Problemas de Fluxo Máximo

Transparências de apoio à leccionação de aulas teóricas

Slide 1

José Fernando Oliveira

Problemas de Fluxo Máximo

Definição: Dada uma rede, com um *nó de entrada* e um *nó de saída*, com *capacidades* associadas a cada ramo, pretende-se saber qual é o fluxo máximo, de um certo bem, que se pode enviar da entrada para a saída.

Modelo:

Slide 2

x_{ij} – fluxo que passa no ramo (i,j), de i para j
c_{ij} – capacidade do ramo (i,j)
Nó 1 – Nó de entrada
Nó t – Nó de saída

$$\max \sum_j x_{1j} = \sum_i x_{it}$$

suj. a:

$$\begin{aligned} \sum_i x_{ik} &= \sum_j x_{kj}, \forall_k && \text{(equilíbrio de fluxos nos nós)} \\ x_{ij} &\leq c_{ij}, \forall_{i,j} && \text{(restrições de capacidade)} \\ x_{ij} &\geq 0, \forall_{i,j} \end{aligned}$$

Algoritmo de fluxo máximo

Slide 3

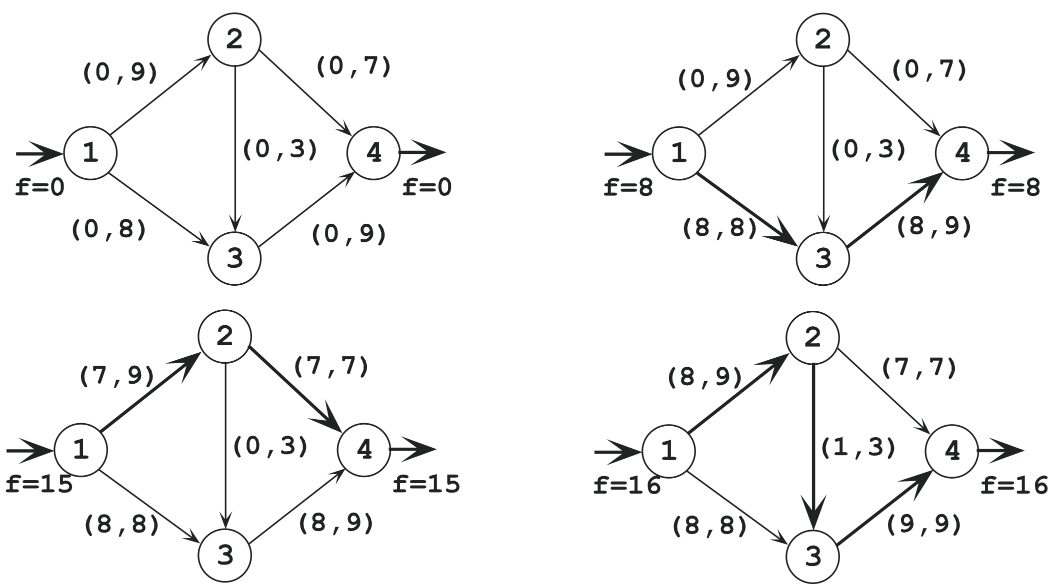
1. injectar um fluxo nulo no nó de entrada;
2. capacidades iniciais dos ramos = capacidade total dos ramos;
3. determinar um *caminho não saturado* (capacidade $\neq 0$) entre o nó de entrada e o nó de saída; se não existir, foi encontrada a SOLUÇÃO ÓPTIMA;
4. somar ao fluxo de entrada um fluxo igual à *capacidade do caminho* seleccionado;
5. alterar as capacidades dos ramos do caminho seleccionado, diminuindo-lhes o fluxo injectado;
6. voltar ao ponto 3.

Notas:

- **caminho** – conjunto de ramos, unindo o nó de entrada ao nó de saída e que não passa duas vezes pelo mesmo nó;
- **capacidade de um caminho** – menor capacidade disponível de entre todos os ramos que fazem parte do caminho;
- **caminho saturado** – caminho com capacidade nula;

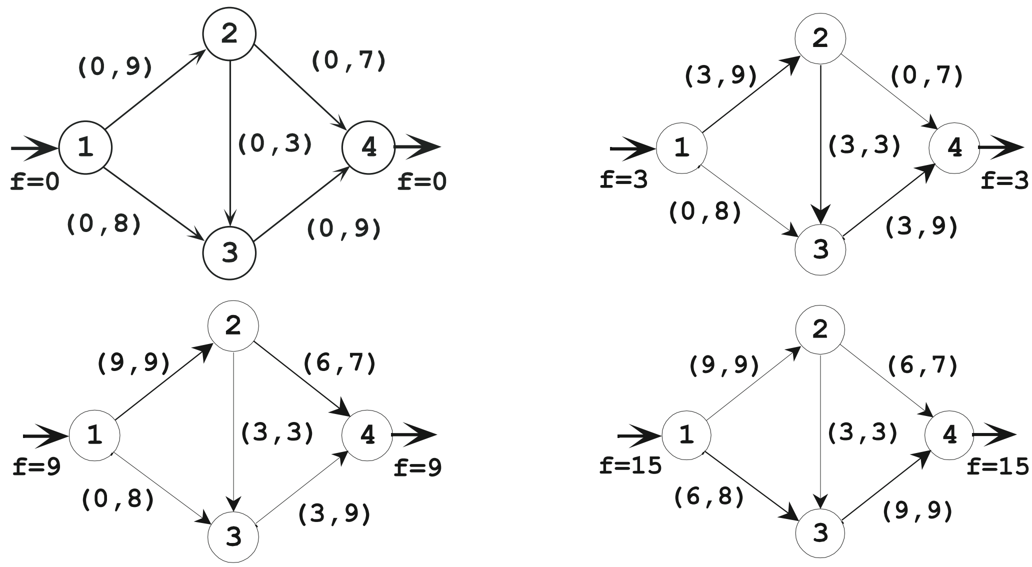
Exemplo de um problema de fluxo máximo

Slide 4



E se os caminhos fossem saturados por outra ordem?

Slide 5



Aparentemente não há nenhum caminho não saturado entre a entrada e a saída e o fluxo "máximo" deu menor do que no caso anterior...

Não é este um algoritmo de optimização?

Fluxos "negativos"

Slide 6

De facto o exemplo anterior não está completamente resolvido (até à optimalidade) porque existe ainda um caminho não saturado. Para se encontrar este caminho não saturado é necessário considerar o conceito de fluxo "negativo", isto é, fluxo que atravessa os ramos no sentido contrário à sua orientação.

Uma das restrições apresentadas no modelo do problema de fluxo máximo impunha que todos os fluxos fossem positivos ou nulos ($x_{ij} \geq 0$). E de facto, na solução final tal terá sempre que acontecer. No entanto, essa solução final obtém-se pela adição dos vários fluxos que vamos injectando na rede. Alguns desses fluxos poderão atravessar algum ramo no sentido contrário ao indicado, devendo nesse caso ser contabilizados como negativos. O resultado final (soma de todos os fluxos que foram injectados nesse ramo) é que terá que ser positivo ou nulo. No fundo um fluxo "negativo" mais não é do que deixar de fazer passar fluxo por esse ramo.

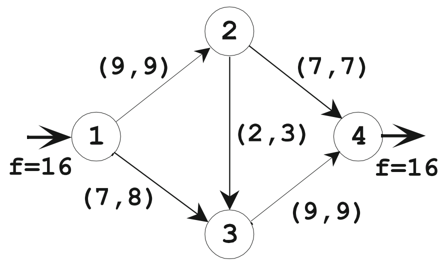
Fluxos "negativos"

Observando a rede deste ponto de vista conclui-se que o caminho 1-3-2-4 não está saturado:

- O ramo 1-3 tem uma capacidade de 2.
- O ramo 2-3, visto do lado do nó 2, está saturado. No entanto visto do lado do nó 3, que é o lado que nos interessa visto o caminho o precorrer de 3 para 2, tem uma capacidade de 3, que é o fluxo que o atravessa de 2 para 3.
- O ramo 2-4 tem uma capacidade de 1.

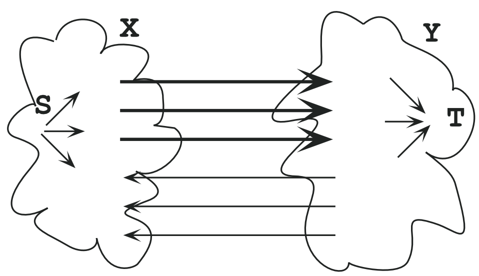
Slide 7

Então a capacidade do caminho é 1 e é possível injectar mais uma unidade de fluxo na rede:



Consegue-se ter a certeza se uma rede está na situação de fluxo máximo através da relação entre fluxo máximo e cortes mínimos.

Fluxos máximos e cortes mínimos



Definição: Um **corte** numa rede com nó de entrada S e nó de saída T é um conjunto de arcos cuja remoção separa a rede em duas partes, X e Y, uma contendo S e outra contendo T.

Slide 8

A **capacidade de um corte** é a soma das capacidades dos arcos do corte, que estão dirigidos de X para Y. Um **corte mínimo** é um corte com a menor capacidade possível.

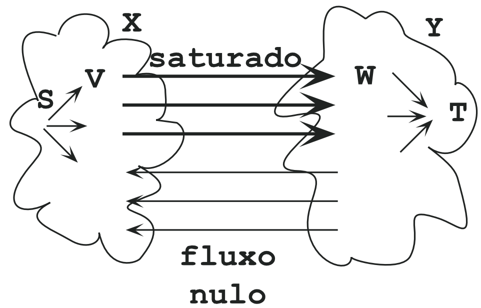
$$\begin{array}{rcl}
\text{Valor de qualquer fluxo} & \leq & \text{Capacidade de qualquer corte} \\
\text{Valor do fluxo máximo} & \leq & \text{Capacidade de qualquer corte} \\
\text{Valor do fluxo máximo} & \leq & \text{Capacidade de um corte mínimo}
\end{array}$$

Teorema do Fluxo Máximo–Corte Mínimo

Teorema: O valor do fluxo máximo é igual à capacidade do corte mínimo.

Slide 9

Demonstração: Seja uma rede na situação de fluxo máximo. Seja X o conjunto de nós que pode ser atingido a partir de S através de um caminho não saturado, e seja Y o conjunto dos restantes nós. $T \in Y$ porque senão $T \in X$ e não se estava numa situação de fluxo máximo.



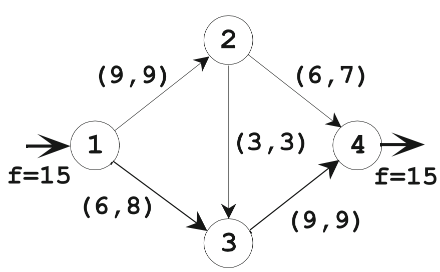
Considere-se o corte composto pelos ramos com uma extremidade em X e outra em Y. Todos os arcos dirigidos de um nó V em X para um nó W em Y estão saturados, pois caso contrário W pertenceria a X e não a Y. Qualquer arco dirigido de Y para X terá fluxo nulo pois caso contrário isso seria um retorno de fluxo de Y para X que, se fosse anulado, aumentaria o fluxo que efectivamente chega a T, o que contraria a hipótese inicial de a rede estar na situação de fluxo máximo. Então, a capacidade do corte, que é igual ao fluxo de X para Y, é igual ao fluxo na rede, que é máximo.

Aplicação de cortes a redes de fluxos

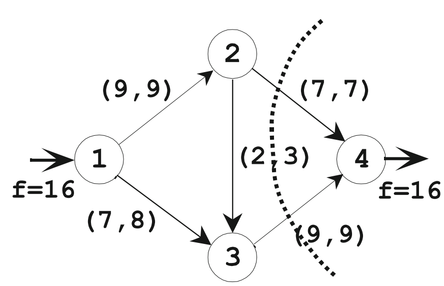
Uma rede está então na situação de fluxo máximo se existir um corte mínimo, isto é que separa a entrada da saída da rede e que só atravessa ramos orientados da entrada para a saída saturados e ramos orientados da saída para a entrada com fluxo nulo.

Slide 10

Nesta rede não existe um corte com essas propriedades.



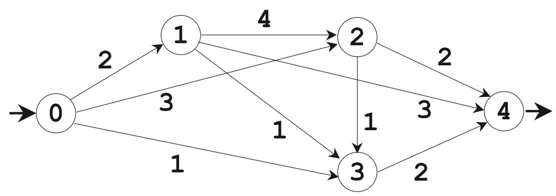
O corte constituído pelos ramos 2-4 e 3-4 é um corte mínimo.



Exercício

Slide 11

Determine a quantidade máxima de um produto que pode ser enviada através da rede seguinte, entre a origem e o destino 4. Existem limitações nas quantidades que podem atravessar cada arco, encontrando-se as respectivas quantidades máximas representadas junto a cada arco. Considere que esse produto se encontra disponível na origem 0 em quantidade ilimitada.

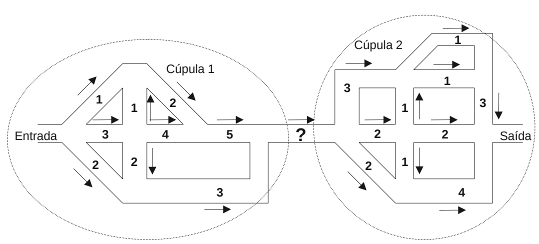


Em que arco(s) aumentaria as quantidades máximas admissíveis, de forma a conseguir aumentar a quantidade máxima de produto que é possível fazer passar pela rede?

Exercício

Slide 12

Uma parte do ShopShopping vai ser construída imitando uma plataforma de exploração subaquática: duas grandes cúpulas ligadas por um grande corredor. Para que a circulação de pessoas no centro comercial decorra de uma forma fluida, é necessário que este corredor não restrinja o fluxo máximo que pode atravessar a secção subaquática do centro comercial. Na figura seguinte representa-se esquematicamente a planta desta parte do ShopShopping.



Em cada corredor está indicada a capacidade (em dezenas de pessoas por minuto) de circulação nesse corredor. O corredor de ligação entre as duas cúpulas está ainda por dimensionar, dado o seu elevado custo, crescente com o aumento de capacidade que se lhe queira atribuir. Note que por questões de segurança e fluidez de circulação os corredores funcionam como caminhos de sentido único (ver setas na figura).

Resolvendo este problema como de fluxo máximo indique qual deve ser a capacidade do corredor de ligação de forma a que o fluxo que atravessa as cúpulas seja máximo e o custo do corredor de ligação o menor possível.

Problemas de Caminho Mínimo

Transparências de apoio à leccionação de aulas teóricas

Slide 13

José Fernando Oliveira

Problemas de Caminho Mínimo

Definição: Determinar o caminho mais curto entre o nó de entrada e o nó de saída de uma rede. A cada ramo (i, j) está associada uma "distância" não negativa d_{ij}. É admissível que $d_{ij} \neq d_{ji}$.

E é um problema de fluxos? Basta reinterpretá-lo como um problema de enviar uma unidade de um bem, do nó de entrada para o nó de saída, minimizando o custo de transporte, em que d_{ij} representa o custo unitário de transporte de i para j.

Slide 14

Pode ser formulado, e resolvido, como um problema de PL, mas...

Algoritmo de Dijkstra – metodologia

Atribuir a todos os nós uma etiqueta, que poderá ser:

- provisória – limite superior para a distância mais curta entre o nó de entrada e o nó em causa (menor distância conhecida até ao momento);
- definitiva – distância mais curta entre o nó de entrada e o nó em causa.

Slide 15

Inicialmente é atribuída ao nó de entrada uma etiqueta definitiva com o valor 0. A todos os outros nós é atribuída uma etiqueta provisória com valor igual a ∞.

O algoritmo tenta transformar as etiquetas provisórias em definitivas. Quando o nó de saída tiver uma etiqueta definitiva o problema foi resolvido.

Algoritmo de Dijkstra

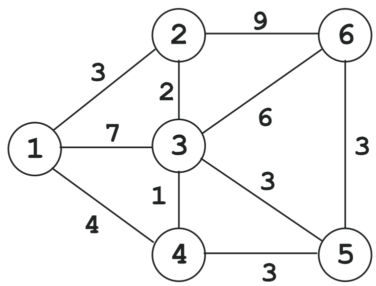
Slide 16

1. atribuir uma etiqueta definitiva ao nó de entrada, com valor 0, e etiquetas provisórias a todos os outros nós, com valor igual a ∞;
2. seja k o nó que mais recentemente recebeu uma etiqueta definitiva. Para todos os nós i ainda com etiquetas provisórias calcule-se o valor da etiqueta definitiva de k mais a distância **directa** entre k e i (d_{ki}). O mínimo, entre este valor e a anterior etiqueta provisória do nó i, é tomado como nova etiqueta provisória de i.
3. seleccionar a menor das etiquetas provisórias e declará-la permanente. Se fôr o nó de saída, determinou-se a distância mínima e termina o algoritmo, senão volta ao ponto 2.

Para determinar a sequência de nós que forma o caminho com distância mínima, deve-se, retrocedendo a partir do nó de saída, procurar os nós com etiquetas permanentes cuja diferença é igual à distância associada ao arco que os une.

Aplicação do algoritmo de Dijkstra

Slide 17



E_i – vector de etiquetas na iteração i
$E_i[j]$ – etiqueta do nó $j, j \in \{1, \ldots, n\}$ na iteração i

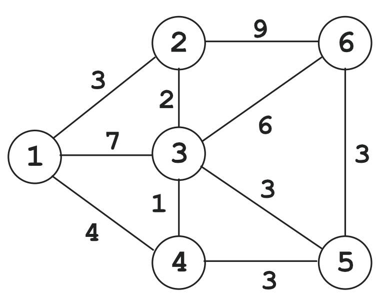
$E_0 = [0^*, \infty, \infty, \infty, \infty, \infty]$
– último nó a ter uma etiqueta definitiva: 1
$E_1[2] = \min(E_0[2], E_0[1] + d_{12}) = \min(\infty, 3) = 3$
$E_1[3] = \min(E_0[3], E_0[1] + d_{13}) = \min(\infty, 7) = 7$
$E_1[4] = \min(E_0[4], E_0[1] + d_{14}) = \min(\infty, 4) = 4$

$E_1 = [0^*, 3^*, 7, 4, \infty, \infty]$
– último nó a ter uma etiqueta definitiva: 2
$E_2[3] = \min(E_1[3], E_1[2] + d_{23}) = \min(7, 5) = 5$
$E_2[6] = \min(E_1[6], E_1[2] + d_{26}) = \min(\infty, 12) = 12$

$E_2 = [0^*, 3^*, 5, 4^*, \infty, 12]$
– último nó a ter uma etiqueta definitiva: 4
$E_3[3] = \min(E_2[3], E_2[4] + d_{43}) = \min(5, 5) = 5$
$E_3[5] = \min(E_2[5], E_2[4] + d_{45}) = \min(\infty, 7) = 7$

Aplicação do algoritmo de Dijkstra – continuação

Slide 18



$E_3 = [0^*, 3^*, 5^*, 4^*, 7, 12]$
– último nó a ter uma etiqueta definitiva: 3
$E_4[5] = \min(E_3[5], E_3[3] + d_{35}) = \min(7, 8) = 7$
$E_4[6] = \min(E_3[6], E_3[3] + d_{36}) = \min(12, 11) = 11$

$E_4 = [0^*, 3^*, 5^*, 4^*, 7^*, 11]$
– último nó a ter uma etiqueta definitiva: 5
$E_5[6] = \min(E_4[6], E_4[5] + d_{56}) = \min(11, 10) = 10$

$E_5 = [0^*, 3^*, 5^*, 4^*, 7^*, 10^*]$
Distância mínima: 10
Caminho mínimo: 1 – 4 – 5 – 6

Algoritmo de Dijkstra na forma tabular

Slide 19

	Nós					
iter	1	2	3	4	5	6
0	0^*	∞	∞	∞	∞	∞
1	0^*	3^*	7	4	∞	∞
2	0^*	3^*	5	4^*	∞	12
3	0^*	3^*	5^*	4^*	7	12
4	0^*	3^*	5^*	4^*	7^*	11
5	0^*	3^*	5^*	4^*	7^*	10^*

Distância mínima: 10

Caminho mínimo: 1 – 4 – 5 – 6

Exemplo de problema de caminho mínimo

Slide 20

O Sr. Ven de Dor, técnico de vendas, vai comprar um carro novo. Este veículo sofrerá uma utilização muito grande (dadas as características da profissão do Sr. Ven de Dor), pelo que, e apesar do o Sr. Ven de Dor se ir reformar daqui a 3 anos (pelo que não precisará mais do carro), poderá ser economicamente mais favorável trocar ainda de carro ao fim de 1 ou 2 anos, em vez de manter este durante os 3 anos. Isto sobretudo porque os custos de utilização e manutenção crescem muito rapidamente com o decorrer dos anos.

O Sr. Ven de Dor sentou-se à sua secretária e calculou o custo total, preço de um carro novo menos o que o stand dá pelo usado, mais os custos de utilização e manutenção (oficina...), de comprar um carro novo no fim do ano i e trocá-lo no fim do ano j.

	i		
j	0	1	2
1	800		
2	1800	1000	
3	3100	2100	1200

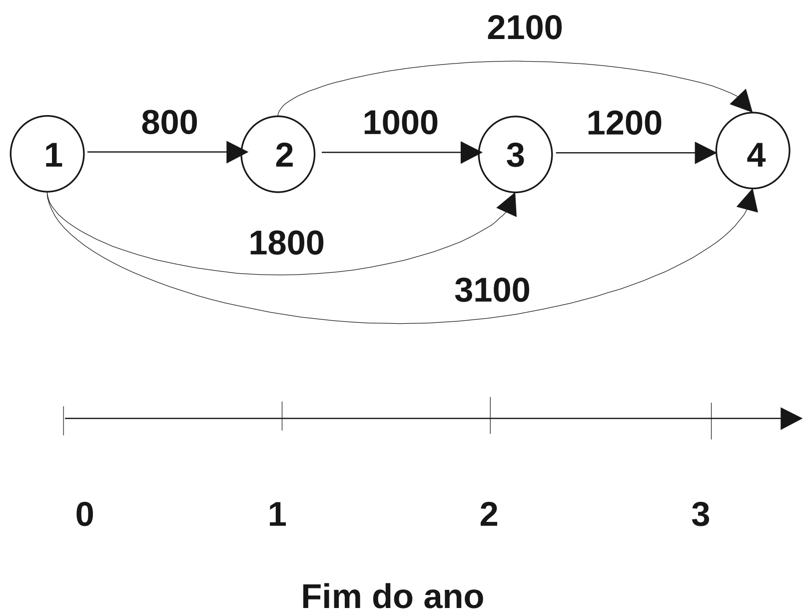
(em milhares de escudos)

Assim, por exemplo, trocar o carro agora comprado (fim do ano 0) no fim do ano 1 e depois manter o carro comprado no fim do ano 1 até ao fim do ano 3, corresponde a um custo de $800 + 2100 = 2900$.

O problema que o Sr. Ven de Dor tem que resolver é determinar quantas vezes deve trocar de carro (se alguma!) de forma a minimizar a sua despesa total com carros durante estes 3 anos.

Resolução (parcial)

Slide 21



Árvore de Suporte de Comprimento Mínimo
Minimal Spanning Tree

Slide 22

Transparências de apoio à leccionação de aulas teóricas

Maria Antónia Carravilla
José Fernando Oliveira

Árvore de Suporte de Comprimento Mínimo

Slide 23

- Definições (para grafos não orientados):
  - uma árvore é um grafo conexo que não contém ciclos;
  - um grafo diz-se conexo se existir uma cadeia (sequência de ramos) ligando qualquer par de nós entre si.
- Problema:

  Determinar a árvore de comprimento total mínimo que suporte todos os nós da rede (i.e. que ligue todos os nós da rede) ("minimal spanning tree").
- Aplicações:
  - redes de comunicações;
  - redes de distribuição de energia.

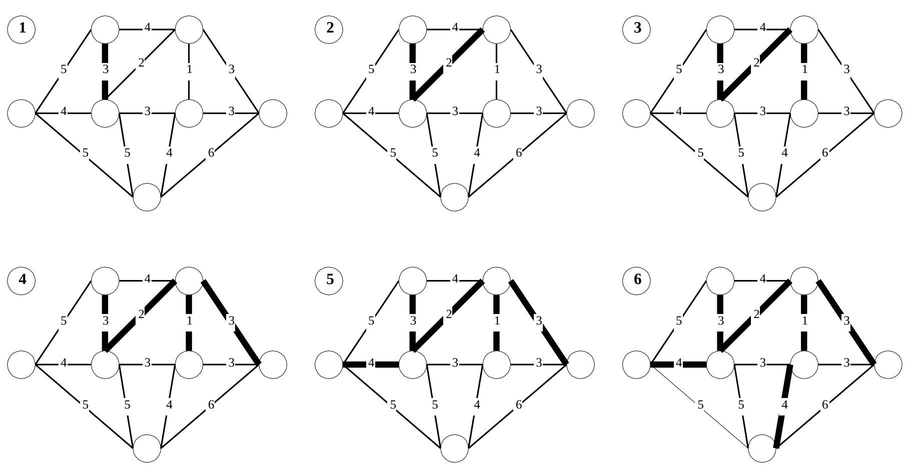
Algoritmo guloso ("Greedy Procedure")

1. Seleccionar um nó arbitrariamente, e ligá-lo ao nó mais próximo;
2. Identificar o nó ainda isolado que esteja mais próximo de um nó já ligado, e ligar estes dois nós;

   Repetir **2.** até que todos os nós estejam ligados entre si.

Slide 24

Algoritmo guloso – Exemplo



Slide 25

Circuitos Eulerianos

Slide 26

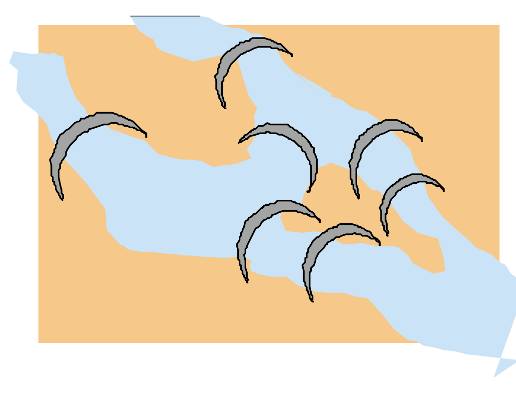
Circuitos Eulerianos – O problema das Pontes de Königsberg

Slide 27

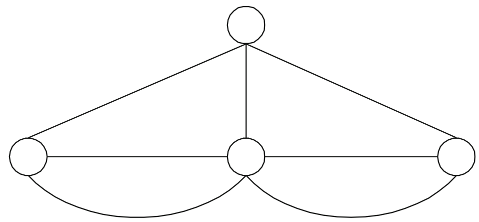
O rio Pregel banha a cidade de Königsberg, na Prússia Oriental, e rodeia a ilha de Kneiphof. A ligar os vários pontos das margens, havia 7 pontes dispostas como se representa na figura.

No seu passeio dominical, os habitantes da cidade procuravam voltar ao ponto de partida, passando uma só vez por todas as pontes.

Leonard Euler estudou o problema e demonstrou a sua impossibilidade em 1736.



Circuitos Eulerianos



Slide 28

Definição de Circuito Euleriano: Um Circuito Euleriano, é um caminho finito em que o nó inicial coincide com o nó final e que passa uma única vez por todos os arcos da rede.

- grafo orientado: Circuito
- grafo não orientado: Ciclo

Teorema de Euler: Um grafo admite um circuito euleriano se e só se for conexo e o número de vértices de grau impar for zero. (o grau de um vértice, corresponde ao número de arcos incidente no vértice)

Problema do Carteiro Chinês
"Chinese Postman Problem (CPP)"

Problema (Kwan, 1962):

Definir circuitos que se aproximem do Circuito Euleriano ideal (repetindo arcos em número mínimo ou de modo a tornar mínimo o aumento de percurso).

Slide 29

Redes de distribuição linear:

- recolha de lixo numa cidade;
- distribuição domiciliária do correio;
- inspecção periódica de redes de energia, de telefones ...

Circuitos Hamiltonianos

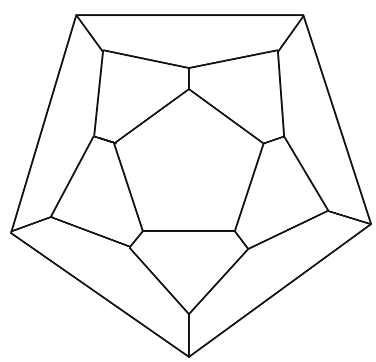
Slide 30

Circuitos Hamiltonianos

Definição de Circuito Hamiltoniano: Um circuito diz-se Hamiltoniano, se passar uma e uma só vez por todos os vértices de uma rede.

Slide 31

A designação provém de um passatempo imaginado pelo islandês Hamilton (1859), no qual se procurava encontrar um caminho que percorresse 20 cidades de todo o mundo, representadas pelos vértices de um dodecaedro de madeira, voltando ao ponto inicial.



Problema do Caixeiro Viajante "Travelling Salesman Problem (TSP)"

Slide 32

Problema:

Pretende-se encontrar o caminho mais curto para um caixeiro viajante que sai de uma cidade, visita n outras cidades e volta àquela de onde partiu, sem repetir nenhuma das cidades visitadas.

Trata-se da pesquisa do circuito hamiltoniano mais curto num grafo de $n+1$ vértices.

O número de soluções possíveis é $n!$.

- Algoritmos conducentes à **solução óptima**, baseiam-se por exemplo em "Branch and Bound" (Little, 1963) e são pouco eficientes.
- Algoritmos que levam a **soluções quase óptimas**, baseiam-se em regras heurísticas (Cicero-Ruggiero, 1972) e são muito eficientes.

Problema do Caixeiro Viajante (TSP) — Modelo

Slide 33

Variáveis de decisão

$$x_{ij} = \begin{cases} 1 & \text{se a cidade } j \text{ seguir imediatamente a cidade } i \\ 0 & \text{se não} \end{cases}$$

Função objectivo

$$\min \sum_{i=1}^{n} \sum_{j=1}^{n} d_{ij} x_{ij}$$

Restrições

$$\begin{array}{rcll} \sum_{i=1}^{n} & x_{ij} & = & 1 \quad \forall_j \\ \sum_{j=1}^{n} & x_{ij} & = & 1 \quad \forall_i \\ \sum_{i \in S_t} \sum_{i \in \bar{S}_t} & x_{ij} & \geq & 1 \quad \forall_{S_t \subset V(cidades)} \end{array}$$

e ainda ... restrições que eliminam Sub-tours ...

Bibliografia

- Ferreira, José António Soeiro (1995). *Apontamentos de Investigação Operacional 1.* FEUP.
- Guimarães, Rui Campos (1979). *Metodologia da Investigação Operacional.* FEUP.

Slide 34

- Guimarães, Rui Campos (1983). *Introdução à Programação Linear.* FEUP.
- Guimarães, Rui Campos (1984). *Planeamento e Controle de Projectos: Método CPM e extensões.* FEUP.
- Hillier, Fraderick S. e Lieberman, Gerald (1995). *Introduction to Operations Research*, Mc Graw-Hill.
- Oliveira, José Fernando (1996). *Apontamentos de Investigação Operacional 1.* FEUP.

- Ravindram, Philips e Solberg (1987). *Operations Research, Principles and Practice.* John Wiley & Sons.
- Taha, Hamdy A. (1997). *Operations Research, an Introduction.* Prentice Hall.
- Tavares, L. V., Oliveira, R. C., Themido, I. H., Correia, F. N. (1997). *Investigação Operacional.* Mc Graw-Hill.

Slide 35