Universidade Do Minho
Engenharia Informática
Investigação Operacional
A100533 - António Silva; A100600 - Diogo Barros; A100745 - Pedro Silva

Trabalho Prático 2

06/05/2023

1 - Formulação do Problema

Neste trabalho, pretende-se atribuir serviços a clientes que serão distribuídos por equipas. O objetivo a cumprir é a minimização do custo total da operação, que inclui custos de deslocação e custos fixos de utilização de veículos.

Cada cliente tem associado a si uma hora de início de servico **aj**.

Uma equipa só pode efectuar o serviço do cliente **j** se, após terminar o serviço de um cliente anterior, puder chegar ao cliente **j** num instante igual ou anterior a **aj**. Se chegar mais cedo, tem de aguardar pela hora de início do serviço.

Tendo em conta que o maior nº de aluno do grupo é **100745**(xABCDE), obtivemos a seguinte tabela.

j	CLIENTE	a_j (¼ HORA)	a_j (HORA DE SERVIÇO)
1	ANA	1	09:15
2	BEATRIZ	7	10:45
3	CARLOS	4	10:00
4	DIOGO	2	09:30
5	EDUARDO	10	11:30
6	FRANCISCA	6	10:30
7	GONÇALO	9	11:15
8	HELENA	8	11:00
9	INÊS	2	09:30
10	JOSÉ	5	10:15

Figura 1 - Tabela de horas de serviço

NOTA:

- como o B é igual a 0, a_j do cliente A é 1 (B+1).
- como o C é igual a 7, a_j do cliente H é 8 (C+1).
- como o D é par (D = 4), este cliente é retirado, por isso se encontra "sublinhado".
- como o E é ímpar (E = 5), este cliente não é retirado.

	B	C	D	E	F	G	H	I	J	K
A	4	1	2	2	3	2	1	0	3	1
B		3	5	3	3	2	3	4	2	5
C			3	2	3	2	0	1	1	2
D				1	3	3	3	2	3	1
E					2	1	2	2	2	2
F						2	3	3	3	4
G							2	2	2	3
H								1	1	1
I									3	2
J										4

tempos de deslocação

	B	C	D	E	F	G	H	I	J	K
A	13	5	6	5	10	7	5	0	7	1
B		11	14	10	8	6	11	13	4	15
C			8	6	10	6	0	5	6	2
D				4	8	8	8	6	11	4
E					6	4	6	5	7	6
F						5	10	10	8	11
G							10	7	5	9
H								5	6	9
I									7	9
J										10

custos de deslocação

Figura 2 - Tabelas de tempos e custos de deslocação

Com a análise das tabelas das **Figuras 1 e 2**, obtemos sempre 3 valores para se aplicar na fórmula para verificar se um caminho é possível ou não. Os valores que se obtém são o $\mathbf{a_i}$ e o $\mathbf{a_j}$, da Figura 1, o $\mathbf{t_{ij}}$, da tabela de tempos de deslocação e o $\mathbf{d_i}$, que como foi referido no enunciado, tem sempre duração de $\frac{1}{4}$ de hora para todos os clientes.

Com estes valores, obtemos a fórmula

$$a_i + d_i + t_{ij} \leq a_j$$

, que verifica se uma equipa pode efectuar o serviço do cliente j, após o fim do serviço do cliente i, e se consegue chegar ao cliente j num instante igual ou anterior a $\mathbf{a_j}$ a respeitar é cumprido.

Com a verificação de que as regras e a fórmula foram respeitadas, conseguimos assim selecionar os caminhos possíveis que uma equipa pode fazer, e acabamos por obter um grafo de compatibilidades e uma tabela que mostra as deslocações possíveis entre clientes (**Figuras 3 e 4**).

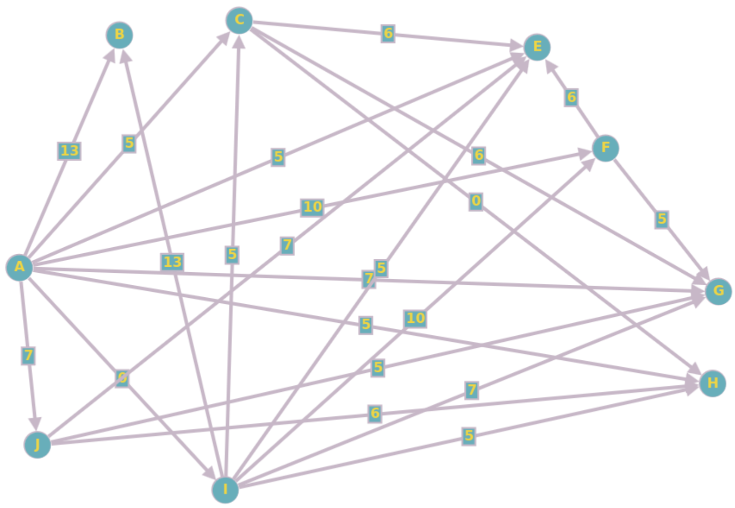

Figura 3 - Grafo de compatibilidades

No grafo de compatibilidades resultante das verificações feitas anteriormente, obtemos todas as rotas possíveis de um cliente i para um cliente j. Nos vértices, temos todos os clientes, entre estes temos ligações/fluxos com um determinado custo.

Por exemplo: o vértice **A** tem ligação com o vértice **B**, no sentido de **A** para **B**, com um custo de deslocação/fluxo igual a 13.

	A	B	C	D	E	F	G	H	I	J	K
A		13	5		5	10	7	5	0	7	2
B											16
C					6		6	0			3
D											
E											7
F					6		5				12
G											10
H											10
I		13	5		5	10	7	5			10
J					7		5	6			11
K	1	15	2		6	11	9	9	9	10	

Figura 4 - Tabela dos custos de deslocação dos caminhos possíveis.

Na tabela da **Figura 4**, a existência de um caminho no grafo é indicada pela existência de um número a fazer corresponder dois vértices. Por exemplo, a primeira linha e segunda coluna, indica que é possível ser feito o caminho do cliente A para o cliente B, tendo em conta as restrições temporais, e que o custo dessa deslocação é de 13 unidades monetárias.

Na coluna K, que simboliza a sede da empresa, os valores das distâncias estão incrementados em 1 U.M., que representa quando as equipas regressam à sede da perspetiva de cada cliente.

2 - Modelo de problema de fluxos em rede

Para resolver este problema, podemos modelar este como um problema de fluxo de redes, com o objetivo de encontrar o fluxo de custo mínimo na rede que definimos com os caminhos possíveis, dando-nos assim a melhor atribuição de equipas a todos os clientes, de modo a que o custo seja o menor possível.

Para encontrar o fluxo de custo mínimo da rede e assegurar que todos os vértices/clientes são percorridas por uma equipa, dividimos cada vértice em dois, sendo divididos em vértices-origem, que são os vértices do lado direito (A1,B1,...,J1) e em vértices-destino, que são os do lado esquerdo (A,B,...,J).

Todas as arestas do grafo saem de um vértice-origem e entram num vértice-destino.

NOTA: Os 2 **K** representam a sede da empresa (Keleirós), ou seja, representam tanto a origem/início do dia, quando iniciam o dia de trabalho, como também representam o destino final, ou seja quando terminam o último deslocamento do último cliente.

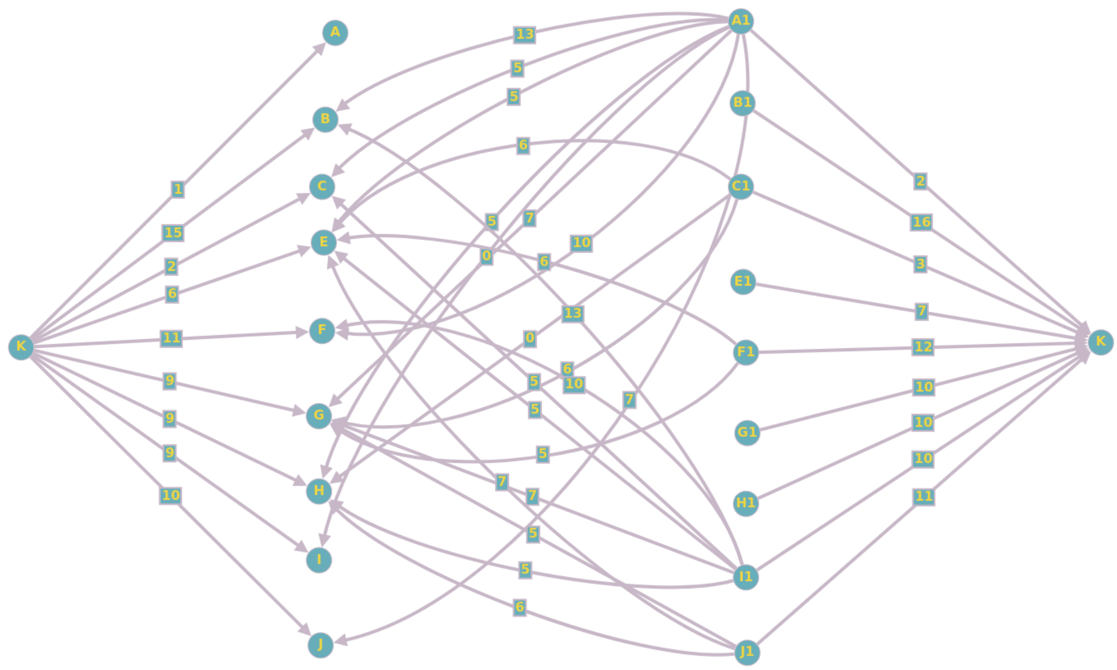

Figura 5 - Grafo bipartido do fluxo de rede.

Explicação de alguns caminhos:

- O vértice-destino **B** é o destino do deslocamento do vértice **K do lado esquerdo**, com um **custo de deslocação de 15**, quando este pretende atender as necessidades do cliente **B**.
- O vértice-destino **B** é o destino do deslocamento do vértice-origem **A1**, com um **custo de deslocação de 13**, quando este pretende atender as necessidades do cliente **B**, ao fim de este atender as necessidades do cliente **A**.
- O vértice-origem **A1** é a origem do deslocamento para o vértice-destino **K do lado direito**, com um **custo de deslocação de 2**, quando este ao fim de cuidar das necessidades do cliente **A**, pretende regressar à sede **Keleirós(K)**.

Com o grafo modulado para o nosso problema, passamos este projeto para o software de otimização em rede Relax4. Para isso, tivemos de "traduzir" o grafo, num ficheiro de texto, que o software Relax4 consiga interpretar, para nos dar a solução.

```
19
40
1 11 13 1
1 12 5 1
1 13 5 1
1 14 10 1
1 15 7 1
1 16 5 1
1 17 0 1
1 18 7 1
3 13 6 1
3 15 6 1
3 16 0 1
5 13 6 1
5 15 5 1
8 11 13 1
8 12 5 1
8 13 5 1
8 14 10 1
8 15 7 1
8 16 5 1
9 13 7 1
9 15 5 1
9 16 6 1
```

```
19 10 1 1
19 11 15 1
19 12 2 1
19 13 6 1
19 14 11 1
19 15 9 1
19 16 9 1
19 17 9 1
19 18 10 1
1 19 2 1
2 19 16 1
3 19 3 1
4 19 7 1
5 19 12 1
6 19 10 1
7 19 10 1
8 19 10 1
9 19 11 1
```

```
1
1
1
1
1
1
1
1
1
-1
-1
-1
-1
-1
-1
-1
-1
-1
0
```

Figura 6 - Transformação do grafo para um ficheiro input (.txt) para o Relax4.

Nas primeiras duas linhas definimos o número de vértices (1ª linha) e o número de arestas/ligações (2ª linha). Entre os 19 vértices (1ºlinha), existem 40 (2ºlinha) arestas, que correspondem aos deslocamentos entre clientes.

Os vértices-origem (A1,B1,...,J1) são numerados no ficheiro de .txt de 1 até 9, e os vértices-destino (A,B,...,J) são numerados no ficheiro de .txt de 10 até 18, e o vértice (K) está numerado como 19.

A capacidade de todas as arestas (último elemento de uma linha do ficheiro .txt) é definida com o valor 1 para assegurar que um cliente só é atendido por uma só equipa.

Explicação da linha com os elementos (1 11 13 1) do ficheiro .txt:

- o 1º valor, **1**, indica o vértice-origem, **A1**.
- o 2º valor, **11**, indica o vértice-destino, **B**
- o 3º valor, **13**, indica o custo de deslocação de, **A1 para B**.
- o 4º valor, **1**, que indica a capacidade de todas as arestas, explicada em cima.

A imagem mais à direita da **Figura 6**, define a oferta/procura de cada vértice, tendo os **vértices-origem (1 a 9, ou seja, A1 a J1)** tem uma **oferta igual a 1**, sendo definida pelo **valor 1**, e os **vértices-destino (10 a 18, ou seja, A a J)** tem uma **procura igual a 1**, sendo definida pelo **valor -1**, com o vértice que representa a **sede da empresa com valor 0**, porque não tem **nem oferta nem procura**.

3 - Solução ótima dada pelo software

Depois de submetermos o ficheiro .txt de input no Relax4, obtivemos os seguintes resultados.

```
C:\relax4>relax4 <Relax4.txt
 END OF READING
 NUMBER OF NODES = 19, NUMBER OF ARCS = 40
 CONSTRUCT LINKED LISTS FOR THE PROBLEM
 CALLING RELAX4 TO SOLVE THE PROBLEM
 ***********************************
 TOTAL SOLUTION TIME =  0. SECS.
 TIME IN INITIALIZATION =  0. SECS.
   1 17  1.
   3 16  1.
   5 13  1.
   8 11  1.
   9 15  1.
   19 10  1.
   19 12  1.
   19 14  1.
   19 18  1.
   2 19  1.
   4 19  1.
   6 19  1.
   7 19  1.
 OPTIMAL COST =   91.
 NUMBER OF AUCTION/SHORTEST PATH ITERATIONS = 31
 NUMBER OF ITERATIONS =  19
 NUMBER OF MULTINODE ITERATIONS =  2
 NUMBER OF MULTINODE ASCENT STEPS =  2
 NUMBER OF REGULAR AUGMENTATIONS =  2
 ***********************************
```

Figura 7 - Solução/Output do ficheiro .txt do input no Relax4

A partir do output fornecido pelo Relax4 da **Figura 7**, podemos verificar que a solução ótima tem um custo de deslocação de 91 U.M. e para que este custo seja atingido, são utilizadas 4 equipas para atender os vários clientes, pois saem 4 unidades de fluxo do vértice K (valor 19 do Relax4), ou seja da sede da empresa, em Keleirós.

A seguinte tabela **(Figura 8)** e grafo **(Figura 9)** apresentam os percursos de cada equipa, a hora de começo, o tempo de deslocação, o custo de deslocação para cada cliente, e por último, o custo total de cada uma das equipas.

Equipas	Cliente	Hora	Tempo de Deslocação	Custo	Custo Total
Equipa 1	Keleirós	9:00	[KA] ¼ Hora	1	30
	Ana	9:15	[AI] 0 Horas	0	
	Inês	9:30	[IB] 1 Hora	13	
	Beatriz	10:45	[BK] 1 Hora ¼	16	
	Keleirós	12:15	-	-	
Equipa 2	Keleirós	9:00	[KC] ½ Hora	2	12
	Carlos	10:00	[CH] 0 Horas	0	
	Helena	11:00	[HK] ¼ Hora	10	
	Keleirós	11:30	-	-	
Equipa 3	Keleirós	9:00	[KF] 1 Hora	11	24
	Francisca	10:30	[FE] ½ Hora	6	
	Eduardo	11:30	[EK] ½ Hora	7	
	Keleirós	12:15	-	-	
Equipa 4	Keleirós	9:00	[KJ] 1 Hora	10	25
	José	10:15	[JG] ½ Hora	5	
	Gonçalo	11:15	[GK] ¾ Hora	10	
	Keleirós	12:15	-		

CUSTO TOTAL: 30+12+24+25 = 91

Figura 8 - Representação da solução ótima numa tabela.

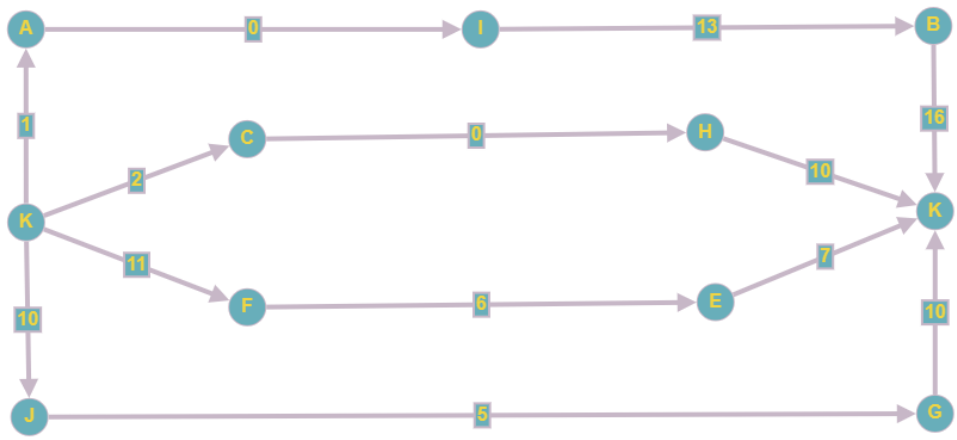

Figura 9 - Representação da solução ótima num grafo.

4 - Validação do Modelo

Para começar a nossa validação do modelo, tivemos de verificar se as nossas soluções iniciais eram válidas perante as restrições de horário de atendimento dos clientes, se todas as equipas faziam um percurso apropriado (sair da sede, atender clientes e voltar à sede) e se todos os clientes eram satisfeitos. Como demonstramos no início do relatório, com o grafo de compatibilidades **Figura 3** e a tabela dos tempos/custos de deslocação **Figura 4**, verificamos que todas essas restrições se tornaram válidas.

O próximo passo para a validação do projeto foi a verificação da solução ótima, e para que isso fosse possível, decidimos modelar o problema como um problema de programação linear, usando o software LPSolve. A função objetivo neste software consiste na minimização do custo de deslocação dos caminhos percorridos pelas equipas durante o atendimento aos seus clientes. As variáveis de decisão usadas são praticamente todas binárias, salvo algumas exceções, e indicam se uma equipa passou em dois clientes.

```
/* Objective function */
min: 13AB + 5AC + 5 AE + 10AF + 7AG + 5AH + 0AI + 7 AJ +
6CE + 6CG + 0CH +
6FE + 5FG +
13IB + 5IC + 5IE+ 10IF + 7IG + 5IH +
7JE + 5JG + 6JH +
1KA + 15KB + 2KC + 6KE + 11KF + 9KG + 9KH + 9KI + 10KJ +
2AK + 16BK + 3CK + 7EK + 12FK + 10GK + 10HK + 10IK + 11JK;

/* Variable bounds */
KA = 1;
AB + AC + AE + AF + AG + AH + AI + AJ + AK = 1;
AB + IB + KB = 1;
BK = 1;
AC + IC + KC = 1;
CE + CG + CH + CK = 1;
AE + CE + FE + IE + JE + KE = 1;
EK = 1;
AF + IF + KF = 1;
FE + FG + FK = 1;
AG + CG + FG + IG + JG + KG = 1;
GK = 1;
AH + CH + IH + JH + KH = 1;
HK = 1;
AI + KI = 1;
IB + IC + IE + IF + IG + IH + IK = 1;
AJ + KJ = 1;
JE + JG + JH + JK = 1;
KA + KB + KC + KE + KF + KG + KH + KI + KJ = AK + BK + CK + EK + FK + GK + HK + IK + JK;

bin AB, AC, AE, AF, AG, AH, AI, AJ,
CE, CG, CH,
FE, FG,
IB, IC, IE, IF, IG, IH,
JE, JG, JH,
KB, KC, KE, KF, KG, KH, KI, KJ,
AK, CK, FK, IK, JK;
```

Figura 10 - Input para o software do LPSolve.

Explicação de algumas restrições do LPSolve:

- as restrições do tipo **KA = 1**, servem para evitar colocar este tipo de rotas no bin, porque faz com que o seu valor possa variar entre 1 ou 0 e o custo resultante desta alteração pode não ser o correto, para o modelo.
- as restrições com somas, do tipo **AB + IB + KB = 1**, pretendem demonstrar ao LPSolve, que estas somas de deslocamentos são válidos, mas ao mesmo tempo nem todos podem ser escolhidos e vai ser da competência do programa escolher as restrições deste tipo que trazem menor custo de deslocação.
- a última restrição que foi proposta ao programa, vai obrigar ao software, a respeitar uma regra crucial, em que o número de equipas que vão sair da sede da empresa, tem de ser iguais ao número de equipas que regressam a esta mesma sede.

Depois de todas as restrições serem aplicadas ao software, o LPSolve teve a **Figura 11** como output do problema:

Variables	M...	result
	91	91
KJ	1	1
KF	1	1
KC	1	1
KA	1	1
JG	1	1
IB	1	1
HK	1	1
GK	1	1
FE	1	1
EK	1	1
CH	1	1
BK	1	1
AI	1	1
KI	0	0
KH	0	0
KG	0	0
KE	0	0
KB	0	0
JK	0	0
JH	0	0
JE	0	0
IK	0	0
IH	0	0
IG	0	0
IF	0	0
IE	0	0
IC	0	0
FK	0	0
FG	0	0
CK	0	0
CG	0	0
CE	0	0
AK	0	0
AJ	0	0
AH	0	0
AG	0	0
AF	0	0
AE	0	0
AC	0	0
AB	0	0

Figura 11 - Output do LPSolve.

Como obtido no Relax4, também o LPSolve obteve o mesmo custo mínimo de deslocação, 91 U.M. e obteve também as mesmas 4 equipas, calculadas pelo Relax4, para atingir a solução ótima. Além disso, apesar de não ser necessário, neste caso também obteve as mesmas rotas.

Equipas:

- **K - A - I - B - K**;
- **K - C - H - K**;
- **K - F - E - K**;
- **K - J - G - K**;

5 - Conclusão

Depois de termos utilizado este método de validação para a solução ótima(LPSolve), podemos concluir com quase completa certeza que 91 U.M deve ser a melhor solução possível. Na nossa situação, as rotas do Relax4 e do LPSolve foram exatamente iguais, mas isso poderia não ser o caso. Caso isso não acontecesse, o que realmente importa é que o custo monetário seja igual.

Podemos visualizar na resolução deste problema, que um problema de fluxo de redes pode facilmente ser transposto num problema de Programação Linear, apesar de se calhar não seja a forma mais fácil de o visualizar como com o grafo de fluxo de redes.

Em conclusão, com este exercício podemos compreender o uso dum grafo de compatibilidades, dum grafo de fluxo de redes, da utilização da aplicação Relax4, compreensão da solução dada e como adaptar este exercício a um de Programação Linear e utilizar o LPSolve para validar a solução ótima.