MAC–IME–USP

Carlos Eduardo Ferreira
Sala 297A Tel.: 3091 6140
e-mail cef@ime.usp.br

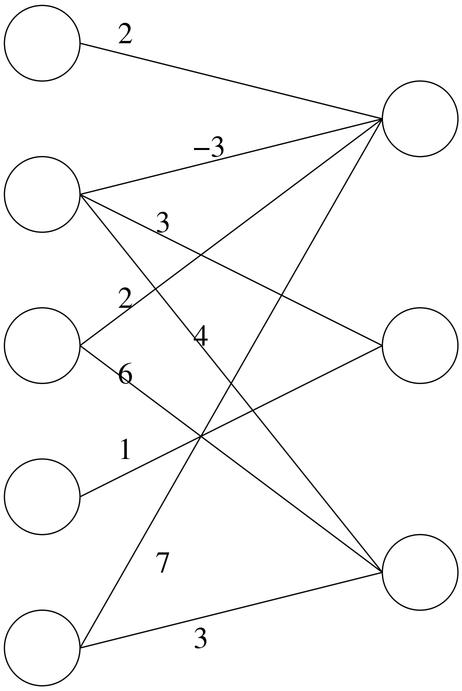
MAC 330 - Programação Inteira - MAC 5780

Segundo semestre de 2002

Lista de Exercício 2

1. Considere o problema de encontrar um emparelhamento de custo máximo em um grafo bipartido.

   - Escreva um problema de programação inteira que modela o problema (como feito em sala de aula).
   - Mostre que a matriz de incidência de um grafo bipartido é totalmente unimodular.
   - Use a relaxação linear para encontrar um emparelhamento de custo máximo no grafo abaixo



2. Uma matriz B apenas com 0 e 1 tem a **propriedade dos 1s consecutivos** se para cada coluna j, $b_{ij} = b_{i'j} = 1$ com $i < i'$ implica que $b_{lj} = 1$ para todo $i < l < i'$.

   Uma condição suficiente mais geral para que uma matriz seja totalmente unimodular é: a matriz A é TU se

   - $a_{ij} \in \{0, -1, +1\}$ para todo i, j.

- para todo subconjunto M de linhas, existe uma partição (M_1, M_2) de M tal que cada coluna j satisfaz
$$|\sum_{i \in M_1} a_{ij} - \sum_{i \in M_2} a_{ij}| \leq 1.$$

Use a condição acima para mostrar que uma matriz com a propriedade dos 1s consecutivos é TU.

3. Considere um grafo $G = (V, E)$ com custos $c_e \geq 0$ nas arestas e um subconjunto $Z \subseteq V$ de vértices terminais.

   - Formule como um problema de programação inteira o problema de encontrar uma árvore de G que conecta todos os terminais (os não-terminais não precisam ser necessariamente conectados) e tenha custo mínimo.
   - Formule o algoritmo guloso para resolver o problema. Mostre através de um contra-exemplo que este algoritmo não encontra sempre a solução ótima.

4. Sabemos que para multiplicar uma matriz $A_{m \times n}$ por uma matriz $B_{n \times p}$ o algoritmo mais simples calcula mnp multiplicações. Considere agora que desejamos calcular várias multiplicações de matrizes:
$$M^1_{m_0 \times m_1} * M^2_{m_1 \times m_2} * \ldots * M^n_{m_{n-1} \times m_n}$$
   Como o produto de matrizes é associativo, podemos fazer este cálculo de várias formas, resultando em um número diferente de multiplicações necessárias para calcular a matriz produto.

   Exemplo: Suponha que queremos calcular o produto das matrizes
$$M^1_{1000 \times 3} * M^2_{3 \times 50} * M^3_{50 \times 2}$$
   Se fizermos, $(M^1 * M^2) * M^3$ teremos 250000 multiplicações. Se fizermos $M^1 * (M^2 * M^3)$, teremos 6300 multiplicações.

   Faça um programa baseado em programação dinâmica que recebe o número n de matrizes a serem multiplicadas, as dimensões destas matrizes ($n + 1$ valores inteiros) e devolve o número mínimo de multiplicações que devem ser feitas para calcular o produto.