Introdução à Teoria dos Grafos

Bacharelado em Ciência da Computação–UFMS, 2005

Problema da Atribuição de Tarefas

Resumo

Existem muitas aplicações que são modeladas em grafos e cuja solução se vincula a algum tipo especial de emparelhamento. Dentre elas, podemos listar algumas famosas como o Problema do Casamento, o Problema da Atribuição de Tarefas, o Problema da Atribuição Ótima, o Problema do Escalonamento de Horários, entre outras. Neste texto, apresentaremos um algoritmo eficiente para solução do Problema da Atribuição de Tarefas.

1 Motivação

Suponha que em uma certa empresa n trabalhadores $x_1, x_2, \ldots, x_n$ estejam disponíveis para executar n tarefas $y_1, y_2, \ldots, y_n$, sendo que cada trabalhador está qualificado para realizar uma ou mais destas tarefas. Queremos uma resposta satisfatória para a seguinte questão: todos os empregados podem ser atribuídos, um empregado por tarefa, para tarefas as quais eles estão qualificados? Este problema é conhecido como o **Problema da Atribuição de Tarefas** ou **Problema da Atribuição de Pessoal**.

2 Modelagem do problema em teoria dos grafos

O Problema da Atribuição de Tarefas pode ser traduzido para a Teoria dos Grafos da seguinte forma. Construímos um grafo bipartido G com partição (X, Y), onde $X = \{x_1, x_2, \ldots, x_n\}$, $Y = \{y_1, y_2, \ldots, y_n\}$ e o vértice x_i está ligado ao vértice y_j se e somente se o trabalhador representado por x_i é qualificado para realizar a tarefa representada por y_j. O problema torna-se então o problema de determinar se o grafo G tem ou não um 1-fator ou um emparelhamento perfeito.

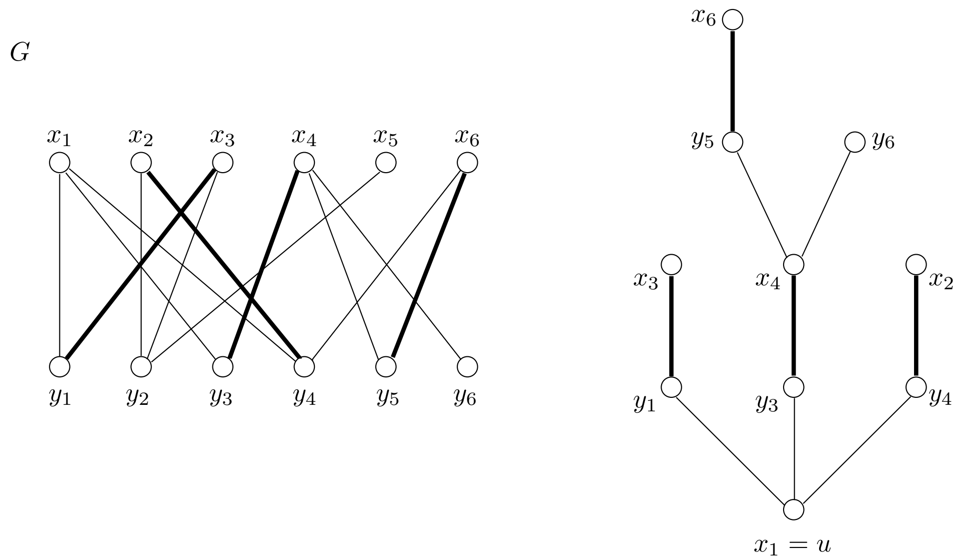
> **Problema** PAT(G): *dado um grafo bipartido G, com partição (X, Y) tal que $|X| = |Y|$, determinar um 1-fator em G.*

De acordo com o teorema de Hall (1937), ou G tem um 1-fator ou existe um subconjunto S de X tal que $|N(S)| < |S|$. Na próxima seção, apresentaremos um algoritmo que soluciona o Problema da Atribuição de Tarefas. Como veremos, dado um grafo bipartido arbitrário G com partição (X, Y), o algoritmo encontra um emparelhamento em G que cobre todo vértice em X ou encontra um subconjunto S de X tal que $|N(S)| < |S|$.

3 1-fatores em grafos bipartidos

Um algoritmo eficiente para encontrar um 1-fator em um grafo bipartido é descrito a seguir. A idéia básica contida no algoritmo é muito simples. Começamos com um emparelhamento arbitrário M. Se M cobre todo vértice em X, então este emparelhamento é um 1-fator. Se não, escolhemos um vértice u em X não coberto por M e procuramos sistematicamente por um caminho aumentador com respeito a M com origem em u. Este método de busca, descrito em detalhes logo abaixo, encontra um tal caminho P, se ele existir; neste caso, $M' = M \triangle E_P$ é um emparelhamento com mais arestas que M, e portanto cobre mais vértices em X. Repetimos este processo agora com M' no lugar de M. Se um caminho como esse não mais existir, o conjunto Z de todos os vértices que estão conectados a u por caminhos alternantes é encontrado. Então, como na demonstração do teorema de Hall (1937), $S = Z \cap X$ satisfaz $|N(S)| < |S|$.

Seja M um emparelhamento em G e seja u um vértice não coberto por em X. Uma árvore alternante $H \subset G$ com raiz u tem as seguintes propriedades: (*i*) $u \in V_H$, e (*ii*) para todo vértice v de V_H, o único caminho de u a v em H é um caminho alternante com respeito a M. Uma árvore alternante com respeita M em um grafo G é mostrada na figura 1.



(a) (b)

Figura 1: (a) Um emparelhamento M em um grafo bipartido G. (b) Uma árvore alternante relativa a M em G.

A busca por um caminho aumentador com respeito a um emparelhamento M e com origem em u está baseada no crescimento de uma árvore alternante H com raiz u. Este procedimento foi proposto inicialmente por Edmonds [3]. Inicialmente, H consiste apenas do vértice u. Então, a árvore "cresce" de tal forma que, em qualquer estágio, ou

(i) todos os vértices de H exceto u são cobertos por M, como na figura 2(a), ou

(ii) H contém um vértice não coberto por M diferente de u, como na figura 2(b).

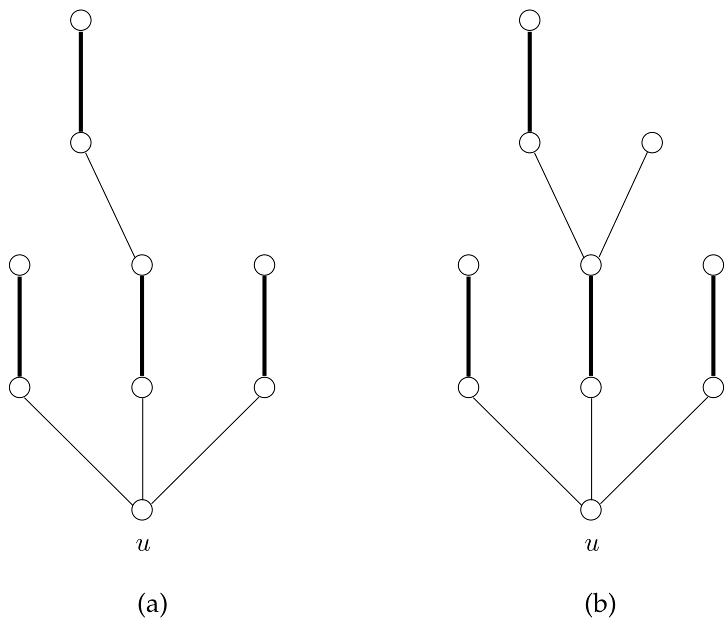


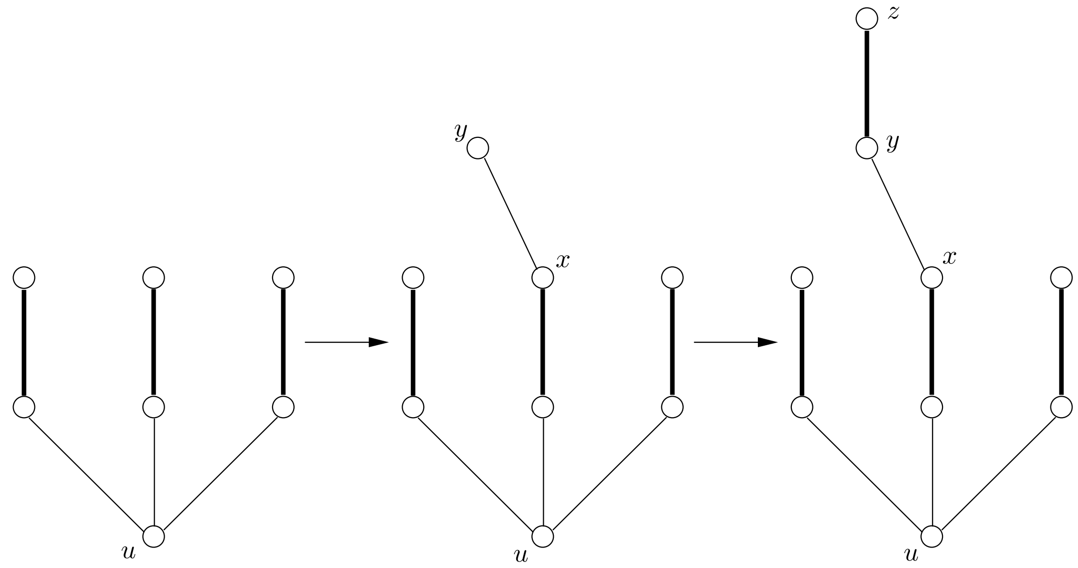
Figura 2: (a) Caso (i); (b) Caso (ii).

Se o caso (i) acontece, como inicialmente, então fazemos $S = V_H \cap X$ e $T = V_H \cap Y$, e temos $N(S) \supseteq T$; assim, ou $N(S) = T$ ou $N(S) \supset T$:

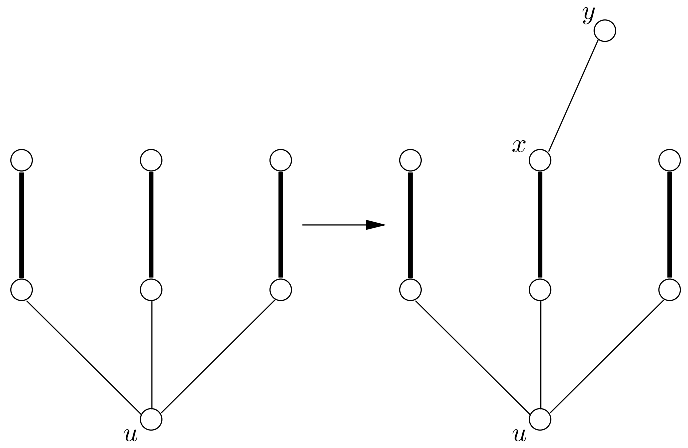
(a) Se $N(S) = T$ então, como os vértices em $S \setminus \{u\}$ estão emparelhados sob M com os vértices em T, $|N(S)| = |S| - 1$, indicando que G não pode possuir um emparelhamento que cobre todos os vértices em X;

(b) Se $N(S) \supset T$, existe um vértice y em $Y \setminus T$ adjacente a um vértice x em S. Como todos os vértices de H, exceto u, estão emparelhados sob M, ou $x = u$ ou então x é emparelhado com um vértice de H. Portanto, a aresta $xy \notin M$. Agora, se y é coberto por M, com $yz \in M$, aumentamos a árvore H adicionando os vértices y e z e as arestas xy e yz a H. Voltamos então ao caso (i). Se y é não é coberto por M, aumentamos H adicionando o vértice y e a aresta xy a H, e então temos o caso (ii). Dessa forma, o caminho de u para y em H é certamente um caminho aumentador com respeito a M e com origem em u, como queríamos.

A figura 3 a seguir ilustra o processo de crescimento de uma árvore alternante H, para um grafo G e um emparelhamento M arbitrários, de acordo com os passos que foram descritos acima. Observe que, nesta figura, existem na verdade dois exemplos: o vértice y, da primeira vez, é coberto por M e não é coberto por M da segunda vez. Dessa forma, essa figura exemplifica, respectivamente, o caso (i) onde todos os vértices exceto u são cobertos por M em H e o caso (ii) onde existe um vértice não coberto por M diferente de u em H.



Caso (i) Caso (i)



Caso (i) Caso (ii)

Figura 3: O processo de crescimento de uma árvore alternante com relação a um emparelhamento.

O algoritmo que acabamos de descrever em alto nível é também conhecido como *Método Húngaro*. O algoritmo pode então ser resumido da seguinte forma.

MÉTODO-HÚNGARO(G): recebe um grafo bipartido G, com partição (X, Y), e devolve um 1-fator de G ou um conjunto $S \subset X$ tal que $|N(S)| < |S|$.

1: Comece com um emparelhamento arbitrário M.
2: Se M cobre todo vértice em X, então pare e devolva M. Caso contrário, seja u um vértice não coberto por M em X. Faça $S = \{u\}$ e $T = \emptyset$.
3: Se $N(S) = T$ então $|N(S)| < |S|$, já que $|T| = |S| - 1$. Pare, já que pelo teorema de Hall (1937) não existe um emparelhamento que cobre todo vértice em X. Caso contrário, seja $y \in N(S) \setminus T$.
4: Se y é coberto por M, seja $yz \in M$. Troque S por $S \cup \{z\}$ e T por $T \cup \{y\}$ e vá para o passo 3; observe que $|T| = |S| - 1$ é mantido depois destas trocas. Caso contrário, seja P um caminho aumentador com respeito a M de u para y. Troque M por $M' = M \triangle E_P$ e vá para o passo 2.

Considere, por exemplo, o grafo G na figura 4, com emparelhamento inicial $M = \{x_2y_2, x_3y_3, x_5y_5\}$. Na figura 4(a) uma árvore alternante relativa ao emparelhamento M destacado é construída, começando com x_1, e o caminho aumentador $x_1y_2x_2y_1$ é encontrado. Então, o algoritmo produz um novo emparelhamento $M' = \{x_1y_2, x_2y_1, x_3y_3, x_5y_5\}$ e uma árvore alternante relativa ao emparelhamento M' é agora construída a partir de x_4 (veja a figura 4(c) e 4(d)). Como não existe caminho aumentador com respeito a M' com origem em x_4, o algoritmo termina. O conjunto $S = \{x_1, x_3, x_4\}$, com vizinhos $N(S) = \{y_2, y_3\}$, mostra que G não tem um 1-fator.

Um algoritmo pode ser proposto para encontrar um emparelhamento de cardinalidade máxima em um grafo bipartido, fazendo pequenas mudanças no algoritmo que acabamos de apresentar. Um algoritmo eficiente que determina um emparelhamento de cardinalidade máxima em um grafo arbitrário foi proposto por Edmonds em [3].

Referências

Este texto foi produzido com a consulta às referências [1] e [2]. Como mencionado anteriormente, o método húngaro pode ser facilmente alterado para encontrar um emparelhamento de cardinalidade máxima em um grafo bipartido. Este algoritmo, neste caso, torna-se um caso particular do algoritmo proposto por Edmonds em [3], que constrói um emparelhamento de cardinalidade máxima em um grafo arbitrário. O algoritmo de Kuhn-Munkres [4,6] encontra um emparelhamento de valor máximo em um grafo bipartido com custos nas arestas. O problema associado a esta solução é chamado **Problema da atribuição ótima**. Uma excelente referência sobre emparelhamentos é o livro de Lovász e Plummer [5].

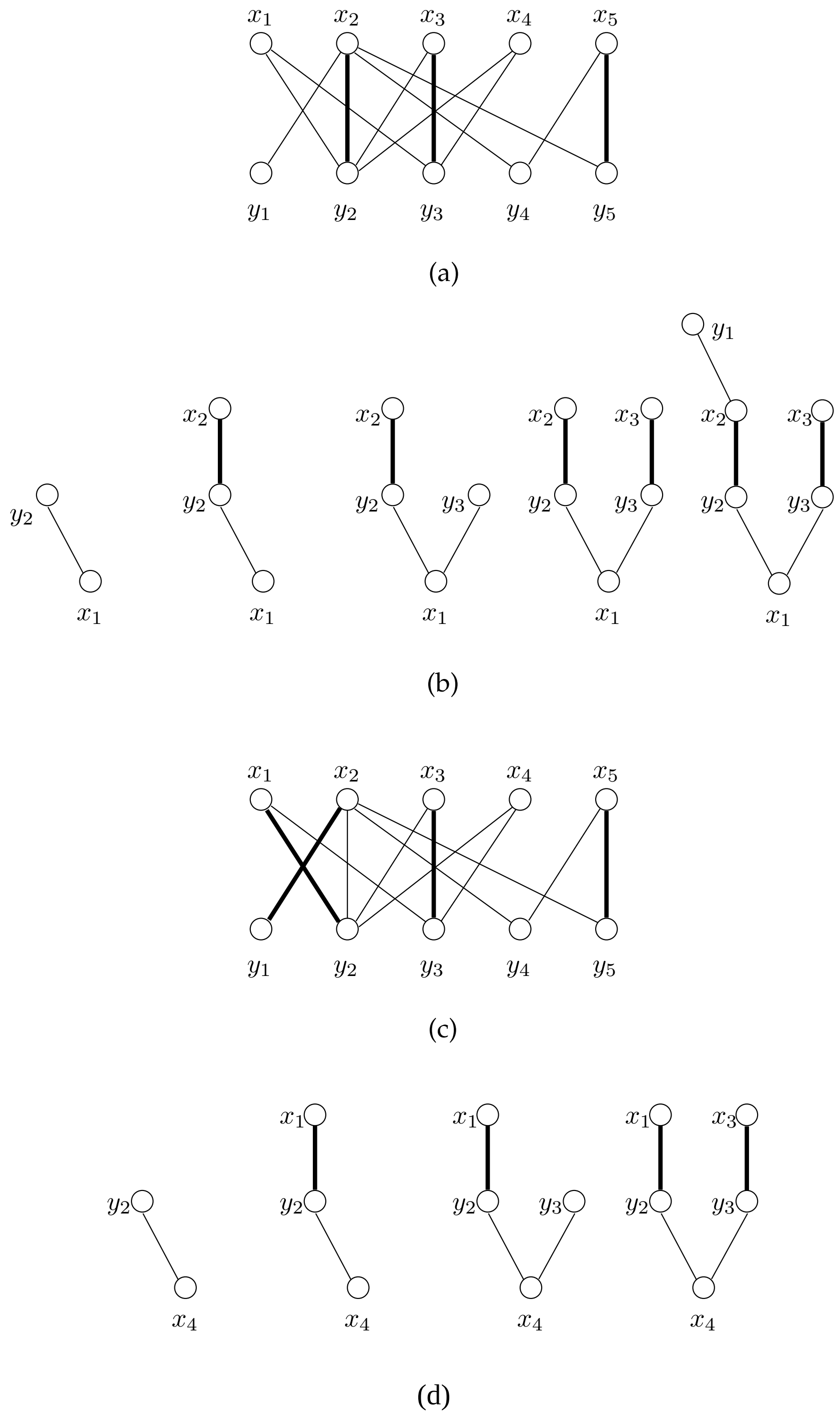


Figura 4: (a) Um emparelhamento M. (b) Uma árvore alternante relativa a M. (c) Um emparelhamento M'. (d) Uma árvore alternante relativa a M'.

[1] J. A. Bondy and U. S. R. Murty, **Graph Theory with Applications**, North-Holland, 1976.

[2] G. Chartrand and O. R. Oellermann, **Applied and Algorithmic Graph Theory**, McGraw-Hill, Inc., 1993.

[3] J. Edmonds, **Paths, trees and flowers**, Canad. J. Math., 17, pp. 449-467, 1965.

[4] H. W. Kuhn, **The Hungarian method for the assignment problem**, Naval Res. Logist. Quart., 2, pp. 83-97, 1955.

[5] L. Lovász and M. D. Plummer, **Matching Theory**, North-Holland, 1986.

[6] J. Munkres, **Algorithms for the assigment and transportation problems**, J. Soc. Indust. Appl. Math., 5, pp. 32-38, 1957.