O problema do trajecto mais curto segundo vários critérios

Marta Pascoal†
Deolinda Rasteiro†
José Luis Santos†
{marta, dml, zeluis}@mat.uc.pt

Centro de Informática e Sistemas da Universidade de Coimbra
Departamento de Matemática
Faculdade de Ciências e Tecnologia
Universidade de Coimbra

braga, abril 1999

Os autores pertencem à Linha *Ciências da Computação* do Centro de Informática e Sistemas da Universidade de Coimbra, parcialmente subsidiado pelo Ministério das Ciências e Tecnologia através do Projecto PRAXIS XXI da Junta Nacional de Investigação Científica e Tecnológica.

http://www.mat.uc.pt/~zeluis/INVESTIGACAO/mestrado_revisto.ps.gz
http://www.mat.uc.pt/~eqvm/CISUC/cisuc.html

†— Sob a orientação do Professor Doutor Ernesto de Queirós Vieira Martins.

Rede Rodoviária

- **Nó**: local ou cidade.

$$\mathcal{N} = \{v_1, \ldots, v_n\},$$

(ou simplesmente $\mathcal{N} = \{1, \ldots, n\}$)

s = local de partida; t = local de chegada.

- **Arco**: estrada (orientada) ligando dois locais.

$$\mathcal{A} = \{(i, j) : i, j \in \mathcal{N}\}.$$

Exemplo:

  - sentidos únicos.
  - custos diferentes para sentidos opostos.

- **Custo**: parâmetros associados a cada arco,

$$c^1_{i,j}, \ldots, c^k_{i,j} \in \mathbb{R}.$$

(tempo, custo, distância, ...)

- **Trajecto**: percurso realizado entre dois locais.

- **Função objectivo**: define o custo de cada trajecto.

Notação

Nós:

$$\mathcal{N} = \{1, \ldots, n\}$$
$$n = \text{número de nós}$$
$$s = \text{nó inicial}$$
$$t = \text{nó terminal}$$

Arcos (orientados):

$$\mathcal{A} = \{(i, j) : i, j \in \mathcal{N}\}$$
$$m = \text{número de arcos}$$

Trajecto: sequência de nós e de arcos percorridos no sentido correcto.
$p = \langle i = n_0, e_1, n_1, \ldots, n_{h-1}, e_h, n_h = j \rangle$,
onde $n_\ell \in \mathcal{N}, \forall \ell = 0, \ldots, h$
e $e_\ell = (n_{\ell-1}, n_\ell) \in \mathcal{A}, \forall \ell = 1, \ldots, h$.

Caminho: trajecto sem nós repetidos.

$\mathcal{P}_{i,j}$: conjunto dos trajectos de i para j
$\mathcal{P} = \mathcal{P}_{s,t}$
$\mathcal{P}_{\mathcal{G}} = \cup_{i,j \in \mathcal{N}} \mathcal{P}_{i,j}$

Problema do Trajecto Mais Curto Multiobjectivo

Dados

- $\mathcal{G} = (\mathcal{N}, \mathcal{A})$
- $\mathbf{c_{i,j}} = (c_{i,j}^1, \ldots, c_{i,j}^k) \in \mathbb{R}^k, \forall (i,j) \in \mathcal{A}$
- $$\begin{array}{rcl} f: \mathcal{P}_{\mathcal{G}} & \longrightarrow & \mathbb{R}^k \\ p & \longmapsto & f(p) = (f_1(p), \ldots, f_k(p)) \end{array}$$
  onde
  $$\forall \ell = 1, \ldots, k; \begin{array}{rcl} f_\ell: \mathcal{P}_{\mathcal{G}} & \longrightarrow & \mathbb{R} \\ p & \longmapsto & f_\ell(p) = \Sigma_p c_{i,j}^\ell. \end{array}$$
- $\mathcal{P}^\star = \mathcal{P}$

Pretende-se:

Determinar $p^* \in \mathcal{P}$ tal que

$$f(p^*) = \min\{f(q) : q \in \mathcal{P}\},$$

isto é

$$\forall i = 1, \ldots, n, f_i(p^*) = \min\{f_i(q) : q \in \mathcal{P}\}.$$

NOTA:

Se existir conflito entre os vários objectivos,
então **não** existe solução óptima.

Relações de Ordem

Em $\mathbb{R}$:

- "$\leq$": relação de ordem TOTAL em $\mathbb{R}$.

  (permite relacionar o custo de dois quaisquer trajectos)

- "$\leq$": relação de pré-ordem TOTAL em $\mathcal{P}$.

$$p \leq q \Leftrightarrow f(p) \leq f(q).$$

Em $\mathbb{R}^k$:

- "$\preceq_D$": relação de ordem PARCIAL em $\mathbb{R}^k$.

  $\mathbf{a} = (a_1, \ldots, a_k), \mathbf{b} = (b_1, \ldots, b_k) \in \mathbb{R}^k$

$$\mathbf{a} \preceq_D \mathbf{b} \Leftrightarrow a_i \leq b_i, \forall i = 1, \ldots, k.$$

- "$\preceq_D$": relação de pré-ordem PARCIAL em $\mathcal{P}$.

$$p \preceq_D q \Leftrightarrow f(p) \preceq_D f(q).$$

p é melhor do que q

Em $\mathbb{R}$:

$$\begin{aligned} p < q &\Leftrightarrow f(p) < f(q) \\ &\Leftrightarrow f(p) \leq f(q) \text{ e } f(p) \neq f(q) \end{aligned}$$

Em $\mathbb{R}^k$:

- Relação de dominância

$$\begin{aligned} p \prec_D q &\Leftrightarrow f(p) \preceq_D f(q) \text{ e } f(p) \neq f(q) \\ &\Leftrightarrow \forall i \in \{1, \ldots, k\}, f_i(p) \leq f_i(q) \text{ e} \\ &\quad\;\; \exists j \in \{1, \ldots, k\} : f_j(p) < f_j(q) \end{aligned}$$

- "$\prec_D$": relação PARCIAL ESTRITA.
- p domina q (ou q é dominado por p)
- $p \in \mathcal{P}$ é não dominado $\Leftrightarrow \nexists\, q \in \mathcal{P} : q \prec_D p$
- $D_p = \{q \in \mathcal{P} : p \prec_D q\}$
  $D = \cup_{p \in \mathcal{P}} D_p$
  $\bar{D} = \mathcal{P} - D$

p^* é trajecto óptimo

Em $\mathbb{R}$:

$$f(p^*) = \min_{p \in \mathcal{P}}\{f(p)\} \;\Leftrightarrow\; f(p^*) \leq f(q), \forall q \in \mathcal{P}$$
$$\Leftrightarrow\; \nexists\, q \in \mathcal{P} : f(q) < f(p^*)$$

Em $\mathbb{R}^k$:

$$f(p^*) = \min_{p \in \mathcal{P}}\{f(p)\} \;\Leftrightarrow\; \nexists\, q \in \mathcal{P} : f(q) \prec_D f(p^*)$$
$$\Leftrightarrow\; p^* \in \bar{D}$$

OBSERVAÇÃO:

"$\preceq_D$" é uma relação parcial

$\Downarrow$

podem existir vários mínimos
e vários minimizantes de f

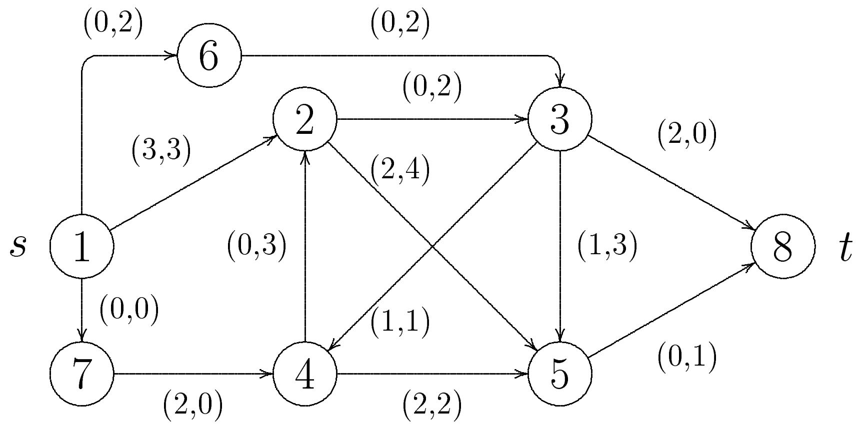
Algoritmos para o Problema do Trajecto Mais Curto

Algoritmo de Pesquisa Exaustiva

- Constrói uma árvore com "todas" as possíveis soluções.
- Escolhe as soluções óptimas.

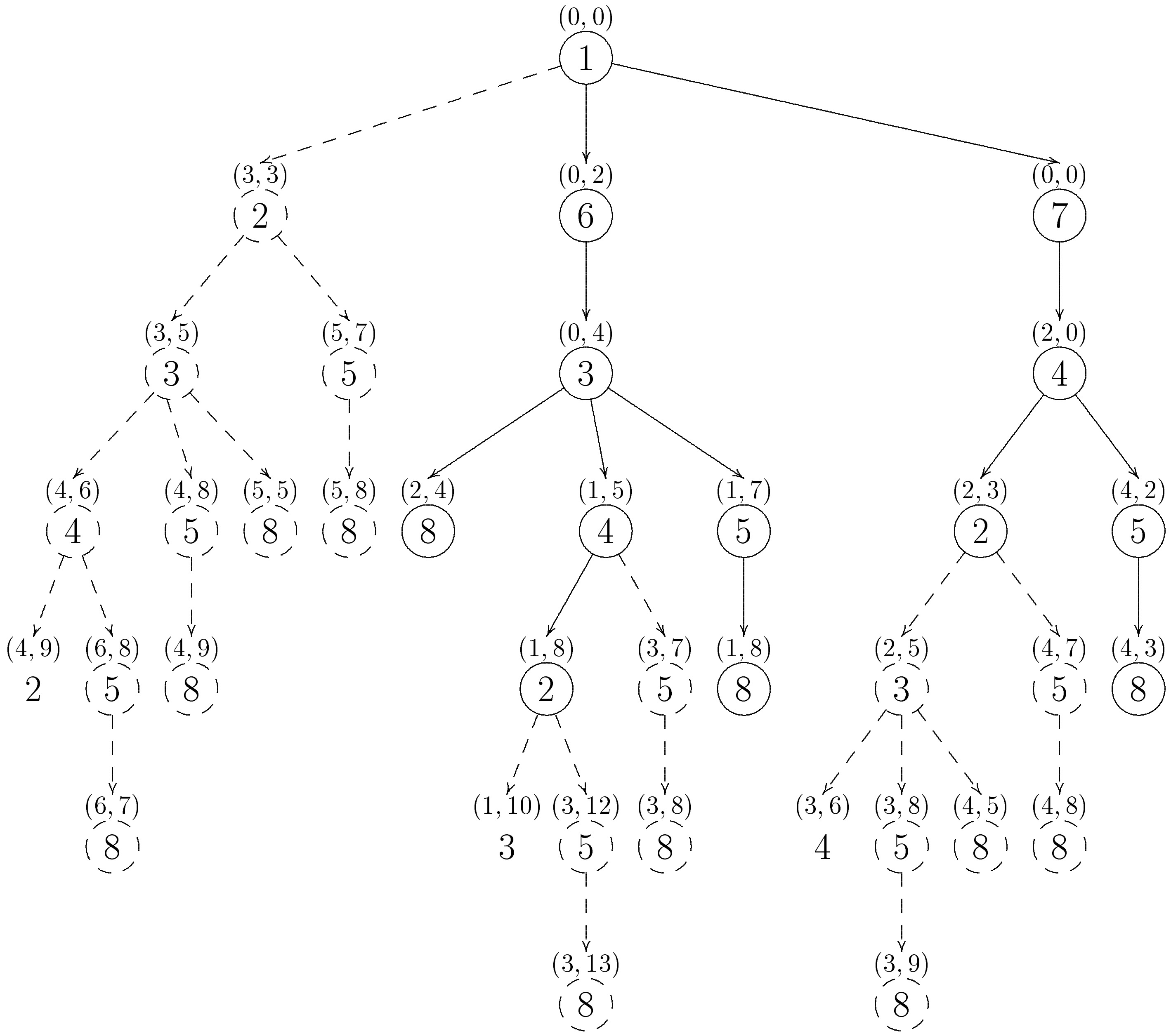
Exemplo para o problema do CAMINHO mais curto em $\mathbb{R}^2$.

Nota:
Não se pode aplicar ao problema do trajecto óptimo
(não é um processo finito).



Rede

(c_k^1, c_k^2) i $\qquad i$ – nó de $\mathcal{G}$; $\quad (c_k^1, c_k^2)$ – custo de s até i



Árvore de pesquisa exaustiva dos caminhos

Princípio de Optimalidade

- Princípio de Optimalidade Forte:
  TODO o trajecto óptimo é constituído por subtrajectos óptimos.

- Princípio de Optimalidade Fraca:
  - Em $\mathbb{R}$:
    EXISTE pelo menos um trajecto óptimo constituído por subtrajectos óptimos.
  - Em $\mathbb{R}^k$:
    $\forall \mathbf{v} \in f(\bar{D})$, EXISTE pelo menos um trajecto óptimo (não dominado) com custo $\mathbf{v}$ constituído por subtrajectos não dominados.

O Problema do Trajecto Mais Curto

Verifica o Princípio de Optimalidade Forte (e Fraca)

Em $\mathbb{R}$:
SSE não existem ciclos negativos.

Em $\mathbb{R}^k$:
SE não existem ciclos negativos.

Algoritmo de Rotulação para o problema do trajecto mais curto em $\mathbb{R}^k$

Princípio de Optimalidade Fraca $\Longrightarrow$ Determinação de $f(\bar{D})$
Princípio de Optimalidade Forte $\Longrightarrow$ Determinação de $\bar{D}$

$\{X_j$: `conjunto de rótulos, ainda não testados, associados ao nó` $j\}$
$\{X$: `conjunto de rótulos ainda não testados, isto é,` $X = \bigcup_{j\in\mathcal{N}} X_j\}$
$\{Y_i$: `subconjunto de` $X_i\}$
$\{\Pi_j$: `conjunto dos rótulos não dominados associados ao nó` $j\}$

$X_s \leftarrow \{\mathbf{0}_{\mathbb{R}^k}\}$
$\Pi_s \leftarrow \{\mathbf{0}_{\mathbb{R}^k}\}$
$X \leftarrow \cup_{\ell\in\mathcal{N}} X_\ell$
enquanto $X \neq \emptyset$ **faz**
  escolher $Y_i \subset X_i$
  para todo o arco $(i,j) \in \mathcal{A}$ **faz**
    $Z \leftarrow \{y_i + \mathbf{c_{i,j}} : y_i \in Y_i\}$
    $\Pi_j \leftarrow$ rótulos não dominados de $\Pi_j \cup Z$
    $X_j \leftarrow \Pi_j \cap (X_j \cup Z)$
    $X \leftarrow \cup_{\ell\in\mathcal{N}} X_\ell$
  fim_para
fim_enquanto

Algoritmo de Rotulação

rótulos não definitivos
- mais fáceis de implementar
- nenhuma restrição

rótulos definitivos
- mais eficientes
- escolhe o rótulo lexicograficamente menor

Exemplos:

- **Problema do trajecto de menor custo por unidade de tempo**

$$\begin{array}{rcl} f: \mathcal{P}_{\mathcal{G}} & \longrightarrow & \mathbb{R} \\ p & \longmapsto & f(p) = \frac{\sum_p c_{i,j}}{\sum_p t_{i,j}}, \end{array}$$

Pretende-se $\min\limits_{p \in \mathcal{P}} \{f(p)\}$

- **Problema do trajecto de menor custo por unidade de capacidade**

$$\begin{array}{rcl} f: \mathcal{P}_{\mathcal{G}} & \longrightarrow & \mathbb{R} \\ p & \longmapsto & f(p) = \frac{\sum_p c_{i,j}}{\min_p \{\mu_{i,j}\}}, \end{array}$$

Pretende-se $\min\limits_{p \in \mathcal{P}} \{f(p)\}$

- **Problema da transmissão de dados pela via mais rápida**

$$\begin{array}{rcl} f: \mathcal{P}_{\mathcal{G}} & \longrightarrow & \mathbb{R} \\ p & \longmapsto & f(p) = \sum_p t_{i,j} + \frac{\sigma}{\min_p \{\mu_{i,j}\}}, \end{array}$$

Pretende-se $\min\limits_{p \in \mathcal{P}} \{f(p)\}$

Problemas do trajecto óptimo em $\mathbb{R}$
que envolvem a minimização
de uma função quociente,

$$\min_{p \in \mathcal{P}} \{ f(p) = \frac{f_1(p)}{f_2(p)} \},$$

não verificam, em geral,
o Princípio de Optimalidade (Fraca)

$$\Downarrow$$

algoritmos de rotulação
não garantem a solução óptima.

Sob certas condições, podem ser vistos com um problema biobjectivo:

$$\min_{p \in \mathcal{P}} \{ f_1(p) \} \quad \text{e} \quad \max_{p \in \mathcal{P}} \{ f_2(p) \}$$

É possível obter a solução óptima para a função quociente
à custa das soluções óptimas do problema biobjectivo.
(depende em geral da função objectivo)

Algoritmos baseados no *"ranking"*

Baseiam-se na enumeração de trajectos (*"ranking"*)

NOTA:

MS e **MPS**: algoritmos muito eficientes na realização do *"ranking"* para o problema do trajecto mais curto em $\mathbb{R}$.

"Ranking" $\Rightarrow$ Ordem TOTAL em $\mathbb{R}^k$.

ordem *lexicográfica*

Dados $\mathbf{a} = (a_1, \ldots, a_k), \mathbf{b} = (b_1, \ldots, b_k) \in \mathbb{R}^k$

$$\boxed{(\mathbf{a} \preceq_{\mathbf{L}} \mathbf{b}) \quad \Leftrightarrow \quad \text{ou} \quad \begin{array}{l} \mathbf{a} = \mathbf{b} \\ \\ \exists i \in \{1, \ldots, k\} \\ \quad \text{tal que} \begin{cases} a_j = b_j, j = 1, \ldots, i-1 \\ \text{e} \\ a_i < b_i \end{cases} \end{array}}$$

$$\boxed{\text{“}\preceq_L\text{”: Relação de ordem TOTAL em } \mathbb{R}^k.}$$

Impõe uma ordenação dos objectivos

$$\Downarrow$$

não modeliza correctamente
o problema do trajecto óptimo multiobjectivo

Definição 1 : *Sejam* $p, q \in \mathcal{P}_{i,j}$. *Define-se a relação "*$\preceq_L$*" em* $\mathcal{P}_\mathcal{G}$ *da seguinte forma:*

$$p \preceq_L q \Leftrightarrow f(p) \preceq_L f(q).$$

Prova-se:

$$\boxed{\text{“}\preceq_L\text{”: Relação de pré-ordem total em } \mathcal{P}.}$$

Teorema 1 : *Sejam* $\mathbf{a}$ *e* $\mathbf{b}$ *dois elementos de* $\mathbb{R}^k$*, tais que* $\mathbf{a} \preceq_{\mathbf{D}} \mathbf{b}$*. Então* $\mathbf{a} \preceq_{\mathbf{L}} \mathbf{b}$ *(isto é, um trajecto só pode ser dominado por trajectos lexicograficamente menores).*

Corolário 1 *Seja A um subconjunto de $\mathbb{R}^k$. O mínimo de A relativamente à relação $\preceq_L$ é* **um** *mínimo de A relativamente à relação $\preceq_D$ (isto é, o trajecto lexicograficamente mais curto é não dominado).*

Ordem lexicográfica + Teorema

$$\Downarrow$$

"ranking"
(usando os algoritmos **MS** e **MPS**)

Algoritmo MS para o trajecto mais curto

Quando terminar o *"ranking"*?

(condição de paragem)

Teorema 2 :
Seja $p \in \mathcal{P}$. *Para determinar* $\bar{D}$ *basta realizar* k "rankings"*, um em cada objectivo, onde o limite superior do i-ésimo* "ranking" *é* $f_i(p)$.

NOTA:

Se p é o trajecto lexicograficamente menor em algum dos objectivos, então não é necessário realizar o *"ranking"* nesse objectivo (basta realizar o *"ranking"* em $k-1$ objectivos).

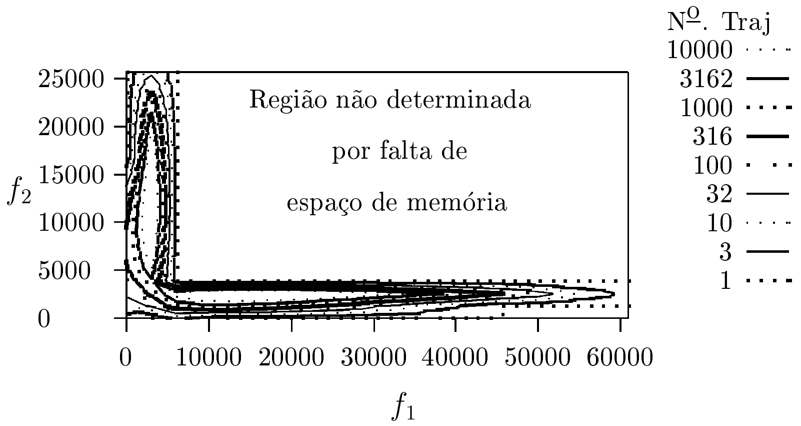


Figura 1: **Número de trajectos determinados segundo os *ranking***

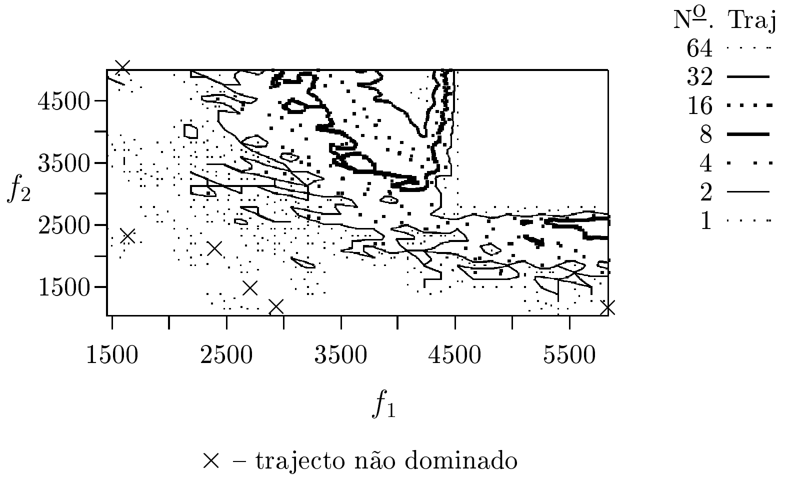


Figura 2: **Ampliação parcial da figura 1**

Como encontrar p para usar o teorema 2?

Versões implementadas:

- **muda1**: realiza $\alpha\%$ do *"ranking"* no primeiro objectivo e seguidamente o *"ranking"* no segundo objectivo.

- **muda2**: realiza o *"ranking"* no primeiro objectivo até garantir que só é necessário realizar $\alpha\%$ do *"ranking"* no segundo objectivo.

$$\boxed{\text{Valor óptimo para } \alpha?}$$

- **muda3**: realiza o *"ranking"* no primeiro objectivo até que a percentagem do *"ranking"* realizado no primeiro objectivo seja maior do que a percentagem do *"ranking"* que falta realizar no segundo objectivo.

- **muda4**: realiza, em paralelo, o *"ranking"* no primeiro e no segundo objectivo até encontrar um trajecto em comum.

$$\boxed{\text{Ranking oblíquo}}$$

Algoritmo MPS

`{Algoritmo MPS}`
$\mathcal{T}_t \leftarrow$ árvore dos trajectos lexicograficamente menores para t

`{Mudança de variável (custos reduzidos)}`
para todo o arco $(i,j) \in \mathcal{A}$ **faz** $\bar{\mathbf{c}}_{\mathbf{i,j}} \leftarrow \mathbf{c}_{\mathbf{i,j}} + \pi_{\mathbf{j}} - \pi_{\mathbf{i}}$

Usando a ordem lexicografica,
    colocar a rede na *sorted forward star form*
$p \leftarrow \mathcal{T}_t(s)$
$X \leftarrow \{p\}$
noDesvio(p) $\leftarrow s$
arcoDesvio(p) $\leftarrow$ arco seguinte a s em p

enquanto $X \neq \emptyset$ **faz**
  $p \leftarrow$ trajecto lexicigrafico menor em X
  $X \leftarrow X - \{p\}$
  $i \leftarrow$ noDesvio(p)
  $k \leftarrow$ arcoDesvio(p)

  `{procurar próximo irmão}`
  procurar_proximo_desvio(i, k)

  $i \leftarrow$ nó a seguir a i em p
  para j desde i até t em p **faz**
    se $p_{s,j}$ é caminho não dominado
    então
      $k \leftarrow$ arco seguinte a j em p
      procurar_proximo_desvio(j, k)
    fim_então
    senão sair do ciclo "**para ... faz**"
  fim_para
fim_enquanto

determinar o desvio seguinte

{$\mathcal{T}$: `árvore dos trajectos lexicograficamente menores para` t}
{X: `conjunto de trajectos candidatos a trajectos não dominados`}
{π_i: `custo do trajecto lexicográfico menor de` i `até` t}
{p: `trajecto da forma` $\langle v_1, (v_1, v_2), \ldots, (v_{\ell-1}, v_\ell), v_\ell \rangle$}
{A_i: `conjunto dos arcos que se iniciam em` i}

procedimento procurar_proximo_desvio($no, arco$)
    repete
        $arco \leftarrow arco + 1$
    até que ($arco \in \mathcal{A}_{no}$) e ($p_{s,no} \diamondsuit \langle arco \rangle$ é caminho ND)
    se existe $p_{s,no} \diamondsuit \langle arco \rangle$
    então
        $novoDesvio \leftarrow p_{s,no} \diamondsuit \langle arco \rangle \diamondsuit \mathcal{T}_{suc(arco)}$
        noDesvio($novoDesvio$) $\leftarrow no$
        arcoDesvio($novoDesvio$) $\leftarrow arco$
        $X \leftarrow X \cup \{novoDesvio\}$
    fim_então
fim_procedimento