CAPÍTULO 7

ANÁLISE DE REDES

7.1 Conceitos Básicos em Teoria dos Grafos

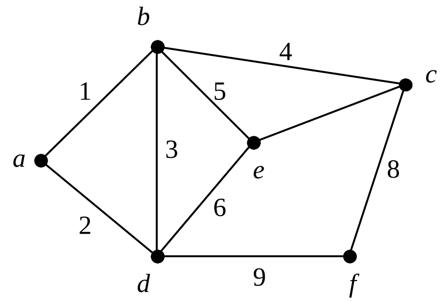
Diversos problemas de programação linear, inclusive os problemas de transporte, podem ser modelados como problemas de fluxo de redes. Algoritmos específicos para determinados tipos de problemas podem ser mais convenientes para a sua solução do que algoritmos mais genéricos.

Antes de continuar, serão apresentadas algumas definições da teoria dos grafos.

Definição 1

Um grafo linear consiste em diversos nós, ou pontos, sendo que cada nó deve estar conectado a um ou mais nós por arcos.

Um exemplo de um grafo linear é apresentado na Figura 7.1.



Nós: *a*, *b*, *c*, *d*, *e*, *f*　　　Arcos: 1, 2, 3, 4, 5, 6, 7, 8, 9

Figura 7.1 - Exemplo de um grafo linear.

Definição 2

Um grafo direto (ou rede direta) é um grafo em que o fluxo ao longo de um arco pode ser efetuado apenas em um sentido.

Entretanto, pode-se substituir um arco com fluxo nos dois sentidos por dois arcos em sentidos opostos. Desta forma, podemos utilizar redes diretas sem que o modelo esteja perdendo a sua generalidade.

Definição 3

Um grafo bipartido é um grafo direto onde os nós são divididos em dois subconjuntos, onde todos os arcos do grafo ligam um nó de um subconjunto a um nó do outro.

Um grafo representando um problema de transporte é um exemplo de grafo bipartido, já que todos os arcos ligam nós das origens a nós dos destinos.

Definição 4

Um caminho ou canal é um conjunto ordenado de arcos que conectam dois nós através de nós intermediários, cada um dos quais estando exatamente em dois arcos do canal.

Um exemplo de canal no grafo da Figura 7.1 é o conjunto dos arcos 1, 5 e 7, que conectam os nós *a* e *c* através dos nós *b* e *e*.

Definição 5

Um grafo conectado é um grafo é um grafo no qual existe caminho entre qualquer par de nós.

O grafo da Figura 7.1 é um grafo conectado.

Definição 6

Um laço é um canal que conecta um nó a ele mesmo.

Os arcos 1, 5, 7, 8, 9 e 2 formam um laço conectando o nó *a* (ou qualquer outro nó do canal) a ele mesmo.

Definição 7

Uma árvore é um grafo conectado que não contém laços.

Exemplos de árvores no grafo da Figura 7.1 incluem os arcos 1, 3, 4, 6, 8 ou os arcos 2, 3, 4, 5, 8. O conjunto de arcos 1, 2, 3, 4, 7, 8 contém um laço (1, 2, 3), portanto não é uma árvore; o conjunto de arcos 1, 3, 7, 8, apesar de não conter laços, não forma uma árvore por não ser um grafo conectado. Pode ser provado que uma árvore com *n* nós possui (*n* - 1) arcos e há pelo menos dois extremos (nós em apenas um arco) em uma árvore.

7.2 Problema de Fluxo Máximo

Um problema de rede usual é a determinação do fluxo máximo entre dois pontos em uma rede. Considere o seguinte exemplo, adaptado de ZIONTS (1974).

"Um produtor de gás natural tem uma rede de tubulações conforme apresentado na Figura 7.2. As capacidades de cada parte da rede estão representadas em bilhões de litros por dia. Um problema ocorreu no ponto *t*, de modo que deseja-se fornecer a maior quantidade de gás possível da produção ao ponto *t*. Portanto, o problema é encontrar a máxima capacidade da rede entre *s* e *t* de modo que a máxima quantidade seja fornecida de *s* para *t*."

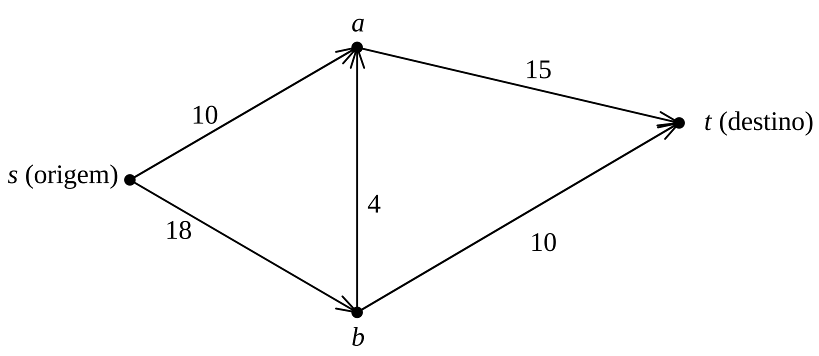


Figura 7.2 - Rede de tubulação de gás.

Mais formalmente, o problema a ser considerado é a maximização do escoamento de um nó *s* (chamado de origem) a um nó *t* (chamado de destino), sujeito às limitações das capacidades dos arcos.

Neste problema, podemos considerar que a quantidade de gás que chega no ponto *t* é igual à quantidade de gás que sai do ponto *s*. Isso pode ser representado por um arco ligando o ponto *t* ao ponto *s*. Desta forma, o problema pode ser representado como um problema de programação linear onde deseja-se maximizar o fluxo do nó *t* ao nó *s* (que é igual ao fluxo que sai do nó *s*, ou ao fluxo que chega no nó *t*). As restrições deste problema, além da capacidade de cada arco da rede, é o fato de que a quantidade de gás que chega em qualquer nó é igual à quantidade de gás que sai deste mesmo nó.

As variáveis de decisão para este problema são:

x_0: fluxo do nó *t* ao nó *s*;

x_1: fluxo do nó *s* ao nó *a*;

x_2: fluxo do nó *s* ao nó *b*;

x_3: fluxo do nó *a* ao nó *t*;

x_4: fluxo do nó *b* ao nó *a*;

x_5: fluxo do nó *b* ao nó *t*.

A função objetivo, a ser maximizada, é o fluxo que chega no nó *t*, representado neste problema por x_0.

Para cada nó, o fluxo de gás que chega é igual ao fluxo de gás que sai. Convencionando sinal negativo ao fluxo de gás que chega e positivo ao fluxo de gás que sai, temos as seguintes restrições:

nó *t*: $x_0 - x_3 - x_5 = 0$

nó *s*: $- x_0 + x1 + x_2 = 0$

nó *a*: $- x_1 + x_3 - x_4 = 0$

nó *b*: $- x_2 + x_4 + x_5 = 0$

As restrições de capacidade são $x_1 \leq 10$, $x_2 \leq 18$, $x_3 \leq 15$, $x_4 \leq 4$ e $x_5 \leq 10$.

O problema pode ser formulado então como apresentado a seguir.

maximizar $z = x_0$

sujeito a:

$$
\begin{array}{ccccccl}
x_0 & & & - x_3 & & - x_5 & = 0 \\
+ x_0 & + x_1 & + x_2 & & & & = 0 \\
 & - x_1 & & + x_3 & - x_4 & & = 0 \\
 & & - x_2 & & + x_4 & + x_5 & = 0 \\
 & x_1 & & & & & \leq 10 \\
 & & x_2 & & & & \leq 18 \\
 & & & x_3 & & & \leq 15 \\
 & & & & x_4 & & \leq 4 \\
 & & & & & x_5 & \leq 10
\end{array}
$$

7.3 Problema de Caminho Mínimo

Um problema bastante comum envolvendo a teoria dos grafos é o problema de rota mais curta, ou caminho mínimo. Para cada arco de um grafo, define-se a distância que ele representa. O objetivo deste tipo de problema é encontrar o caminho mais curto entre dois nós. O problema do caminho mínimo pode ser utilizado também para representar custos ou tempos mínimos, em vez de distâncias.

O algoritmo para a solução de problemas de caminho mínimo que será estudado é o algoritmo de Dijkstra. Este algoritmo determina a distância mínima entre o vértice de origem (s) e os demais vértices.

Para melhor apresentar o algoritmo de Dijkstra, vamos analisar o grafo da Figura 7.3.

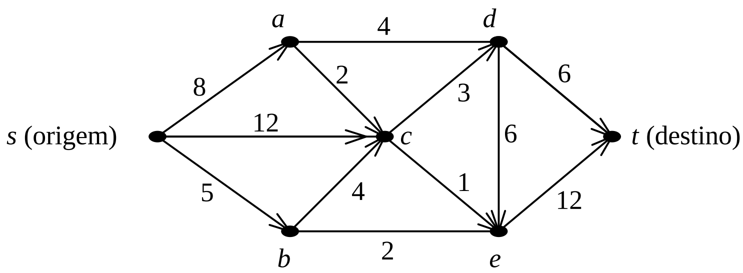


Figura 7.3 - Exemplo de grafo para o problema de caminho mínimo.

De forma a encontrar a rota mais curta, vamos montar duas tabelas. Na primeira (tabela de distâncias mínimas), são colocadas três colunas: o nome do nó, o nó de onde vem o caminho mínimo até o nó de origem e a distância do caminho mínimo. Na segunda (tabela auxiliar), são colocadas três colunas: o nome do nó, o nó de onde vem o caminho considerado (não necessariamente o mínimo) e a distância deste caminho.

Distâncias mínimas

nó	ant.	dist.

Auxiliar

nó	ant.	dist.

O primeiro nó a ser analisado é o nó de origem. Sua distância ao nó de origem é 0. Este nó será então inserido na tabela de distâncias mínimas. Como o caminho mínimo para este nó não vem de nenhum outro nó, na segunda coluna será colocado apenas um traço.

Distâncias mínimas

nó	ant.	dist.
s	-	0

Auxiliar

nó	ant.	dist.

Inserido um nó na primeira tabela, colocaremos na segunda tabela todos os nós atingidos por este nó. No nosso exemplo, os nós a serem inseridos são os nós *a*, *b* e *c*. Para estes nós, o nó de onde vem o caminho é o próprio nó *s*. A distância do nó de origem é a distância do arco percorrido somada com a distância do nó anterior até a origem. No caso, a distância do nó *s* é 0. A tabela auxiliar fica então da seguinte maneira.

Distâncias mínimas

nó	ant.	dist.
s	-	0

Auxiliar

nó	ant.	dist.	
a	*s*	8	
b	*s*	5	
c	*s*	12	

O próximo nó a entrar na primeira tabela será o nó de menor distância até a origem. No caso, o nó a ser incluído é o nó *b*. Como na primeira tabela, o nó *b* ainda não foi inserido, podemos incluí-lo.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5

Auxiliar

nó	ant.	dist.	
a	*s*	8	
b	*s*	5	x
c	*s*	12	

Note que é marcado um X ao lado do nó na segunda tabela, indicando que este nó não deve mais ser considerado no teste de que nó entra na primeira tabela.

Os nós atingidos pelo nó *b* são os nós *c* e *e*, que devem ser inseridos na tabela auxiliar. A distância destes nós até a origem é encontrada somando-se a distância do arco com a distância mínima do nó *b* até a origem (mostrada na tabela de distâncias mínimas). Desta forma, as tabelas ficam da seguinte maneira.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5

Auxiliar

nó	ant.	dist.	
a	*s*	8	
b	*s*	5	x
c	*s*	12	
c	*b*	9	
e	*b*	7	

O próximo nó a entrar na primeira tabela é o nó *e*, vindo de *b*, com distância até a origem 7.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7

Auxiliar

nó	ant.	dist.	
a	*s*	8	
b	*s*	5	x
c	*s*	12	
c	*b*	9	
e	*b*	7	x

Os único nó atingido por ele é o nó *t*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7

Auxiliar

nó	ant.	dist.	
a	*s*	8	
b	*s*	5	x
c	*s*	12	
c	*b*	9	
e	*b*	7	x
t	*e*	19	

O próximo nó a entra é o nó *a*, vindo de *s*, com distância até a origem de 8.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	
c	*b*	9	
e	*b*	7	x
t	*e*	19	

Os nós atingidos pelo nó *a* são os nós *c* e *d*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	
c	*b*	9	
e	*b*	7	x
t	*e*	19	
c	*a*	10	
d	*a*	12	

O próximo nó a entrar é o nó *c*, vindo de *b*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8
c	*b*	9

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	
c	*b*	9	x
e	*b*	7	x
t	*e*	19	
c	*a*	10	
d	*a*	12	

O nó *c* atinge os nós *d* e *e*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8
c	*b*	9

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	
c	*b*	9	x
e	*b*	7	x
t	*e*	19	
c	*a*	10	
d	*a*	12	
d	*c*	12	
e	*c*	10	

O próximo nó a entrar é o nó *c*, vindo de *a*. Como o nó *c* já está incluído, marcamos ele com um X e passamos para o próximo, que é o nó *e* (vindo de *c*) que também já está incluído. O nó *c* (vindo de *s*) também já está incluído. O próximo nó que ainda não está incluído é o nó *d*, vindo de *a* ou de *c*, já que as duas distâncias são iguais.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8
c	*b*	9
d	*a*	12

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	x
c	*b*	9	x
e	*b*	7	x
t	*e*	19	
c	*a*	10	x
d	*a*	12	x
d	*c*	12	x
e	*c*	10	x

Os nós atingidos pelo nó *d* são os nós *e* e *t*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8
c	*b*	9
d	*a*	12

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	x
c	*b*	9	x
e	*b*	7	x
t	*e*	19	
c	*a*	10	x
d	*a*	12	x
d	*c*	12	x
e	*c*	10	x
e	*d*	18	
t	*d*	18	

O nó *e* já está incluído; então o próximo nó a ser incluído é o nó *t*, vindo de *d*.

Distâncias mínimas

nó	ant.	dist.
s	-	0
b	*s*	5
e	*b*	7
a	*s*	8
c	*b*	9
d	*a*	12
t	*d*	18

Auxiliar

nó	ant.	dist.	
a	*s*	8	x
b	*s*	5	x
c	*s*	12	x
c	*b*	9	x
e	*b*	7	x
t	*e*	19	
c	*a*	10	x
d	*a*	12	x
d	*c*	12	x
e	*c*	10	x
e	*d*	18	x
t	*d*	18	x

Como todos os nós já foram incluídos na tabela de distâncias mínimas, o problema está resolvido. A tabela mostra, para cada nó do grafo, a distância mínima até o nó de origem. O caminho mínimo vem do nó indicado na coluna de nó anterior. Desta forma, pode-se determinar o caminho mínimo repetindo-se este passo sucessivamente até que o nó de origem seja encontrado. Para encontrarmos o caminho mínimo do nó *t*, por exemplo, pegamos o nó anterior a ele (*d*). O nó anterior ao nó *d* é o nó *a*, cujo nó anterior é o nó de origem (s).

Desta forma, o caminho mínimo da origem até o nó *t* é (*s* - *a* - *d* - *t*).

O algoritmo pode então ser definido da seguinte maneira:

Passo 1. Inserir o nó de origem na tabela de distâncias mínimas. Sua distância até o nó de origem é 0.

Passo 2. Colocamos na tabela auxiliar todos os nós atingidos por este nó. Para os nós incluídos na tabela auxiliar, o nó de onde vem o caminho é o nó recém inserido na tabela de distâncias mínimas. A distância do nó de origem é a distância do arco percorrido somada com a distância do nó recém inserido na tabela de distâncias mínimas até a origem.

Passo 3. O próximo nó a entrar na tabela de distâncias mínimas será o nó, entre os nós não marcados com um X, de menor distância até a origem. Marca-se este nó com um X. Caso este nó já esteja inserido na tabela de distâncias mínimas, devemos retornar ao passo 3.

Passo 4. Após inserir o nó na tabela de distâncias mínimas, volta-se ao passo 2 até que todos os nós do grafo tenham sido inseridos na tabela de distâncias mínimas.

Passo 5. A tabela de distâncias mínimas indica a distância do caminho mínimo de cada nó até o nó de origem, e o nó de onde vem o caminho mínimo.

Obs. 1: se todos os nós da tabela auxiliar já tiverem sido marcados e alguns nós ainda não tiverem sido incluídos na tabela de distância mínima, isto indica que não há caminho do nó de origem até os nós que não foram incluídos.

Obs. 2: o algoritmo de Dijkstra não é válido caso existam arcos com valores negativos.