Optimização em Redes

Objectivos

– identificar, formular (utilizando modelos de optimização em redes ou de PL) e resolver pelo *Solver*/*Excel* problemas de fluxo de custo mínimo e de caminho mais curto. Identificar e definir o problema da árvore geradora mínima e utilizar o algoritmo de Prim para obter uma solução óptima.

Definições

Grafo ou ***Rede*** é um par ordenado $G=(V,A)$ onde:

$V=\{1,2,\ldots,n\}$ é o conjunto de ***vértices*** ou ***nodos*** ou ***nós***;

$A=\{(a_1,a_2,\ldots,a_m)\}$ é o conjunto de linhas que representam ligações entre os vértices (correspondência de *V* em *V*); $a_k=(i,j),\ i,j\in V$;

Os elementos de *A* com orientação designam-se por ***arcos*** e os elementos de *A* não orientados são ***arestas***.

Uma ***rede*** $G=(V,A)$ diz-se ***orientada*** se todos os elementos de *A* são arcos e diz-se ***não orientada*** se todos os elementos de *A* são arestas. Se *A* é formado por arcos e arestas a ***rede*** diz-se ***mista***.

Um ***arco/aresta*** $(i,j)\in A$ diz-se ***incidente*** em *i* e em *j*.

Se $(i,j)\in A$ for um arco (orientado), então:

i é o ***vértice inicial*** de (i,j); *j* é o ***vértice final*** de (i,j);

j é ***sucessor*** de *i*; *i* é ***antecessor*** (ou ***predecessor***) de *j*.

Se $(i,j)\in A$ for uma aresta (não orientada), então:

i e *j* são as ***extremidades*** de (i,j); *i* e *j* são ***vértices adjacentes***.

Numa rede orientada chama-se ***caminho orientado*** de $x\in V$ para $y\in V$ a uma sequência de arcos distintos $C(x,y)=\{(x,x_1),(x_1,x_2),\ldots,(x_{k-1},x_k),(x_k,y)\}$, de *x* para *y*, onde o vértice final de um arco coincide com o inicial do arco seguinte na sequência e os vértices são todos distintos, com excepção, eventualmente, de *x* e *y*. Se $x=y$ o caminho diz-se um ***circuito***.

Se não se tiver em conta a orientação dos arcos, surgem os conceitos de ***caminho não orientado*** (ou ***cadeia***) e de ***ciclo*** (ou ***circuito não orientado***).

Uma rede diz-se ***conexa*** (***fortemente conexa***) se existe pelo menos uma cadeia (caminho) entre dois quaisquer dos seus vértices.

Uma ***árvore*** é uma rede conexa e sem ciclos. Dada uma rede $G=(V,A)$, uma árvore com o mesmo conjunto de vértices, *V*, e cujo conjunto de arestas esteja contido em *A*, diz-se uma ***árvore geradora*** (ou ***árvore de suporte***) de *G*.

Podem considerar-se diversos parâmetros associados quer aos vértices quer aos arcos e arestas de uma rede (custos, distâncias, tempos, capacidades, ofertas, procuras, pesos, etc.).

Podem ainda ser definidos três tipos de vértices:

- ***origem*** é um nodo gerador de fluxo, onde o fluxo que sai do nodo excede o fluxo que a ele chega;
- ***destino*** representa um nodo que absorve fluxo, onde o fluxo que chega ao nodo excede o fluxo que dele sai;
- ***intermédio*** (***de transfega*** ou ***de transshipment***) é um nodo onde há conservação de fluxo, ou seja, onde o fluxo que sai do nodo iguala o fluxo que a ele chega.

Problema de Fluxo de Custo Mínimo (PFCM)

Dada uma rede orientada, com pelo menos um vértice origem e pelo menos um vértice destino, pretende-se determinar como deve ser feita a passagem de fluxo, respeitando as capacidades dos arcos, as ofertas nas origens, as procuras nos destinos, o equilíbrio nos nodos intermédios e minimizando o custo total associado.

Dados do Problema:

$G=(V,A)$ rede orientada;

Para cada nodo, $i \in V$, b_i é o fluxo gerado no vértice, e:

se $b_i > 0$, i é uma origem com oferta b_i;

se $b_i < 0$, i é um destino com procura $-b_i$;

se $b_i = 0$, i é um ponto intermédio;

Para cada $(i,j) \in A$ são conhecidos:

c_{ij} o custo unitário de passagem de fluxo;

$u_{ij}>0$ a ***capacidade do arco***, ou seja o fluxo máximo que pode passar no arco.

Assume-se que as capacidades dos arcos são compatíveis com a passagem de fluxo das origens para os destinos e que a oferta total iguala a procura total, ou seja: $\sum_{i \in V} b_i = 0$

A formulação do PFCM recorrendo a um modelo de optimização em redes exige a definição da rede associada, $G=(V,A)$, a identificação de todos os parâmetros envolvidos (c_{ij}, u_{ij}, b_i) bem como do problema a resolver.

Definindo x_{ij} como a quantidade de fluxo a passar no arco $(i,j) \in A$, a formulação do PFCM em programação linear é:

$$Min\ Z = \sum_{(i,j) \in A} c_{ij} x_{ij}$$

$$\text{s.a:} \begin{cases} \sum_{j:(i,j) \in A} x_{ij} - \sum_{k:(k,i) \in A} x_{ki} = b_i & \forall i \in V \\ 0 \le x_{ij} \le u_{ij} \quad \forall (i,j) \in A & \end{cases}.$$

Nesta formulação, escreve-se uma restrição para cada vértice ($i \in V$) impondo que a diferença entre o fluxo que sai e o que chega iguala o fluxo nele gerado (b_i). Relativamente aos arcos há que respeitar as capacidades, como se impõe no segundo conjunto de restrições. As variáveis, representando o fluxo que passa em cada arco, são não negativas.

Propriedades do PFCM

Propriedade 1: O PFCM tem pelo menos uma solução admissível.

Corolário: Um PFCM tem solução óptima.

Propriedade 2: Um PFCM com capacidades inteiras nos arcos e valores inteiros para o fluxo gerado em cada vértice, tem pelo menos uma solução óptima em que todas as variáveis assumem valores inteiros.

Resolução do PFCM recorrendo ao *Solver*/*Excel*

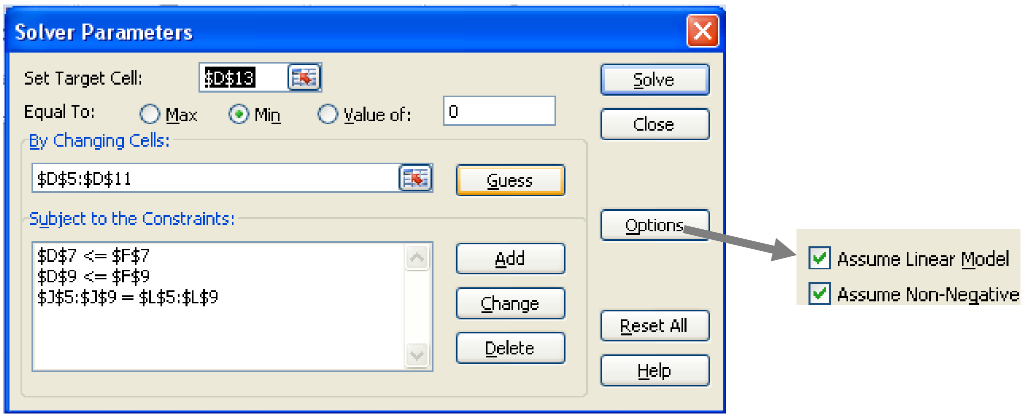
Exemplo – exercício 1.k) (capítulo 1) – numa folha de ***Excel*** é escrito o problema a resolver. Usualmente consideram-se duas tabelas: i) uma com a informação das ligações da rede: arcos (B5-C11), valores de fluxo (D5:D11) inicialmente nulos, capacidades (F5:F11) e custos unitários (G5:G11); e outra ii) com a informação do fluxo em cada vértice da rede: fórmula indicando a diferença entre o "fluxo que sai e o que entra" (J5:J9), e o fluxo gerado (L5:L9). Deve ainda ser escrita a fórmula respeitante à função objectivo (D13).

	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	Distribution Unlimited Co.					(dados)						
3							Custo			Fluxo que		Oferta ou
4		De	Para	Transporte		Capacidade	Unitário		Vértice	"sai"-"entra"		Procura
5		F1	W1	0	<=		$ 900		F1	0	=	50
6		F1	DC	0	<=		$ 400		F2	0	=	40
7		F1	F2	0	<=	10	$ 200		DC	0	=	0
8		F2	DC	0	<=		$ 300		W1	0	=	-30
9		DC	W2	0	<=	80	$ 100		W2	0	=	-60
10		W1	W2	0	<=		$ 300				Total	0
11		W2	W1	0	<=		$ 200					
12												
13			Custo Total	$ -								

	C	D
13	Custo Total	=SUMPRODUCT(D5:D11;G5:G11)

	I	J
3		Fluxo que
4	Vértice	"sai"-"entra"
5	F1	=SUMIF($B$5:$B$11;"F1";$D$5:$D$11)-SUMIF($C$5:$C$11;"F1";$D$5:$D$11)
6	F2	=SUMIF($B$5:$B$11;"F2";$D$5:$D$11)-SUMIF($C$5:$C$11;"F2";$D$5:$D$11)
7	DC	=SUMIF($B$5:$B$11;"DC";$D$5:$D$11)-SUMIF($C$5:$C$11;"DC";$D$5:$D$11)
8	W1	=SUMIF($B$5:$B$11;"W1";$D$5:$D$11)-SUMIF($C$5:$C$11;"W1";$D$5:$D$11)
9	W2	=SUMIF($B$5:$B$11;"W2";$D$5:$D$11)-SUMIF($C$5:$C$11;"W2";$D$5:$D$11)

Solver – Indicação da célula da função objectivo (D13), do objectivo ("Min" ou "Max") e das variáveis de decisão (D5:D11). Junção ("Add") das restrições de capacidade (D5:D11<=F5:F11) e das restrições nos nodos (J5:J9=L5:L9).



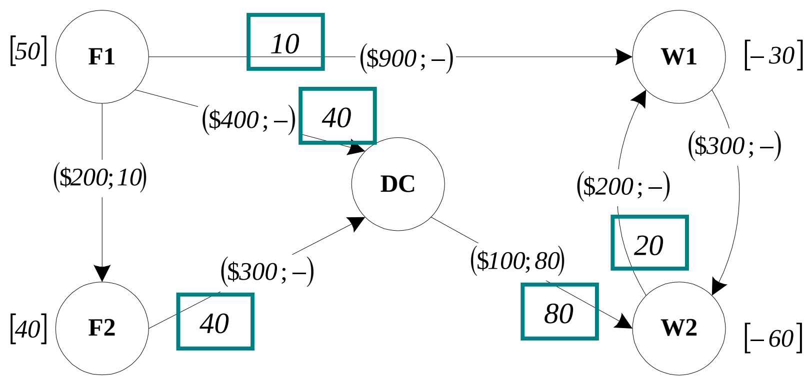
Solução – Interpretação da solução, utilizando o relatório de resposta do *Solver* ou a informação na folha de *Excel* associada à formulação do problema.

Distribution Unlimited Co.

De	Para	Transporte		Capacidade	Custo Unitário		Vértice	Fluxo que "sai"-"entra"		Oferta ou Procura
F1	W1	10	<=		$ 900		F1	50	=	50
F1	DC	40	<=		$ 400		F2	40	=	40
F1	F2	0	<=	10	$ 200		DC	0	=	0
F2	DC	40	<=		$ 300		W1	-30	=	-30
DC	W2	80	<=	80	$ 100		W2	-60	=	-60
W1	W2	0	<=		$ 300				Total	0
W2	W1	20	<=		$ 200					
	Custo Total	$ 49.000								

R: Transportar 10 unidades de F1 directamente para W1 e 40 para o centro de distribuição DC. De F2 enviar 40 unidades para DC. Reencaminhar as 80 unidades recebidas em DC para W2. Reter em W2 60 unidades, das 80 recebidas, e reencaminhar as restantes 20 para W1. O custo total associado a este plano é de $ 49000.

Apresentação da solução na rede: $\boxed{x_{ij}}$



Variantes do PFCM

(1) Oferta Total > Procura Total: $\sum_{i\in V} b_i > 0$

O fluxo gerado nas origens representa um valor máximo a respeitar. Assim as restrições associadas às origens devem ser de tipo "$\leq$".

(2) Oferta Total < Procura Total: $\sum_{i\in V} b_i < 0$

O fluxo gerado nos destinos representa um valor máximo a respeitar. Assim as restrições associadas aos destinos, estando relacionadas com o valor simétrico da procura, devem ser de tipo "$\geq$".

Problema do Caminho Mais Curto (PCMC)

Dada uma rede orientada, com uma só origem e um só destino, pretende-se identificar o caminho de comprimento total mínimo, entre todos os caminhos existentes na rede da origem para o destino.

Dados do Problema:

$G=(V,A)$ rede orientada; $s \in V$ origem; $t \in V$ destino

$c_{ij} > 0$ o comprimento (custo, tempo, distância, etc.) do arco $(i,j)\in A$.

A formulação do PCMC recorrendo a um modelo de optimização em redes exige a definição ou representação gráfica da rede, $G=(V,A)$, a identificação da origem, do destino e dos custos associados aos diferentes arcos, bem como do problema a resolver.

Definindo $x_{ij} = \begin{cases} 1 & \text{se } (i,j) \text{ é usado no caminho} \\ 0 & \text{c.c.} \end{cases}$ $\forall (i,j)\in A$, a formulação do PCMC em programação linear é:

$$Min\ Z = \sum_{(i,j)\in A} c_{ij}x_{ij}$$

$$\text{s.a:} \begin{cases} \sum_{j:(s,j)\in A} x_{sj} = 1 \\ \sum_{i:(i,t)\in A} x_{it} = 1 \\ \sum_{j:(i,j)\in A} x_{ij} - \sum_{k:(k,i)\in A} x_{ki} = 0 & \forall i \in V \setminus \{s,t\} \\ x_{ij} \in \{0,1\} \quad \forall (i,j)\in A \end{cases}.$$

Nesta formulação, as duas primeiras restrições definem os vértices s e t como sendo, respectivamente, a origem e o destino do caminho. O terceiro conjunto de restrições caracteriza os restantes vértices como possíveis vértices do caminho.
Note-se que a definição das variáveis como binárias pode ser substituída pelas restrições de sinal, $x_{ij}\geq 0$. Se forem consideradas unitárias as capacidades dos arcos da rede, este problema pode assim ser visto como um caso particular do problema de fluxo de custo mínimo com uma só origem e um só destino, com oferta e procura unitárias, respectivamente.

Resolução do PCMC recorrendo ao *Solver*/*Excel*

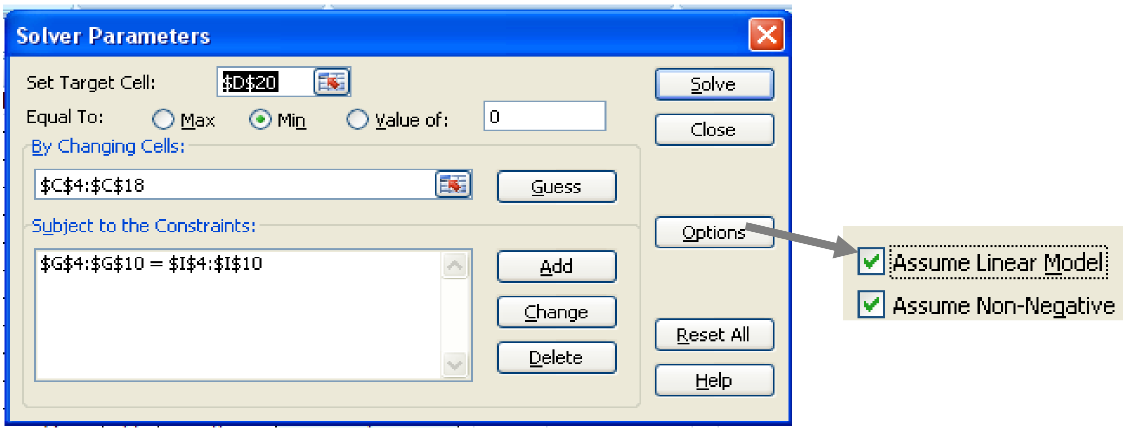
Exemplo Protótipo – *SEERVADA PARK*, 1º problema – numa folha de *Excel* é escrito o problema a resolver. Tal como no PFCM, usualmente consideram-se duas tabelas: uma, i) com a informação das ligações da rede: arcos (A4:B18), valores da solução (C4:C18) inicialmente nulos e distâncias (D4:D18); e outra ii) com a informação em cada vértice da rede (F4:F10): fórmula indicando a diferença entre o "que sai e o que entra" (G4:G10), e o valor que indica se o vértice é origem (I4), destino (I10) ou ponto de eventual passagem (I5:I9). Deve ainda ser escrita a fórmula respeitante à função objectivo (D20).

	A	B	C	D	E	F	G	H	I
1	Seervada Park			dados					
2									Oferta ou
3	de	para	solução	distância		Vértice	total		Procura
4	O	A	0	2		O	0	=	1
5	O	B	0	5		A	0	=	0
6	O	C	0	4		B	0	=	0
7	A	B	0	2		C	0	=	0
8	A	D	0	7		D	0	=	0
9	B	A	0	2		E	0	=	0
10	B	C	0	1		T	0	=	-1
11	B	D	0	4					
12	B	E	0	3					
13	C	B	0	1					
14	C	E	0	4					
15	D	E	0	1					
16	D	T	0	5					
17	E	D	0	1					
18	E	T	0	7					
19									
20		Distância Total		$ -					

	B	C	D
20	Distância Total		=SUMPRODUCT(C4:C18;D4:D18)

	F	G
3	Vértice	total
4	O	=SUMIF($A$4:$A$18;"O";$C$4:$C$18)
5	A	=SUMIF($A$4:$A$18;"A";$C$4:$C$18)-SUMIF($B$4:$B$18;"A";$C$4:$C$18)
6	B	=SUMIF($A$4:$A$18;"B";$C$4:$C$18)-SUMIF($B$4:$B$18;"B";$C$4:$C$18)
7	C	=SUMIF($A$4:$A$18;"C";$C$4:$C$18)-SUMIF($B$4:$B$18;"C";$C$4:$C$18)
8	D	=SUMIF($A$4:$A$18;"D";$C$4:$C$18)-SUMIF($B$4:$B$18;"D";$C$4:$C$18)
9	E	=SUMIF($A$4:$A$18;"E";$C$4:$C$18)-SUMIF($B$4:$B$18;"E";$C$4:$C$18)
10	T	=-SUMIF($B$4:$B$18;"T";$C$4:$C$18)

Solver – Indicação da célula da função objectivo (D20), do objectivo ("Min" ou "Max") e das variáveis de decisão (C4:C18). Junção ("Add") das restrições de caminho nos nodos (G4:G10=I4:I10).



Solução – Interpretação da solução obtida, utilizando o relatório de resposta do *Solver* ou a informação na folha de *Excel* associada à formulação do problema.

	A	B	C	D	E	F	G	H	I
1	**Seervada Park**								
2									Oferta ou
3	de	para	solução	distância		Vértice	total		Procura
4	O	A	1	2		O	1	=	1
5	O	B	0	5		A	0	=	0
6	O	C	0	4		B	0	=	0
7	A	B	1	2		C	0	=	0
8	A	D	0	7		D	0	=	0
9	B	A	0	2		E	0	=	0
10	B	C	0	1		T	-1	=	-1
11	B	D	1	4					
12	B	E	0	3					
13	C	B	0	1					
14	C	E	0	4					
15	D	E	0	1					
16	D	T	1	5					
17	E	D	0	1					
18	E	T	0	7					
19									
20		Distância Total		13					

R: O caminho mais curto de *O* a *T* é (*O*,*A*,*B*,*D*,*T*) e tem distância total *13*.

Problema da Árvore Geradora Mínima (PAGM)

Dada uma rede não orientada e com comprimentos associados às arestas, pretende-se determinar uma árvore geradora (i.e., uma árvore que inclua todos os vértices da rede) mínima, ou seja, com o menor comprimento total.

A imposição de conexidade do grafo associado à solução, pois uma árvore é um grafo conexo sem ciclos, dificulta a sua formulação em programação linear. Assim, neste caso, exige-se apenas a formulação do problema em redes e determina-se a solução óptima do problema, recorrendo ao algoritmo de Prim.

Dados do Problema:

$G=(V,A)$ rede não orientada;

c_{ij} o comprimento (custo, tempo, distância, etc.) da aresta $(i,j)\in A$.

A formulação do PAGM recorrendo a um modelo de optimização em redes exige a definição da rede associada, $G=(V,A)$, a identificação dos custos associados às diferentes ligações, bem como do problema a resolver.

Propriedade 3: Uma árvore geradora, de um grafo com n vértices, tem n nodos e $n-1$ arestas.

Note-se que uma árvore para ser geradora tem de incluir todos os vértices da rede inicial. Por outro lado, para poder conter todos os n vértices sem formar ciclos, tem de incorporar n-*1* arestas. Assim, no algoritmo de Prim são efectuadas n-*1* iterações, sendo, em cada iteração, escolhida a ligação de menor distância que ligue à árvore um nodo que ainda não lhe pertença.

Algoritmo de Prim (1957)

Objectivo da iteração *k*

Escolher entre os vértices que não pertencem à árvore o que lhe está mais próximo. Ligar esse vértice à árvore utilizando a ligação que tenha associada essa distância mínima.

Repetir até que todos os vértices estejam incluídos na árvore.

Algoritmo

0. Dados: Rede $G=(V,A)$ não orientada, conexa, com n vértices;
Comprimentos das arestas;

1. Inicialização

Escolher um qualquer vértice e a aresta de menor comprimento nele incidente; Inicializar a árvore com a aresta seleccionada e os dois vértices respectivos;
$k \leftarrow 2;$

2. Iteração *k*

Se todos os vértices pertencem à árvore **Ir para 3.**

C.c., seleccionar a aresta de comprimento mínimo, entre as arestas incidentes num vértice da árvore e noutro que não lhe pertença;

Juntar à árvore a aresta seleccionada e o vértice em que esta incide e que não pertence à árvore;

$k \leftarrow k+1;$

Voltar a 2..

3. Desenhar a árvore geradora mínima e determinar o respectivo comprimento total. **FIM.**

Notas: Em caso de empate, no passo **2.**, a escolha é arbitrária, só se escolhendo uma aresta em cada iteração.

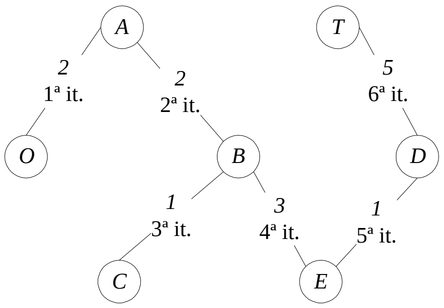
Resolução do PAGM recorrendo ao algoritmo de Prim

Exemplo Protótipo – *SEERVADA PARK*, 2º problema – rede com $n=7$ vértices $\Rightarrow$ $n\text{-}1=6$ iterações.

Iteração	Nodo na árvore	Nodo adjacente $\notin$ árvore + perto	Distância da ligação	Ligação a juntar
1	*O*	*A*	*2*	(*O*,*A*)
2	*O*	*C*	*4*	(*A*,*B*)
	A	*B*	***2*** ←	
3	*O*	*C*	*4*	(*B*,*C*)
	A	*D*	*7*	
	B	*C*	***1*** ←	
4	*O*	-		(*B*,*E*)
	A	*D*	*7*	
	B	*E*	***3*** ←	
	C	*E*	*4*	
5	*O*	-		(*E*,*D*)
	A	*D*	*7*	
	B	*D*	*4*	
	C	-		
	E	*D*	***1*** ←	
6	*D*	*T*	***5*** ←	(*D*,*T*)
	E	*T*	*7*	

Legenda: ← ligação de menor distância.

Árvore geradora mínima



A distância total da AGM é *14*

Nota: Na última iteração, 6, como só os vértices *D* e *E*, na árvore, estão ligados ao único que ainda não pertence à árvore, *T*, não se consideraram os restantes nodos na tabela.