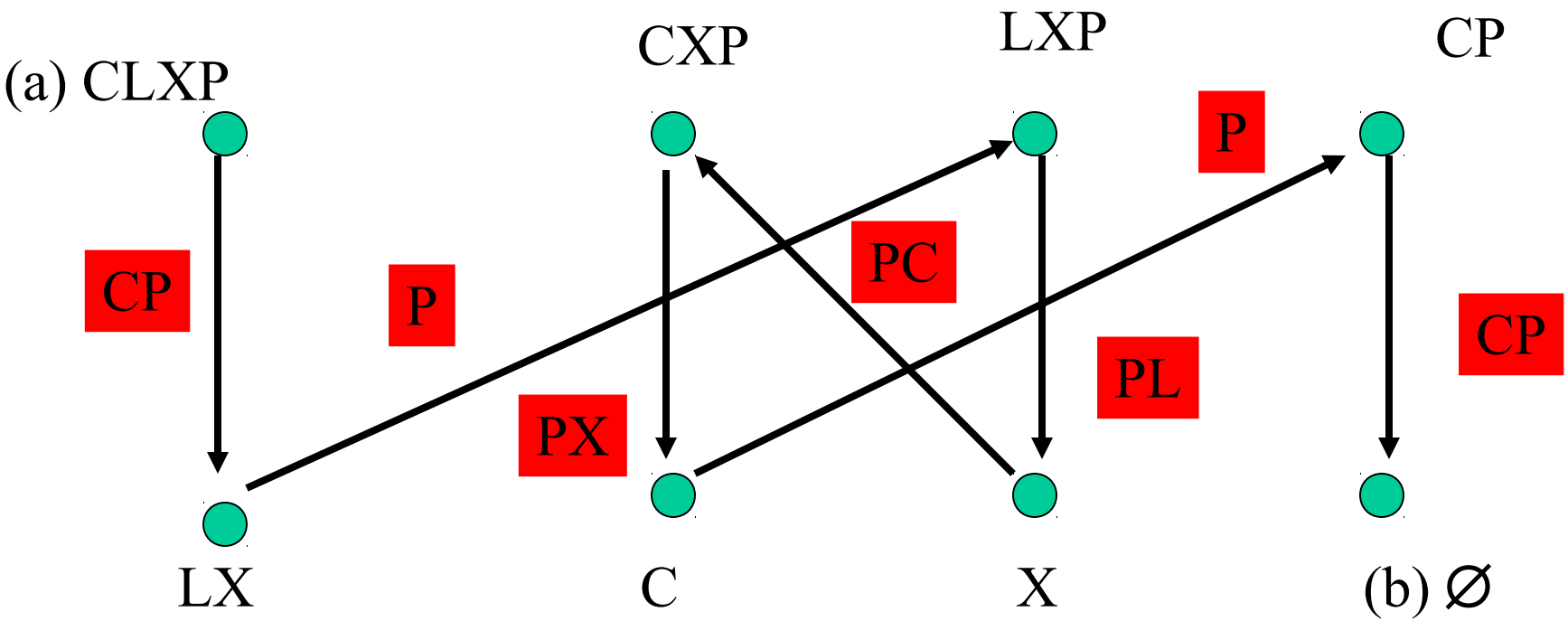
- Le chemin sert à modéliser la solution à apporter au problème. Il apporte alors la solution du problème.
- Le problème est l'évaluation du chemin (réseau de transport). La modélisation du graphe est une donnée.

Modélisation

- Exemple : le loup (L), la chèvre (C), le chou (X) et le passeur (P)
  - Passage d'une rive à l'autre d'une rivière sous contraintes
    - La chèvre mange le chou
    - Le loup mange la chèvre
    - Que deux dans la barque

Solution

- Passage d'un état (a) : tout le monde se trouve sur la rive droite à un état (b) : plus rien n'est sur la rive droite sous la forme suivante :



Problème évaluation

- Etant donné un graphe G, associons à chaque arc u un nombre $l(u) \geq 0$ que nous appelons la « longueur » de l'arc u.
- Problème : Trouver le coût d'un chemin élémentaire, c, allant d'un noeud « a » à un noeud « b » et tel que la longueur totale

  $l_c = \sum l(u)$ pour u appartenant à c

  soit aussi petite que possible

Objectifs

- Les noeuds sont numérotés 1, 2, ... n
- Les arcs sont valués (problème base de données)
- Trouver un chemin critique lors de planification ou Trouver un chemin entre les noeuds n°1 et n°k de coût minimum (→ (1, k, Min(f))) : abus de langage : « plus court chemin »
- Hypothèses : pas de circuit dans les chemins solution (élémentaires)

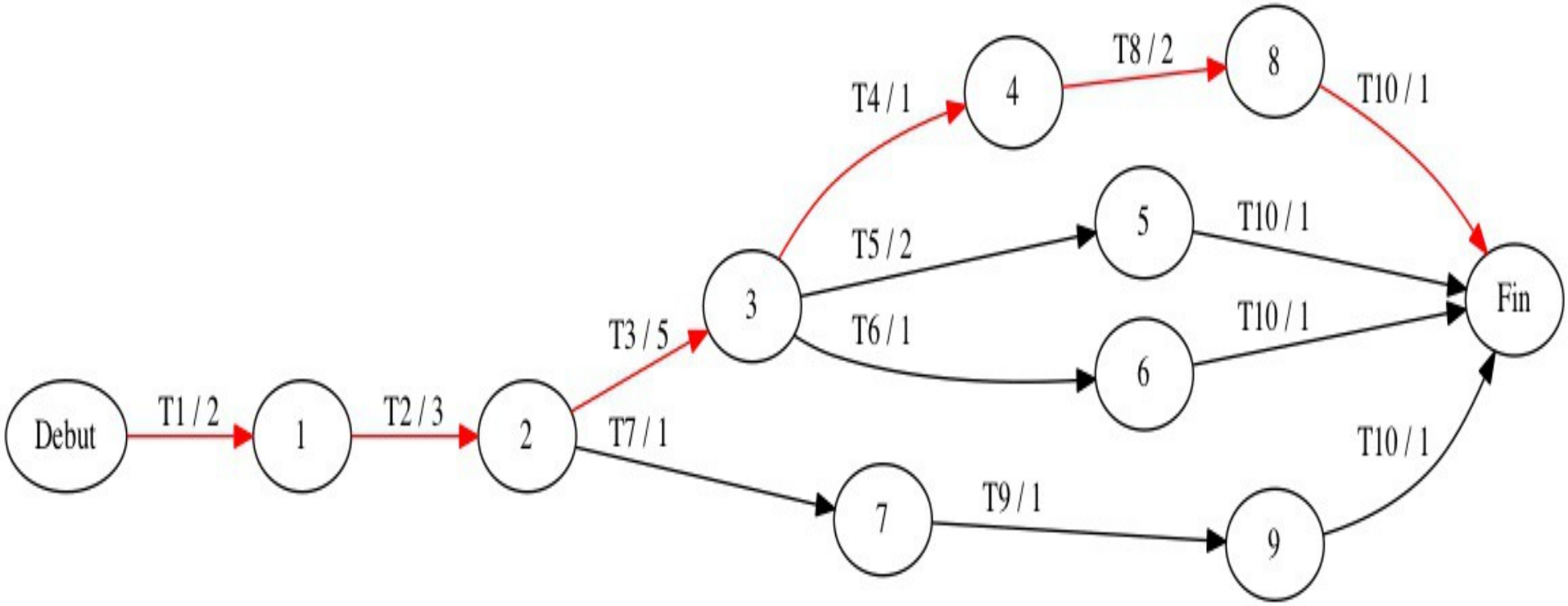
Ordonnancement

- L’ordonnancement est l’élaboration d’un plan d’action permettant de déterminer le séquencement ou le parallélisme possible entre l’exécution de tâches définies (WBS).
- Estimer la durée des tâches
- Dates au plus tôt, dates au plus tard,
- A surveiller comme le lait sur le feu.

Chemin critique (www.qualite.qc.ca)

- Planification: construire une maison
  - T1 : Analyse des besoins (2 semaines)
  - T2 : Concevoir les plans (3 semaines)
  - T3 : Gros oeuvre (5 semaines)
  - T4 : Electricité (2 semaines)
  - T5 : Huisserie (1 semaine)
  - T6 : Carrelage (1 semaine)
  - T7 : Commander les meubles (2 semaines)
  - T8 : Peinture (2 semaines)
  - T9 : Livraison des meubles (1 semaine)
  - T10 : Finitions (1 semaine)
- Modélisation :
  - Noeuds : début tâche (début/tâches/fin)
  - Arc : tâches / durée

Chemin critique



Analyse

- Chemin critique : Séquence de tâches. Détermine la durée totale
  - continu du début à la fin du projet : Tout retard affectant une tâche du chemin critique (tâche critique) est intégralement répercuté sur la durée du projet et change sa date de fin.
- Marge : possibilité qu'à une tâche d'être retardée ou de prendre du retard sans son exécution sans impacter le projet.
  - Les tâches critiques ont une marge nulle.
- Marge totale : égale à la différence entre le début au plus tard de la tâche suivante la plus contraignante et la fin au plus tôt de la tâche elle-même. C'est aussi la différence entre les dates au plus tard et les dates au plus tôt de la tâche elle même.
- Marge libre : égale à la différence entre la date de début au plus tôt du successeur le plus précoce et la date de fin au plus tôt de la tâche elle-même.

Calcul de chemins

Un exemple : Algorithme de Dijkstra

Démarche

- Système de potentiels (μ).
- Séquence de noeuds visités, S
- Le potentiel d'un noeud, $\mu(i)$, **dans S** est égal au coût minimal d'un chemin de 1 à i.
- Le potentiel d'un noeud **hors S** est
  - soit indéterminé (valeur infinie)
  - soit de potentiel déterminé, mais susceptible d'être amélioré.

Pré-conditions

- Les coûts des arcs doivent être positifs ou nuls.
- Cet algorithme ne s'applique pas pour la recherche du chemin maximal
- Le fonctionnement de l'algorithme est indépendant de la numérotation des noeuds.
- On suppose ici qu’il existe un chemin de 1 à k
- Cet algorithme donne le <u>coût minimal</u> d'un chemin de 1 à k.

Algorithme (Principe)

- $\mu(1) = 0$
- $\forall\ i \in \Gamma+(1)$, $\mu(i)$ est le coût de l'arc (1,i)
- $\forall\ j \notin \Gamma+(1)$, $\mu(j) = \infty$
- S = <1> .
- A chaque étape, détermination
  - Du(es) noeud j qui doit entrer dans S ;
  - Mise à jour des potentiels de ses successeurs.
- Itération du processus jusqu'à ce que le noeud k appartienne à S.

Mise à jour de μ(j)

- Si i est sélectionné pour entrer dans S

Alors

pour chacun de ses successeurs j

si le fait d'aller de 1 à j en passant par i
diminue le coût du chemin
($\mu(i)$ + Coût (i,j) < $\mu(j)$)

Alors

$\mu(j) = \mu(i)$ + Coût (i,j)

finsi

fpour

finsi

Algorithme

$\mu(1) = 0$; $S = <1>$

Pour tout noeud $i \in \Gamma+(1)$, $\mu(i) = \text{coût}(1,i)$

Pour tout noeud $i \notin \Gamma+(1)$, $\mu(i) = \infty$

Tant que $k \notin S$, faire

Pour tout noeud $i \notin S$ / $\mu(i) = \text{Min}(\mu(j), j \notin S)$

$S = S \cup <i>$

Pour tout $j \in \Gamma+(i)$,

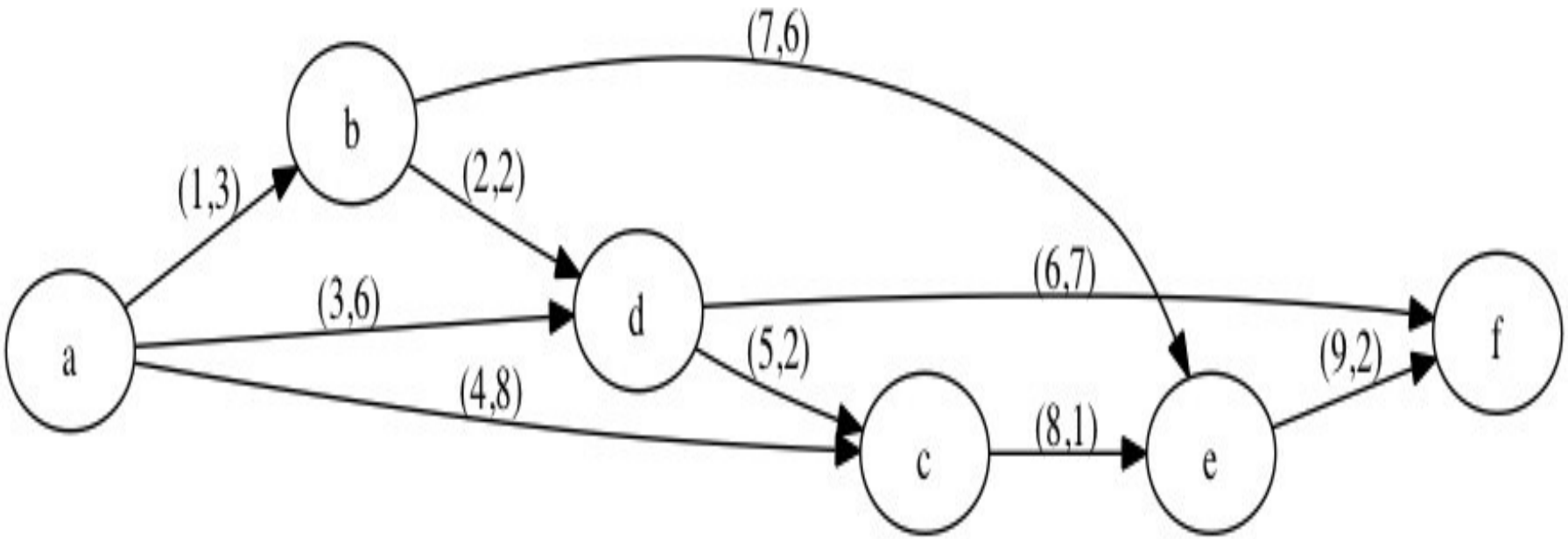
$\mu(j) = \text{Min}(\mu(i)+\text{Coût}(i,j), \mu(j))$

Fpour

Fpour

Ftq

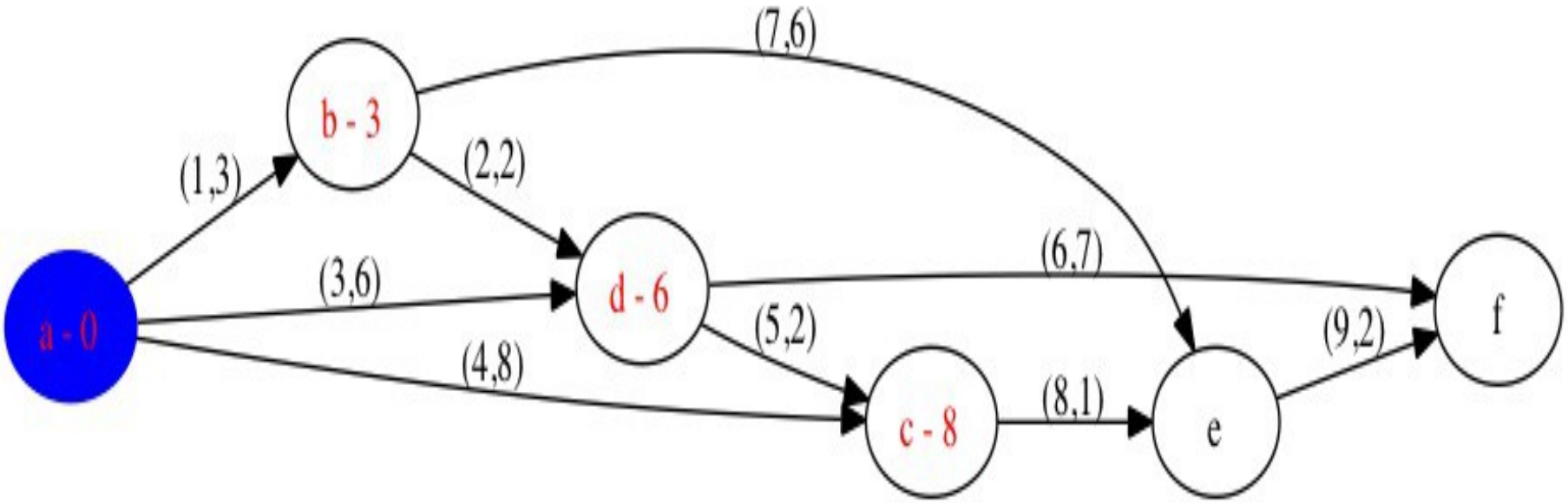
Exemple



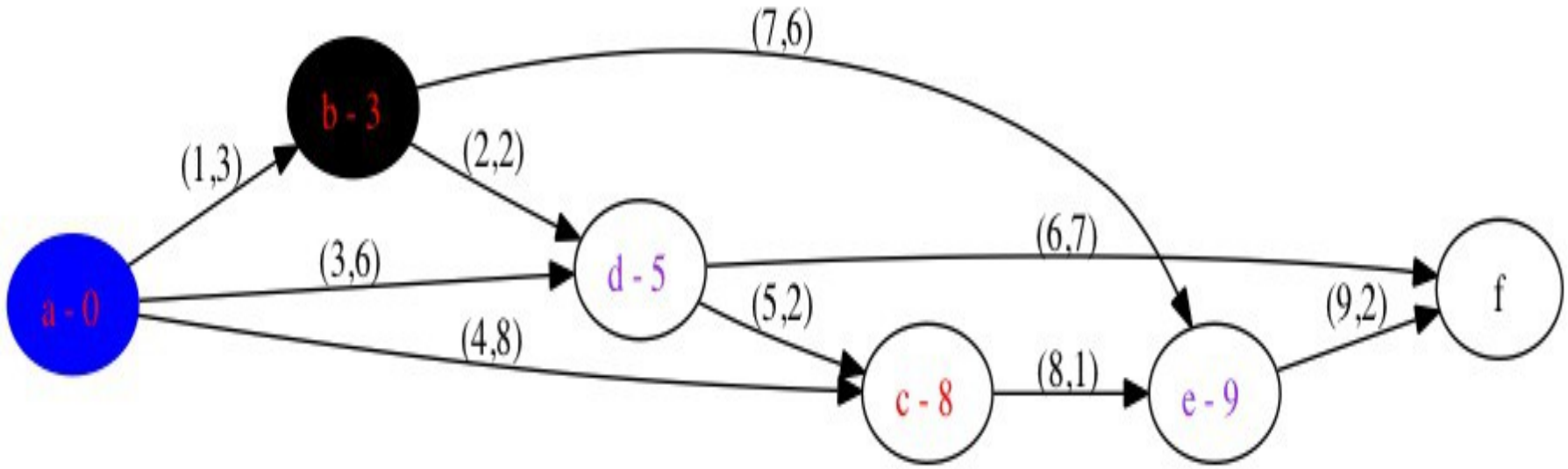
Evolution

- Potentiel en rouge
- En bleu les noeuds appartenant à S
- En noir les noeuds qui vont entrer dans S, (ils vont donc éventuellement modifier le potentiel de leurs successeurs)
- Les mises à jour de potentiels sont signalées en violet

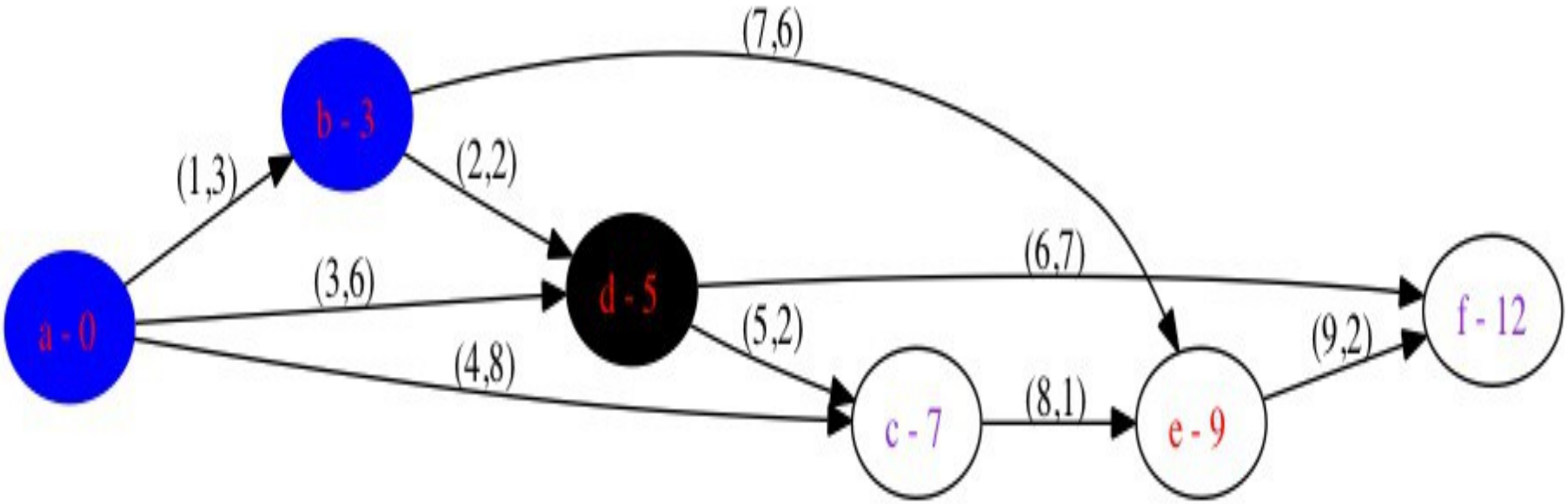
Initialisation



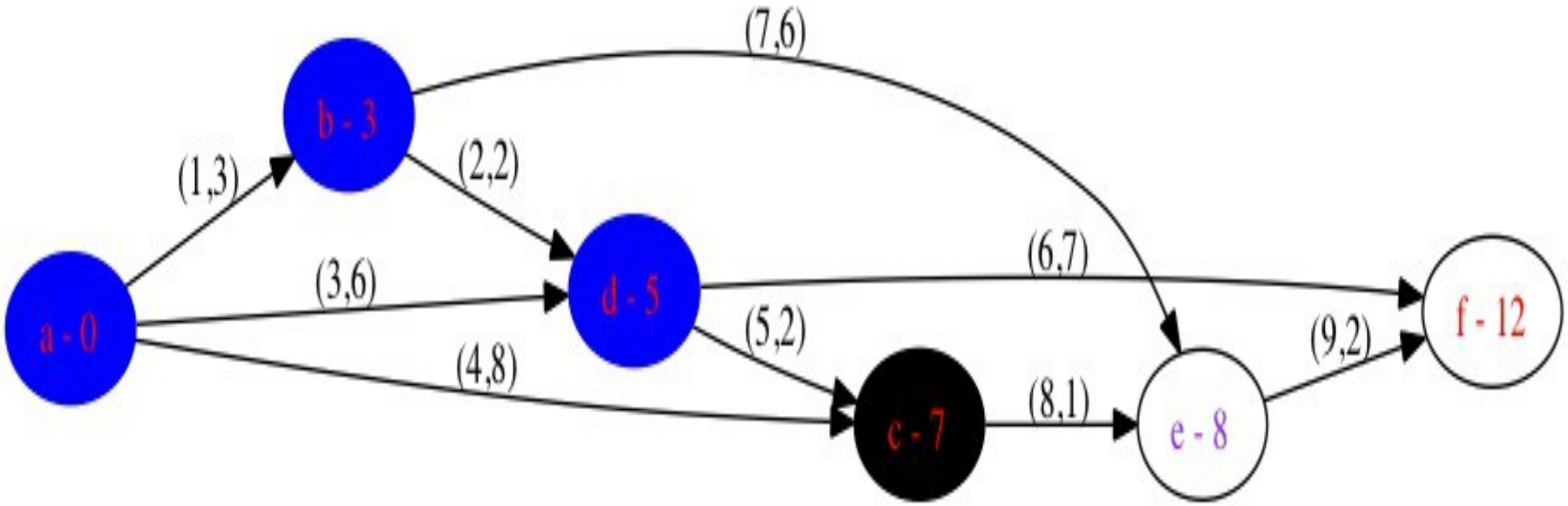
Première étape



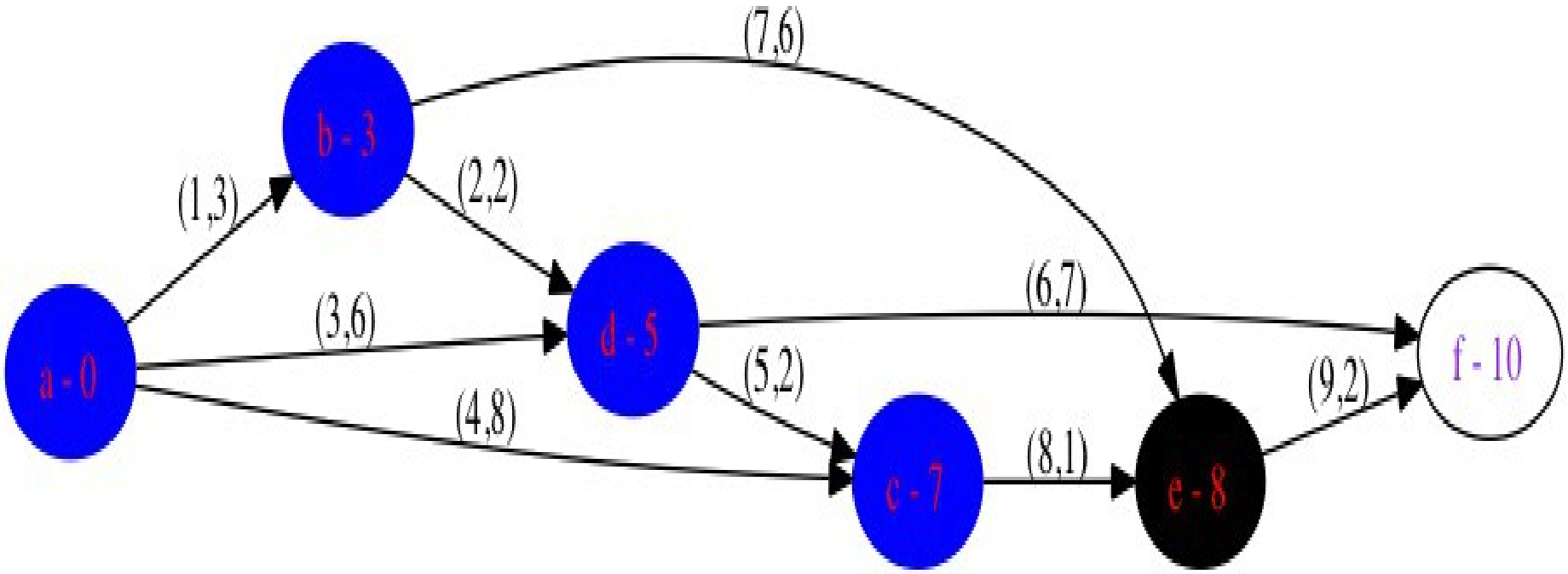
Deuxième étape



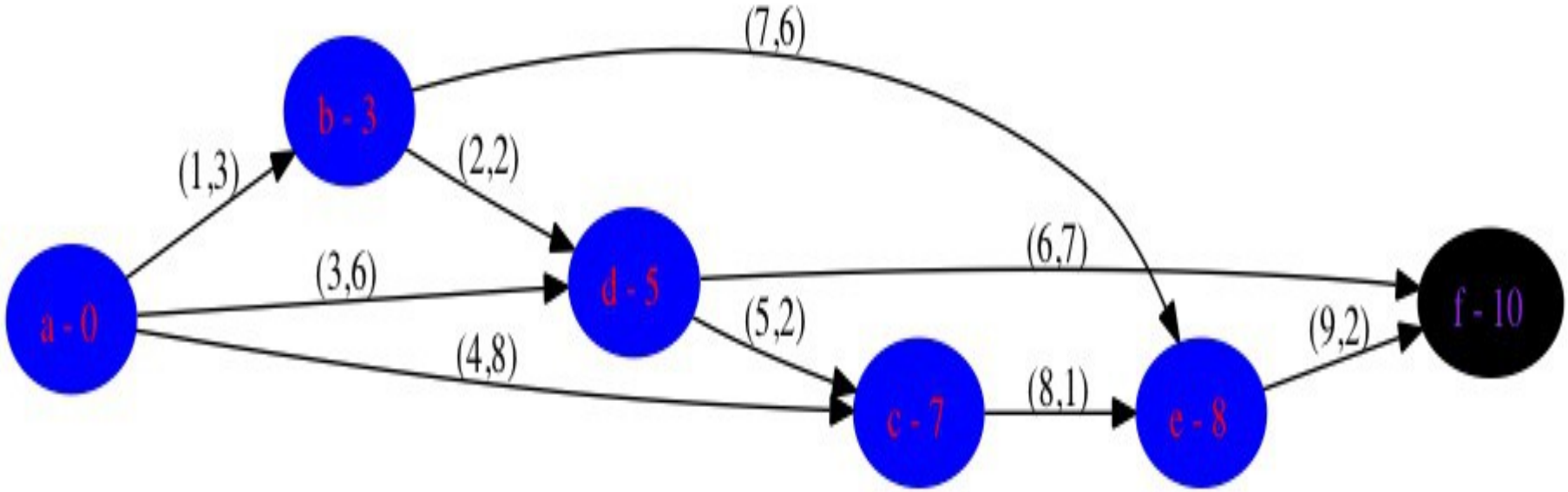
Troisième étape



Quatrième étape



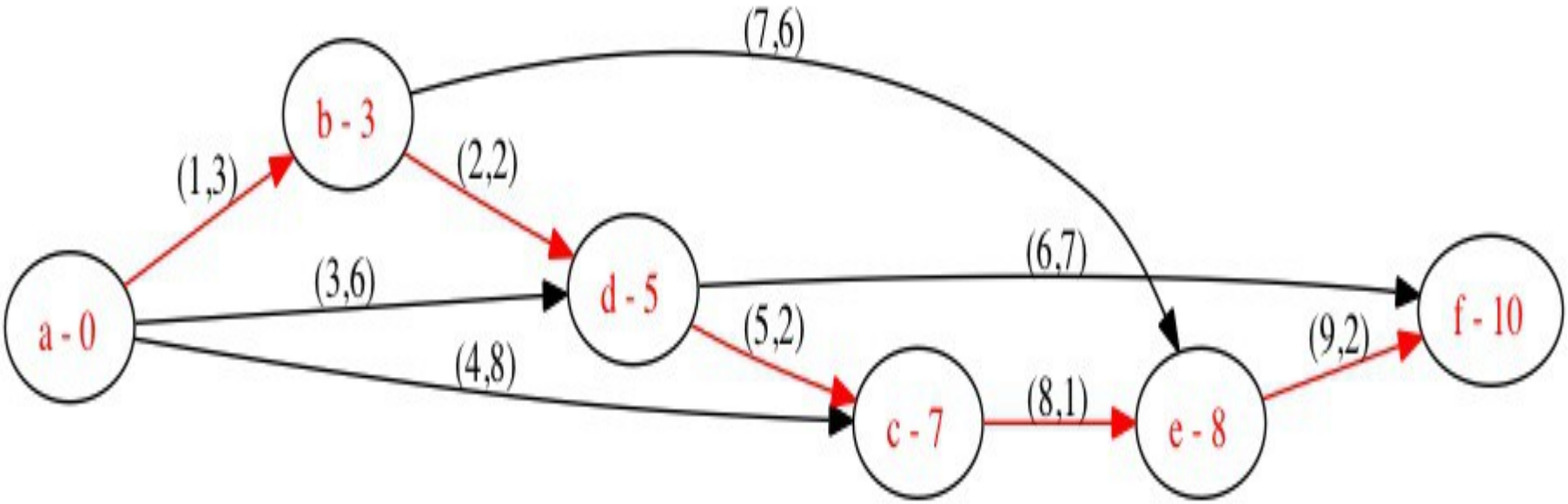
Cinquième et dernière étape



Retrouver le chemin

- Pour retrouver le chemin minimal, il faut que chaque noeud « se souvienne » du prédécesseur qui lui a donné son potentiel optimal.
- Utilisation de la récursion dans le sens opposé des arcs.
- Sur notre exemple, le chemin minimum est (a-b-d-c-e-f) . Pour le trouver, on « remonte » la liste des prédécesseurs « privilégiés ».

Exemple



Autres algorithmes

- Dantzig
- Ford
- Algorithme naïf et semi-naïf
- Bellman (pas de circuit)
- Bellman-Kalaba
- Méthodes matricielles (Warshall, …)
- …

Rôle de la fonction d'étiquetage

- Apporter une sémantique plus importante
  - Sur les nœuds
  - Sur les arcs
  - Sur les graphes
- Restreindre les procédures de recherche
- … augmenter la complexité des algorithmes de recherche
  - Lien entre graphes et automates (expressions régulières)
  - Noeuds transparents (fonction agrégats)
  - Introduction de l'ordre

Requête mettant en jeu des opérateurs de graphes

- Requête élémentaire
  - Donner le graphe du métro
  - Quels sont les arcs du graphe vérifiant une propriété donnée (e.g., compagnie Air France)
  - Quels sont les nœuds du graphe vérifiant une propriété donnée (e.g., ville de plus de 100000 habitants)
    - Equivalent à une sélection/projection relationnelle

Requête mettant en jeu des opérateurs de graphes (2)

- Requête complexe
  - Evaluation d'un chemin de A à B :
    - Sous contraintes agrégatives
    - Sous expression régulière
  - Composition d'opérateurs :
    - Autres chemins
    - Intersection
    - Inclusion
- Equivalent à un calcul de fermeture transitive

Exemple (1) – Expression régulière

- Question : Evaluer un chemin entre deux villes A et B en prenant d'abord une nationale puis l'autoroute puis de nouveau une nationale :
- Modélisation :

  → (A, B, (nationale)+ (Autoroute)+ (Nationale)+)

Exemple (2) – Noeuds transparents

- Question : Evaluer un chemin entre deux villes A et B en minimisant le coût d'hôtel et d'autoroute :
- Modélisation :

  Attention la sommation n'est pas effectuée pour tous les nœuds mais seulement une partie en fonction de l'heure d'arrivée et de départ (i.e., des arcs incidents).

Exemple (3) - Ordre

- Question : Evaluer un chemin entre Paris et Nice en passant d'abord par Grenoble puis par Lyon
- Modélisation :

  Attention ce n'est pas la simple juxtaposition des chemins Paris-Grenoble, Grenoble-Lyon, Lyon-Nice (à cause des circuits)

Résultat(s) de l'opérateur de chemin

- Dans le cas réel, le résultat n'est pas un chemin mais un ensemble de chemins.
- Le résultat final n'est pas l'union des chemins (notamment dès qu'une fonction d'agrégat est en jeu)
  - Problème de signature des opérateurs
  - Restitution des résultats
- Gestion de la composition d'opérateurs

Distance (hors sémantique)

- Distance entre deux noeuds :
  - Longueur du plus court chemin/chaîne entre ces noeuds (distincts)
- Diamètre d'un graphe :
  - La plus longue des distances entre deux noeuds quelconques, mais distincts, du graphe
- Excentricité d'un sommet :
  - La plus longue des distances à partir de ce sommet
- Centre d'un graphe
  - noeud(s) d'excentricité minimale (appelée rayon)

Distance (2)

- Cycle :
  - Longueur : nombre de arêtes (ou noeuds)
  - g(G) : longueur du cycle minimal
  - Périmètre de G : longueur maximale d'un cycle
- Acyclique :
  - g(G) = 0
  - Périmètre infini