

Optimisation Combinatoire pour la conception de circuits

*Plus courts chemins et application à la
compaction*

Alix Munier Kordon

Alix.Munier@lip6.fr

<http://www-asim.lip6.fr/~alix/>

Laboratoire Lip6

Université Paris 6/ Paris 12

Plan

- Définition et complexité
- Algorithme de Bellman-Ford
- Algorithme de Dijkstra
- Compaction

Définition d'un plus court chemin

Instance: $G = (V, E)$ un graphe orienté,
 $w : V \rightarrow \mathbb{Z}$, un entier K et un couple $(s, p) \in V^2$.

Question: peut on trouver un chemin
élémentaire ν de s à p tel que $\sum_{e \in \nu} w(e) \leq K$?

Circuit Hamiltonien

Instance: $H = (W, A)$ un graphe orienté.

Question: est ce que H contient un circuit qui passe exactement une fois par chaque sommet ?

Circuit Hamiltonien est NP-complet
(Garey-Johnson)

Complexité

Theorème 1. Plus court chemin *est NP-complet.*

On montre que

Circuit hamiltonien α Plus court chemin

Hypothèse simplificatrice

Les algorithmes de chemins (polynomiaux) ne fonctionnent au minimum que pour des graphes dont tous les circuits c vérifient

$$\sum_{e \in c} w(e) \geq 0$$

Chemins à origine unique

La valeur d'un chemin ν est $w(\nu) = \sum_{e \in \nu} w(e)$.

Pour tout couple $(u, v) \in V^2$,
 $\delta(u, v)$ = valeur
d'un plus court chemin de u à v .

Soit $s \in V$. On cherche à calculer, pour tout $u \in V$,
 $\delta(s, u)$.

Ajustement

Pour tout $u \in V$, soit $d_u \geq \delta(s, u)$

procédure ajustement($d, (u, v)$)

Si $d_u + w(u, v) < d_v$ **alors**

$$d_v = d_u + w(u, v);$$

Lemme 2. *Si, pour tout $u \in V$, $d_u \geq \delta(s, u)$, alors la procédure d'ajustement construit un vecteur d'_u tel que $d_u \geq d'_u \geq \delta(s, u)$.*

Algorithme de Bellmann-Ford

```
fonction Bellman-Ford( $G = (V, E), w, s, d$ )  
   $d_s = 0$ ;  
  Pour tout  $u \in V - \{s\}, d_u = +\infty$ ;  
  Pour  $i = 1$  a  $|V| - 1$  faire  
    Pour tout  $(u, v) \in E$  faire  
      ajustement( $d, (u, v)$ );  
  Pour tout  $(u, v) \in E$  faire  
    Si  $d_v > d_u + w(u, v)$  alors  
      retourner faux;  
  retourner vrai
```

Validité et complexité de Bellman-Ford

Theorème 3. *Si Bellman-Ford est vrai, le graphe G ne contient pas de circuit de valeur strictement négative.*

Theorème 4. *Si G ne contient pas de circuit de valeur strictement négative, alors les valeurs $d_u, u \in V$ obtenues par Bellman-Ford vérifient $d_u = \delta(s, u)$*

Complexité: $\mathcal{O}(|V||E|) = \mathcal{O}(|V|^3)$

Arborescence des plus courts chemins

On suppose que G est sans circuit de valeur négative. Soit le sous graphe partiel $G_\pi = (V_\pi, E_\pi)$ de G définie par:

- V_π est l'ensemble des sommets u de V tels que il existe un chemin de s à u .
- E_π est un ensemble d'arêtes de E tel que tout sommet $u \in V_\pi - \{s\}$ n'a qu'un seul prédécesseur v dans G_π et que $d_u = d_v + w(u, v)$.

Arborescence des plus courts chemins

Theorème 5. $G_\pi = (V_\pi, E_\pi)$ est une arborescence de racine s . De plus, $\forall u \in V_\pi$, la valeur du chemin de s à u est $\delta(s, u)$.

Algorithme de Dijkstra

Applicable pour des valuations positives ou nulles. $\text{Extraire-min}(F, d)$ renvoie l'élément $u \in F$ de valeur d_u minimale et supprime u de F .

```
procedure Dijkstra( $G = (V, E), w, s, d$ )  
   $d_s = 0$ ; Pour tout  $u \in V - \{s\}, d_u = +\infty$   
   $F = V$ ;  
  tant que  $F \neq \emptyset$  faire  
     $u = \text{Extraire-min}(F, d)$ ;  
    Pour tout  $v \in \Gamma^+(u)$  faire  
       $\text{ajustement}(d, (u, v))$ ;
```

Validité de l'algorithme de Dijkstra

$$\mathcal{B}(V - F) = \{v \in F / \Gamma^-(v) \cap (V - F) \neq \emptyset\},$$

Lemme 6. *Pour tout $u \in \mathcal{B}(V - F)$*

$$d_u = \min\{d_v + w(v, u), v \in V - F\}$$

Theorème 7. *Soit $u^* \in \mathcal{B}(V - F)$ de valeur d_{u^*} minimale dans $\mathcal{B}(V - F)$. Alors $\delta(s, u^*) = d_{u^*}$.*

Complexité de l'algorithme de Dijkstra

En utilisant pour F un tas:

- Extraire-min est de complexité $\mathcal{O}(\log |F|)$
- Quand une valeur est ajustée, il faut modifier le tas: $\mathcal{O}(\log |F|)$,

En tout, $\mathcal{O}(|E| \log(|S|))$.

Compaction

En entrée: les composants sont des rectangles $R_i^* = (x_i^*, y_i^*, m_i^*, l_i^*, h_i^*)$

- de taille fixée (transistors),
- ou déformables (fils).

(x_i^*, y_i^*) : coordonnées du bord inférieur gauche,
 $m_i^* \in \{1, \dots, m\}$ couche correspondante,
 (l_i^*, h_i^*) longueur et hauteur du rectangle.

Compaction

Définition 8. *Deux rectangles R_i et R_j sont compatibles si ils peuvent être mis l'un sur l'autre sans aucune restriction. On note alors $R_i \pi R_j$.*

Ce peut-être le cas de composants sur des couches différentes sans interaction, ou de fils au même potentiel.

Compaction

Définition 9. Deux rectangles R_i^* et R_j^* incompatibles se recouvrent horizontalement ssi:

- $y_j^* < y_i^* + h_i$ et $y_i^* < y_j^* + h_j$,
- $x_j^* < x_i^*$ ou ($x_j^* = x_i^*$ et $j < i$)

On note alors $R_j^* < R_i^*$

Compaction en une dimension

On cherche ici à diminuer les valeurs de coordonnées x_i en jouant sur la deformation des fils horizontaux de sorte que:

- La relation $<$ ne soit pas modifiée,
- les contraintes de *groupement* et *d'espacement* soient vérifiées,
- et la longueur du circuit soit minimale.

Soient $R_i = (x_i, y_i, m_i, l_i, h_i)$ l'ensemble des rectangles obtenues.

Déformation des fils

- Si R_i^* est un fil horizontal, alors il peut être réduit de sorte que:

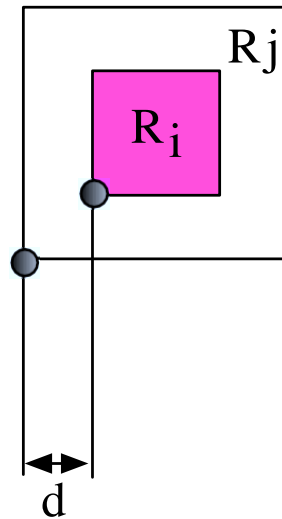
$$l_i \geq l_i^{\min}$$

- Tous les autres rectangles sont indéformables

$$l_i = l_i^{\min} = l_i^*$$

Contraintes de groupement

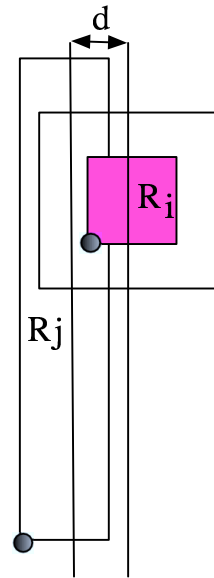
Si R_i^* et R_j^* sont deux rectangles qui contribuent au même élément du circuit, ils peuvent être astreint à être l'un sur l'autre:



$$x_i + d = x_j$$

Contraintes de groupement

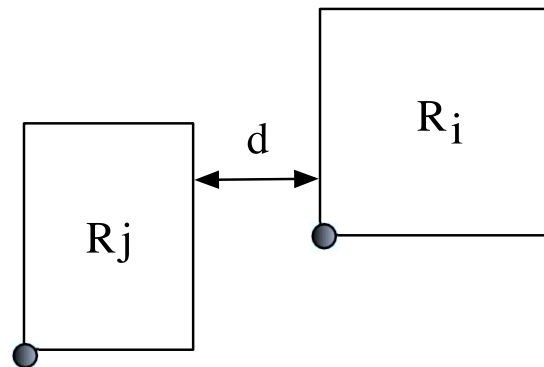
La connexion entre un fil et un transistor doit avoir une largeur d'au moins d :



$$\left| x_i + \frac{l_i^*}{2} - \left(x_j + \frac{l_j^*}{2} \right) \right| \leq d$$

Contrainte d'espacement

Deux rectangles (par exemple sur une même couche) ne peuvent s'intersecter:



$$\text{Si } R_j < R_i, x_j + l_j^* + d \leq x_i.$$

Graphe de contraintes

$G=(V,E)$ est définie tel que:

- Un sommet par rectangle,
- A toute contrainte $x_j - x_i \geq d_{ij}$, un arc $e = (i, j)$ de valuation $w(e) = d_{ij}$.

La solution minimale de ce système est calculable par Bellman-Ford ($O(n^3)$).

Heuristique

On compacte le circuit tour à tour selon chaque dimension jusqu'à ce que il n'y ai plus de modification.