

Sobre o problema de caminhos de comprimento mínimo em grafos

Nota à cerca do algoritmo de Dijkstra

por J. A. Melo de Sousa

Departamento de Matemática — Universidade de Coimbra

Resumo

Neste artigo são analisadas algumas versões do algoritmo de DIJKSTRA, para a determinação do caminho de comprimento mínimo entre dois vértices de um grafo, e proposta uma outra.

Résumé

Dans cet article on analyse quelques versions de l'algorithme de DIJKSTRA, pour le problème du plus court chemin entre deux sommets d'un graphe, et on propose une autre.

Summary

In this paper some versions of the DIJKSTRA's algorithm are analysed in order to find, in a graph, the shortest path between two nodes, and a proposal of another one is presented.

1. Começamos por definir os conceitos básicos utilizados ao longo do texto.

Um grafo $G = (X, U)$ é o par constituído por um conjunto $X = \{x_1, x_2, \dots, x_n\}$ de elementos, usualmente denominados *nodos*

ou *vértices* do grafo, e por uma família $U = \{u_1, u_2, \dots, u_m\}$ de elementos do produto cartesiano

$$X \times X = \{(x_i, x_j) \mid x_i \in X, x_j \in X\}.$$

Uma outra maneira de representar o grafo é a seguinte:

Seja Γ uma correspondência multívoca de X em X , definida para todo o $x_i \in X$ do seguinte modo:

$$\Gamma(x_i) = \{(x_j, x_i), \dots, (x_k, x_i)\}$$

em que $(x_i, x_j), (x_i, x_j), \dots, (x_i, x_k)$ são elementos de $X \times X$. Então o grafo $G = (X, U)$ pode representar-se também por $G = (X, \Gamma)$. Se os elementos $(x_i, x_j) \in X \times X$ constituem pares ordenados, são chamados *arcos* do grafo e este é dito *orientado* ou *dirigido*; caso contrário, esses pares são denominados *arestas* ou *lados* e o grafo é dito *não orientado*. A família das arestas é representada por $E = \{e_1, e_2, \dots, e_m\}$ e o grafo $G = (X, E)$ é denominado *multigrafo*. Podemos sempre considerar um multigrafo como um grafo orientado. Para tal basta considerar que cada aresta entre os vértices x_i e x_j se pode «desdobrar» em dois arcos do tipo (x_i, x_j) e (x_j, x_i) .

Num arco (x_i, x_j) , o vértice x_i é chamado *extremidade* ou *vértice inicial* e x_j

extremidade ou *vértice final*. Um arco da forma (x_i, x_i) é denominado *lacete*.

Um *p-grafo* é aquele em que um elemento $(x_i, x_j) \in X \times X$ aparece quanto muito *p* vezes. Ao número de vértices de um grafo chamaremos *ordem* do grafo.

Dois arcos (arestas) são ditos *adjacentes* se têm pelo menos uma extremidade em comum. Se um vértice x é extremidade inicial (final) de um arco u , diremos que o arco u é *incidente* em x em direcção ao exterior (interior). Para arestas, diremos apenas que a aresta é *incidente* no vértice x .

A uma sequência $\mu = (u_1, u_2, \dots, u_k)$ de arcos de G tal que cada arco da sequência tenha uma extremidade comum com o arco seguinte, chamaremos *cadeia*. Uma cadeia que passa uma só vez por um mesmo vértice é dita *elementar*; uma cadeia é *simples* se não utiliza duas vezes o mesmo arco. *Caminho* é uma cadeia μ de um tipo particular onde para todo o arco u_i com $i < k$, a extremidade final de u_i coincide com a extremidade inicial de u_{i+1} . Torna-se evidente que em multigrafos os conceitos de cadeia e caminho coincidem. *Ciclo* é uma cadeia μ tal que:

- 1.º o mesmo arco não figure duas vezes na sequência;
- 2.º os dois vértices das extremidades da cadeia coincidam.

Circuito é um ciclo orientado, ou seja, para todo o $i < k$, a extremidade terminal (ou final) de u_i coincide com a extremidade inicial de u_{i+1} .

Fecho transitivo de um vértice de um grafo é o sub-conjunto de vértices do grafo que se podem atingir a partir do vértice dado, por um caminho de comprimento qualquer:

$$\hat{\Gamma}(x) = \{x\} \cup \Gamma\{x\} \cup \Gamma^2\{x\} \cup \dots$$

Por grafo *conexo* entendemos aquele em

que para todo o par de vértices x_i e x_j distintos, existe sempre uma cadeia que os liga. Num grafo $G = (X, U)$, chamamos *raiz* a um vértice x tal que exista um caminho de x para todos os restantes vértices do grafo.

Uma *árvore* é um grafo conexo e sem ciclos. Chamaremos *arborescência* a uma árvore munida de uma *raiz*.

Denominaremos *distância* entre dois vértices ligados por um arco, à função a definida no conjunto dos arcos do grafo sobre o conjunto dos números reais não negativos. Assim $a(x_i, x_j)$ representará o valor da distância entre os vértices x_i e x_j .

2. Partindo do conhecimento dos conceitos acabados de definir, poderemos enunciar o problema em questão, do seguinte modo:

Dado um 1 — grafo $G = (X, U)$ conexo, definida uma função distância entre dois vértices do grafo ligados por um arco, determinar o valor e o caminho de «comprimento mínimo» entre dois determinados vértices x_I (inicial) e x_F (final). Uma condição necessária e suficiente para que tal caminho exista é a seguinte:

O vértice x_F terá que pertencer ao fecho transitivo do vértice x_I .

3. Dos vários algoritmos conhecidos da literatura, descrevemos o mais eficiente, do ponto de vista computacional, da autoria de E. W. DIJKSTRA [1].

Analiseemos a forma como em cada iteração se procede à determinação do vértice e do arco (aresta) a seleccionar no grafo, para formação do caminho (cadeia) de comprimento mínimo. Para tal, os vértices serão subdivididos em três conjuntos:

— Conjunto A — dos vértices para os quais o caminho mínimo a partir de x_I é já conhecido (em cada iteração um novo vértice é adicionado a (transferido para) este conjunto).

— Conjunto B — dos vértices de onde o próximo vértice a ser adicionado ao conjunto A , será seleccionado; este conjunto contém todos os vértices que estão ligados pelo menos a um vértice do conjunto A , mas que ainda não pertencem a A .

— Conjunto C — dos restantes vértices.

Por seu lado, os arcos (arestas) também estarão subdivididos em três conjuntos:

— Conjunto I — dos arcos que formam os caminhos mais curtos de x_I para os respectivos vértices do conjunto A .

— Conjuntos II — dos arcos donde vai ser seleccionado o próximo a transferir para o conjunto I ; um e um só arco deste conjunto ligará um vértice de A a um vértice de B .

— Conjuntos III — dos restantes arcos.

À partida todos os vértices pertencem ao conjunto C e todos os arcos ao conjuntos III .

Iniciaremos o processo de cálculo transferindo o vértice x_I para A e aplicando consecutiva e repetidamente os seguintes passos:

Passo 1 — Consideremos todos os arcos r que ligam directamente o vértice acabado de transferir para o conjunto A , com vértices P nos conjuntos B ou C . Então:

a) Se o vértice P pertence ao conjunto C , ele será transferido para o conjunto B e o correspondente arco r transferido para o conjunto II .

b) Se o vértice P pertence ao conjunto B , vejamos se a utilização do arco r conduz a um caminho de comprimento menor entre x_I e P do que o conhecido até à data que utiliza o correspondente arco do conjunto II . Em caso afirmativo, o arco r substitui aquele em II . Caso contrário, o arco r é rejeitado.

Passo 2 — Cada vértice em B pode ligar-se com o vértice x_I de uma só maneira, se nos restringirmos apenas aos arcos do

conjunto I e a um único do conjunto II . Assim, existirão sempre caminhos de x_I para os vértices do conjunto B , com determinados valores-comprimento. Transferimos o vértice de B que determina o valor mínimo para o conjunto A , passando do conjunto II para o conjunto I o correspondente arco. Voltamos novamente ao Passo 1 e repetimos o processo até o vértice final x_F ser transferido para o conjunto A . Então o caminho procurado é determinado pela sucessão ou dos vértices (pertencentes ao conjunto I), considerados de x_F para X_I , ou dos arcos (pertencentes ao conjunto A).

3. Por razões de carácter computacional, DREYFUS [2] apresenta do mesmo algoritmo outra versão.

Assinala os vértices x_i do grafo com rótulos, que poderão ser do tipo $[x_j, d(x_i)]^{(1)}$, definitivos ou provisórios, consoante o valor do rótulo representa ou não o valor do caminho de comprimento mínimo de x_I para x_i .

Inicialmente, rotula o vértice x_I com o rótulo definitivo $[-, 0]$ e todos os restantes com o rótulo provisório $[-, \infty]^{(2)}$.

Então, compara, um por um, o valor de cada rótulo, excepção feita para o vértice x_I , com o valor obtido através da soma do valor do rótulo do vértice x_I (igual a 0) com a distância directa de x_I ao vértice em questão. Sendo N o número de vértices do grafo, o conjunto dos valores dos $N-1$ rótulos provisórios admitirá um mínimo. Seja k o vértice cujo rótulo tem valor mínimo. Declara então esse rótulo definitivo. Seguidamente, para cada um dos restantes $N-2$ rótulos provisórios, compara o seu valor com o do

⁽¹⁾ x_j representa o vértice que precede x_i no caminho de x_I para x_i ; $d(x_i)$ representa o valor (comprimento) do caminho de x_I para x_i , denominado *valor do rótulo*.

⁽²⁾ O sinal ∞ significa que ainda não foi determinado nenhum caminho de x_I até esse vértice.

rótulo de k declarado definitivo adicionado à distância directa deste vértice ao vértice em análise. O menor destes dois valores será o valor do novo rótulo provisório.

De forma análoga à descrita, determina o mínimo dos valores dos $N-2$ rótulos provisórios e declara definitivo o rótulo cujo valor é mínimo, o qual se toma como base para nova transformação dos $N-3$ rótulos provisórios. Ao fim de quanto muito $N-1$ iterações, o rótulo do vértice x_F será definitivo, terminando o algoritmo, e determinando o(s) caminho(s) de comprimento mínimo:

a) Pela sucessão dos vértices indicados na 1.^a parte dos rótulos, partindo de x_F até atingir x_I ; ou

b) Pela sucessão dos arcos para os quais se verifica

$$(3.1) \quad d(x_j) = d(x_i) - a(x_j, x_i)$$

com $1 \leq i, j \leq N$ e $i \neq j$, tomando inicialmente $i = F$ e terminando quando $j = I$.

4. Obtêm-se, de facto, por esta última via, 3. b), todos os caminhos de comprimento mínimo entre os dois vértices considerados como início e fim do caminho, x_I e x_F . No entanto o mesmo resultado já não é obtido quando se aplica a versão original do algoritmo ou aquela que é indicada em 3. a), ou ainda a descrita por PRICE [3]. Demonstra-se esta afirmação resolvendo o problema indicado na Fig. 1, para ambas as versões.

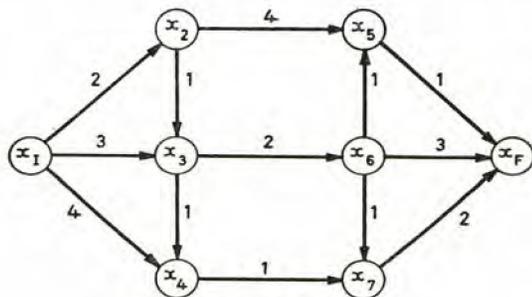


Fig. 1

Os quadros I e II representados nas Figs. 2 e 3 fornecem as soluções obtidas.

5. Baseados no algoritmo de DIJKSTRA tal como é descrito por DREYFUS e PRICE, propomos esta outra versão, por forma a obter todos os caminhos de comprimento mínimo entre x_I e x_F , tendo em atenção a via indicada em 3. a).

Antes de iniciarmos propriamente a descrição do algoritmo comecemos por explicar o significado de alguns símbolos nele contido:

- l — indicador da ordem da iteração corrente;
- K_i — vértices que em cada iteração ficam assinalados com rótulos definitivos;
- i — índices desses vértices, que satisfazem a expressão (5.1).
- $q_p^{(l)}$ — número de rótulos que assinalam o vértice x_p , satisfazendo as expressões (5.2) e (5.3) na iteração de ordem l ;
- $f_p^{(l)}$ — número de rótulos que assinalam o vértice x_p , satisfazendo a expressão (5.4) na iteração de ordem l ;
- $T_p^{(l)}$ — número total de rótulos apostos ao vértice x_p , na iteração de ordem l .

Descrição do algoritmo

Passo 0 — Assinalemos x_I com o rótulo definitivo $[-, 0]$ e todos os restantes vértices x_i com rótulos provisórios $[-, \infty]$. Seja P o conjunto de vértices x_p que têm rótulos provisórios.

► Façamos $K_i^{(l)} = K_1^{(l)} \equiv x_I$ e prossigamos directamente com o Passo 2.

Passo 1. Determinemos os vértices $K_i^{(l)}$ com $i \in \mathbb{N}$ e tal que $1 \leq i \leq N-1$ por forma a que:

$$(5.1) \quad d(K_i^{(l)}) = \text{Min}_{x_p \in P} [d(x_p)],$$

QUADRO I

Resolução segundo a versão de DIJKSTRA [1]

Início	A	B	C	I	II	III
	x_I		x_2, x_3, x_4 x_5, x_6, x_7 x_F			$(x_I, x_2), (x_I, x_3)$ $(x_I, x_4), (x_2, x_3)$ $(x_3, x_4), (x_5, x_6)$
1.1	x_I	x_2, x_3 x_4	x_5, x_6 x_7, x_F		$(x_I, x_2), (x_I, x_3)$ (x_I, x_4)	$(x_3, x_6), (x_4, x_7)$
2.1	x_I, x_2	x_3, x_4	x_5, x_6 x_7, x_F	(x_I, x_2)	$(x_I, x_3), (x_I, x_4)$	$(x_6, x_5), (x_6, x_7)$ $(x_5, x_F), (x_6, x_F)$ (x_7, x_F)
1.2	x_I, x_2	x_3, x_4 x_5	x_6, x_7 x_F	(x_I, x_2)	$(x_I, x_3), (x_I, x_4)$ (x_2, x_5)	
2.2	x_I, x_2, x_3	x_4, x_5	x_6, x_7 x_F	$(x_I, x_2), (x_I, x_3)$	$(x_I, x_4), (x_2, x_5)$	
1.3	x_I, x_2, x_3	x_4, x_5 x_6	x_7, x_F	$(x_I, x_2), (x_I, x_3)$	$(x_I, x_4), (x_2, x_5)$ (x_3, x_6)	
2.3	x_I, x_2, x_3, x_4	x_5, x_6	x_7, x_F	$(x_I, x_2), (x_I, x_3), (x_I, x_4)$	$(x_2, x_5), (x_3, x_6)$	
1.4	x_I, x_2, x_3, x_4	x_5, x_6 x_7	x_F	$(x_I, x_2), (x_I, x_3), (x_I, x_4)$	$(x_2, x_5), (x_3, x_6)$ (x_4, x_7)	
2.4	x_I, x_2, x_3, x_4 x_7	x_5, x_6	x_F	$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ (x_4, x_7)	$(x_2, x_5), (x_3, x_6)$	
1.5	x_I, x_2, x_3, x_4 x_7	x_5, x_6 x_F		$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ (x_4, x_7)	$(x_2, x_5), (x_3, x_6)$ (x_7, x_F)	
2.5	x_I, x_2, x_3, x_4 x_7, x_6	x_5, x_F		$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ (x_4, x_7) (x_3, x_6)		
1.6	x_I, x_2, x_3, x_4 x_7, x_6	x_5, x_F		$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ $(x_4, x_7), (x_3, x_6)$	$(x_2, x_5), (x_7, x_F)$	
2.6	x_I, x_2, x_3, x_4 x_7, x_6, x_5	x_F		$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ $(x_4, x_7), (x_3, x_6), (x_2, x_5)$	(x_7, x_F)	
1.7	x_I, x_2, x_3, x_4 x_7, x_6, x_5	x_F		$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ $(x_4, x_7), (x_3, x_6), (x_2, x_5)$	(x_7, x_F)	
2.7	x_I, x_2, x_3, x_4 x_7, x_6, x_5, x_F			$(x_I, x_2), (x_I, x_3), (x_I, x_4)$ $(x_4, x_7), (x_3, x_6), (x_2, x_5)$ (x_7, x_F)		

Fig. 2

Solução obtida: $[(x_I, x_4), (x_4, x_7), (x_7, x_F)]$ com o valor total de 7 unidades.

QUADRO II (5)

Resolução segundo DREYFUS [2] (via indicada em 3. a) e PRICE [3]

x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8
$[-, 0]$	$[-, \infty]$						
	$[x_1, 2]$	$[x_1, 3]$	$[x_1, 4]$	$[-, \infty]$	$[-, \infty]$	$[-, \infty]$	$[-, \infty]$
		$[x_1, 3]$	$[x_1, 4]$	$[x_2, 6]$	$[-, \infty]$	$[-, \infty]$	$[-, \infty]$
			$[x_1, 4]$	$[x_2, 6]$	$[x_3, 5]$	$[-, \infty]$	$[-, \infty]$
				$[x_2, 6]$	$[x_3, 5]$	$[x_4, 5]$	$[-, \infty]$
				$[x_2, 6]$			$[x_7, 7]$
							$[x_7, 7]$

Fig. 3

Solução obtida: $[(x_1, x_4), (x_4, x_7), (x_7, x_8)]$ com o valor total de 7 unidades.

considerando-se os vértices $K_i^{(l)}$ com rótulos definitivos.

Se algum $K_i^{(l)} = x_8$, o algoritmo termina, e os caminhos de comprimento mínimo são determinados «caminhando» de x_8 para x_1 , atendendo à sucessão dos vértices considerados na 1.ª parte dos rótulos e tomando em linha de conta todas as alternativas.

Passo 2 — Consideremos de per si, todos os vértices x_p que estão directamente ligados por um arco aos vértices $K_i^{(l)}$; se não existirem tais vértices x_p , então façamos $K_i^{(l)} = K_i^{(l-1)}$ e procuremos os vértices $x_p \in P$

que estejam directamente ligados aos $K_i^{(l-1)}$. Se tais vértices não existirem, repetimos o processo até encontrarmos um $K_i^{(l)}$ que esteja ligado a, pelo menos, um vértice $x_p \in P$. Então, dar-se-á um dos três casos seguintes:

1.º Caso — Se, para cada $K_i^{(l)}$

$$(5.2) \quad d_i(x_p) = d(K_i^{(l)}) + a(K_i^{(l)}, x_p) < d(x_p)$$

(3) No quadro, as entradas assinaladas a duplo traço indicam o vértice e o rótulo que nessa iteração foi considerado definitivo.

então o novo valor do rótulo do vértice x_p será

$$(5.3) \quad d(x_p) = \underset{i}{\text{Min}} [d_i(x_p)]$$

e atribuíamos novos rótulos, em número de $q_p^{(i)}$ (com $q \leq i$) ao vértice x_p (4).

2.º Caso — Se, para cada $K_i^{(l)}$ se tem

$$(5.4) \quad d_i(x_p) = d(K_i^{(l)}) + a(K_i^{(l)}, x_p) = d(x_p)$$

então sejam $f_p^{(l)}$ o número de rótulos que verificam (5.4). Há no entanto a juntar a este quantitativo o número de rótulos que o vértice em consideração tinha na iteração anterior $T_p^{(l-1)}$.

Então, o vértice x_p ficará com $T_p^{(l)} = T_p^{(l-1)} + f_p^{(l)}$ rótulos.

3.º Caso — Se, para cada $K_i^{(l)}$

$$(5.5) \quad d_i(x_p) = d(K_i^{(l)}) + a(K_i^{(l)}, x_p) > d(x_p)$$

então nenhuma alteração é feita nos rótulos do vértice considerado.

Depois de todos os vértices $x_p \in P$ directamente ligados com os $K_i^{(l)}$ terem sido considerados, retornamos ao Passo 1 e damos início a nova iteração.

A demonstração da convergência desta versão do algoritmo de DIJKSTRA, é a mesma da apresentada por DREYFUS e PRICE.

6. Para uma melhor compreensão deste processo de cálculo, descrevemos pormenorizadamente a resolução do problema apresentado em 4. na Fig. 1.

(4) Os $q_p^{(i)}$ rótulos provisórios são do tipo $[K_i^{(l)}, d(x_p)]$ e tanto quantos os novos $K_i^{(l)}$ determinados nessa iteração, verificando a expressão (5.1).

1.ª Iteração — $l = 1$.

Passo 0 — Vértice x_I assinalado com o rótulo definitivo $[-, 0]$ e todos os restantes vértices com rótulos provisórios $[-, \infty]$.

$$P = \{x_2, x_3, x_4, x_5, x_6, x_7, x_8\}$$

$$K_i^{(1)} = K_1^{(1)} \equiv x_I$$

$$d(x_I) = 0$$

Passo 2

— Vértice x_2

$$d_1(x_2) = d(x_I) + a(x_I, x_2) = 0 + 2 = 2 < d(x_2) \quad (1.º \text{ caso})$$

$$q_2^{(1)} = 1; T_2^{(1)} = 1.$$

— Vértice x_3

$$d_1(x_3) = d(x_I) + a(x_I, x_3) = 0 + 3 = 3 < d(x_3) \quad (1.º \text{ caso})$$

$$q_3^{(1)} = 1; T_3^{(1)} = 1.$$

Vértice x_4

$$d_1(x_4) = d(x_I) + a(x_I, x_4) = 0 + 4 = 4 < d(x_4) \quad (1.º \text{ caso})$$

$$q_4^{(1)} = 1; T_4^{(1)} = 1.$$

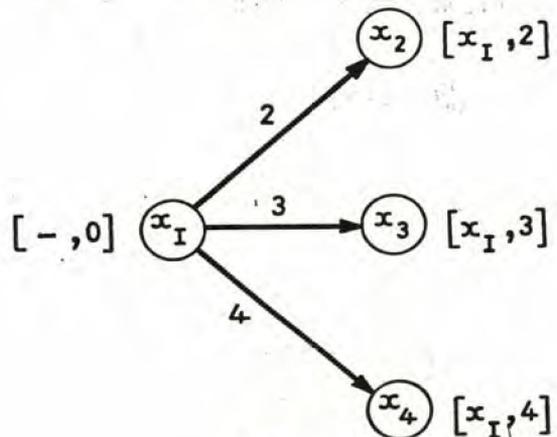


Fig. 4

2.^a Iteração - $l = 2$.

Passo 1

$$\begin{aligned} P &= \{x_2, x_3, x_4, x_5, x_6, x_7, x_F\} \\ d(K_1^{(2)}) &= \text{MIN}[d(x_2), d(x_3), d(x_4), \\ &\quad d(x_5), d(x_6), d(x_7), d(x_F)] \\ &= \text{MIN}[2, 3, 4, \infty, \infty, \infty, \infty] \\ &= 2 \end{aligned}$$

$$d(K_1^{(2)}) = d(K_1^{(2)}) = 2 \rightarrow K_1^{(2)} \equiv x_2.$$

Ao vértice x_2 é atribuído o rótulo definitivo $[x_2, 2]$.

Passo 2

- Vértice x_3

$$\begin{aligned} d_1(x_3) &= d(x_2) + a(x_2, x_3) = 2 + 1 = \\ &= 3 = d(x_3) \quad (2.^{\circ} \text{ caso}) \end{aligned}$$

$$f_3^{(2)} = 1; T_3^{(1)} = 1; T_3^{(2)} = T_3^{(1)} + f_3^{(2)} = 1 + 1 = 2.$$

- Vértice x_5

$$\begin{aligned} d_1(x_5) &= d(x_2) + a(x_2, x_5) = 2 + 4 = \\ &= 6 < d(x_5) \quad (1.^{\circ} \text{ caso}) \end{aligned}$$

$$q_5^{(2)} = 1; T_5^{(2)} = 1$$

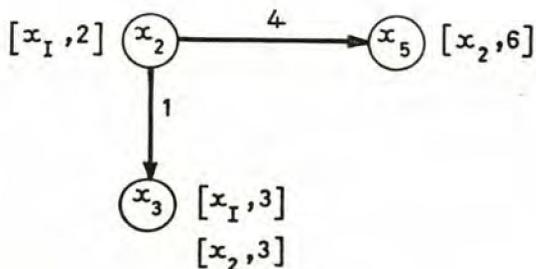


Fig. 5

3.^a Iteração - $l = 3$.

Passo 1

$$\begin{aligned} P &= \{x_3, x_4, x_5, x_6, x_7, x_F\} \\ d(K_1^{(3)}) &= \text{MIN}[d(x_3), d(x_4), d(x_5), \\ &\quad d(x_6), d(x_7), d(x_F)] \\ &= \text{MIN}[3, 4, 6, \infty, \infty, \infty] \\ &= 3 \end{aligned}$$

$$d(K_1^{(3)}) = d(K_1^{(3)}) = 3 \rightarrow K_1^{(3)} \equiv x_3.$$

Ao vértice x_3 são atribuídos os rótulos definitivos $[x_3, 3]$ e $[x_2, 3]$.

Passo 2

- Vértice x_4

$$\begin{aligned} d_1(x_4) &= d(x_3) + a(x_3, x_4) = 3 + 1 = \\ &= 4 = d(x_4) \quad (2.^{\circ} \text{ caso}) \end{aligned}$$

$$f_4^{(3)} = 1; T_4^{(1)} = T_4^{(2)} = 1; T_4^{(3)} = T_4^{(2)} + f_4^{(3)} = 1 + 1 = 2.$$

- Vértice x_6

$$\begin{aligned} d_1(x_6) &= d(x_3) + a(x_3, x_6) = 3 + 2 = \\ &= 5 < d(x_6) \quad (1.^{\circ} \text{ caso}) \end{aligned}$$

$$q_6^{(3)} = 1; T_6^{(3)} = 1.$$

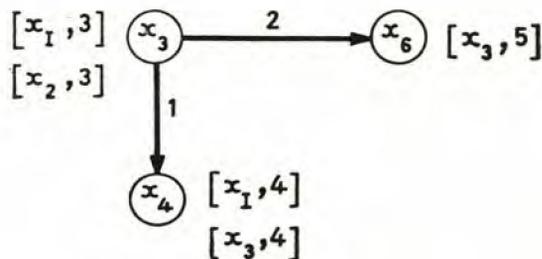


Fig. 6

4.^a Iteração - $l = 4$.

Passo 1

$$P = \{x_4, x_5, x_6, x_7, x_F\}$$

$$d(K_1^{(4)}) = \text{MIN} [d(x_4), d(x_5), d(x_6), d(x_7), d(x_F)]$$

$$= \text{MIN} [4, 6, 5, \infty, \infty]$$

$$= 4$$

$$d(K_1^{(4)}) = d(K_1^{(4)}) = 4 \rightarrow K_1^{(4)} \equiv x_4.$$

Ao vértice x_4 são atribuídos os rótulos definitivos $[x_7, 4]$ e $[x_3, 4]$.

Passo 2

- Vértice x_7

$$d_1(x_7) = d(x_4) + a(x_4, x_7) = 4 + 1 = 5 < d(x_7) \quad (1.^\circ \text{ caso})$$

$$q_7^{(4)} = 1; T_7^{(4)} = 1$$

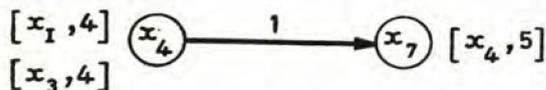


Fig. 7

5.^a Iteração - $l = 5$

Passo 1

$$P = \{x_5, x_6, x_7, x_F\}$$

$$d(K_1^{(5)}) = \text{MIN} [d(x_5), d(x_6), d(x_7), d(x_F)]$$

$$= \text{MIN} [6, 5, 5, \infty]$$

$$= 5$$

$$d(K_1^{(5)}) = d(K_1^{(5)}) = d(K_2^{(5)}) = 5 \rightarrow K_1^{(5)} \equiv x_6$$

e

$$K_2^{(5)} \equiv x_7$$

Aos vértices x_6 e x_7 são atribuídos respectivamente os rótulos definitivos $[x_3, 5]$ e $[x_4, 5]$.

Passo 2

- Vértice x_5

$$d_1(x_5) = d(x_6) + a(x_6, x_5) = 5 + 1 = 6 = d(x_5) \quad (2.^\circ \text{ caso})$$

$$f_5^{(5)} = 1; T_5^{(4)} = T_5^{(3)} = T_5^{(2)} = 1;$$

$$T_5^{(5)} = T_5^{(4)} + f_5^{(5)} = 1 + 1 = 2.$$

- Vértice x_F

$$d_1(x_F) = d(x_6) + a(x_6, x_F) = 5 + 3 = 8 < d(x_F) \quad (1.^\circ \text{ caso})$$

$$d_2(x_F) = d(x_7) + a(x_7, x_F) = 5 + 2 = 7 < d(x_F) \quad (1.^\circ \text{ caso}).$$

O novo valor do rótulo do vértice x_7 será:

$$d(x_F) = \text{MIN} [d_1(x_F), d_2(x_F)]$$

$$= \text{MIN} [8, 7]$$

$$= 7$$

$$q_F^{(5)} = 1; T_F^{(5)} = 1$$

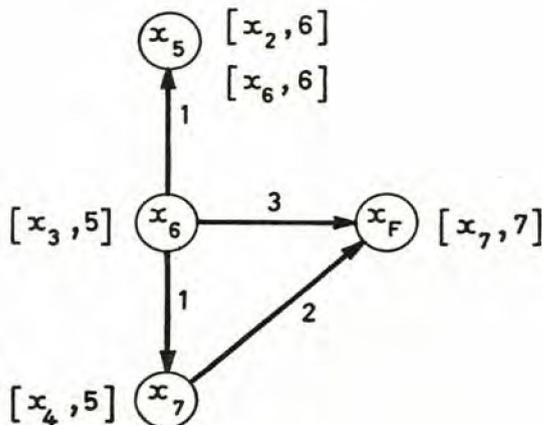


Fig. 8

6.^a Iteração — $l = 6$

Passo 1

$$\begin{aligned} P &= \{x_5, x_F\} \\ d(K_i^{(6)}) &= \text{MIN}[d(x_5), d(x_F)] \\ &= \text{MIN}[6, 7] \\ &= 6 \\ d(K_i^{(6)}) &= d(K_1^{(6)}) = 6 \rightarrow K_1^{(6)} \equiv x_5 \end{aligned}$$

Ao vértice x_5 serão atribuídos os rótulos definitivos $[x_2, 6]$ e $[x_6, 6]$.

Passo 2

— Vértice x_F

$$\begin{aligned} d_1(x_F) &= d(x_5) + a(x_5, x_F) = 6 + 1 = \\ &= 7 = d(x_F) \quad (2.^\circ \text{ caso}) \\ f_F^{(6)} &= 1; T_F^{(6)} = T_F^{(5)} + f_F^{(6)} = 1 + 1 = 2 \end{aligned}$$

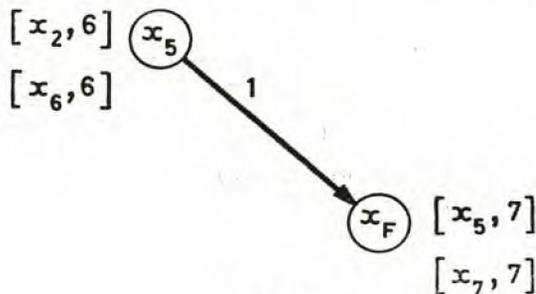


Fig. 9

7.^a Iteração — $l = 7$

Passo 1

$$\begin{aligned} P &= \{x_F\} \\ d(K_i^{(7)}) &= \text{MIN}[d(x_F)] \\ &= 7 \\ d(K_i^{(7)}) &= d(K_1^{(7)}) = 7 \rightarrow K_1^{(7)} \equiv x_F \end{aligned}$$

e o algoritmo terminou.

Para visualizarmos melhor o conjunto das soluções do problema, poderemos utilizar o seguinte processo gráfico:

Construamos uma arborescência de raiz x_F ; genericamente, cada vértice x_i da arborescência terá como sucessor(es) o(s) vértice(s) indicado(s) na 1.^a parte do(s) seu(s) rótulo(s). Parte-se, portanto, de x_F até atingir x_I . Cada «ramo» da arborescência será um caminho. A solução do problema consiste no conjunto dos «ramos» da arborescência (em sentido inverso).

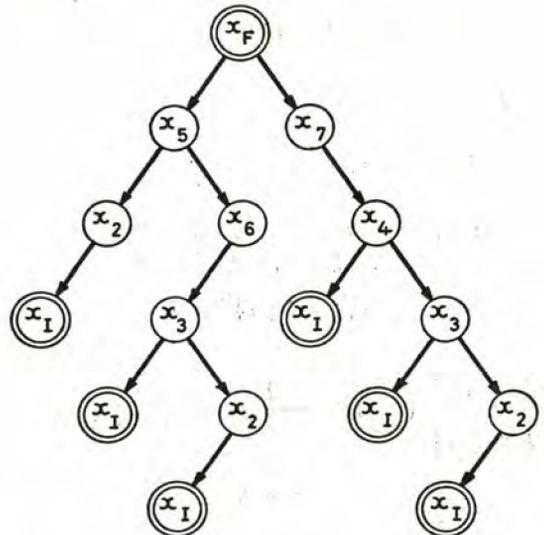


Fig. 10

Solução obtida:

- 1) $[(x_I, x_2), (x_2, x_5), (x_5, x_F)]$
- 2) $[(x_I, x_3), (x_3, x_6), (x_6, x_5), (x_5, x_F)]$
- 3) $[(x_I, x_2), (x_2, x_3), (x_3, x_6), (x_6, x_5), (x_5, x_F)]$
- 4) $[(x_I, x_4), (x_4, x_7), (x_7, x_F)]$
- 5) $[(x_I, x_3), (x_3, x_4), (x_4, x_7), (x_7, x_F)]$
- 6) $[(x_I, x_2), (x_2, x_3), (x_3, x_4), (x_4, x_7), (x_7, x_F)]$

Todos estes caminhos têm o valor total de 7 unidades.

7. Uma extensão natural do problema focado será aquela que conduzirá à determinação do(s) caminho(s) de comprimento mínimo entre um dado vértice inicial e todos os restantes vértices do grafo. As soluções, sujeitas às restrições anteriormente focadas, são determinadas de modo análogo. Há apenas que abrir um parêntesis no que concerne ao ponto onde os processos de cálculo terminam.

Na versão de DIJKSTRA, o algoritmo findará quando todos os vértices pertencerem ao conjunto A ; nas outras versões, quando rótulos definitivos tiverem sido atribuídos a todos os vértices.

*
* .

Cumpre-se sempre com agrado o elementar dever de agradecimento a todos os que, de algum modo, contribuíram para este trabalho.

Para a Fundação Calouste Gulbenkian que o subsidiou, através da concessão de uma bolsa de estudo ao autor; para Amílcar Gonçalves, J. M. Simões Pereira, R. Agonia Pereira e Mário Rosa que o leram e criticaram; e ainda para M. H. Teixeira de Sousa e Manuel Guimarães que, respectivamente, o dactilografaram e elaboraram os desenhos, vão os agradecimentos do autor.

REFERÊNCIAS

- [1] DIJKSTRA, E. W., *A note on two problems in connexion with graphs*, *Numerische Mathematik*, **1**, (1959), pág. 270.
- [2] DREYFUS, S. E., *An appraisal of some shortest paths algorithms*, *Operations Research*, **17**, N.º 3 (1969), pág. 396.
- [3] PRICE, W. L., *Graphs and Networks*, Butterworths, (1971), pág. 46.