Relatório de Implementação do Trabalho Prático: A Terra Herdada por Tio Tom

Leonardo V. Anastácio[1], **Lucas L. Mentz**[1]

[1]Teoria dos Grafos – Bacharelado em Ciência da Computação
Centro de Ciências Tecnológicas – Universidade do Estado de Santa Catarina
Joinville – SC – Brasil

{leonardo.valerio,lucas.mentz}@edu.udesc.br

1. Introdução

'A Terra Herdada por Tio Tom' descreve um problema no qual o dono de latifúndio quer vender suas terras em partes pequenas. O problema trata de emparelhamento máximo. O problema informa como entrada uma matriz, que representa a fazenda do tio Tom, onde os pontos marcados são caracterizados como 'lago' e os não marcados como terrenos para possível venda. Além disso, há uma regra que propriedades só podem ser vendidas em pares de terrenos, ou seja, em partes retangulares de tamanho de dois terrenos. O objetivo é encontrar a quantidade máxima de pares de terrenos que podem ser vendidos.

2. Abstração e Fundamento da Solução

A solução encontrada corresponde em interpretar o mapa da fazenda como um grafo (G_1) em forma de matriz de adjacência, em que um terreno, que condiz a um vértice, está conectado com todos seus terrenos adjacentes. O modo utilizado de encontrar a máxima distribuição de pares de terrenos foi com uso do algoritmo de cálculo de fluxo máximo Ford-Fulkerson. Esse algoritmo percorre os caminhos possíveis entre um vértice denominado origem e outro denominado destino. Para fazer essa solução trazer resultado como um número representando o emparelhamento máximo chegamos a um procedimento que se resume a particionar o grafo e executar Ford-Fulkerson com múltiplas origens e múltiplos destinos – explicado em detalhes na seção 3.

A bipartição é feita com auxílio do algoritmo de busca em profundidade (BFS) partindo de qualquer nó do mapa. O algoritmo BFS percorre um grafo de uma forma que seu percurso representa uma árvore – porém aqui iremos descaracterizar a árvore por criar ciclos – que denominaremos G_2. A bipartição é feita separando os vértices de acordo com sua altura em G_2 (partindo do nó raiz da árvore) de forma que os nós correspondentes à altura de número par serão classificados como 'esquerda' e o restante como 'direita'. Após a separação, os vértices condizentes como 'esquerda' ficarão à esquerda e os restantes à direita de um grafo bipartido, denominado G_3. Para permitir o cálculo do emparelhamento máximo com o algoritmo de fluxo máximo é necessário criar um vértice (source) e adicionar arestas de peso 1 deste aos vértices à esquerda de G_3 – origens – e outro novo vértice (sink) com arestas também de peso 1 partindo dos vértices à direita – destinos – e chegando a este. Por último, para encontrar a solução do problema é executado algoritmo de Ford-Fulkerson no grafo bipartido, partindo do vértice source e chegando em sink. Este algoritmo visa calcular o máximo fluxo possível, que nessa representação do grafo corresponde ao emparelhamento máximo.

3. Passo-a-Passo da Solução

A entrada do problema será correspondente a matriz de adjacência. Cada célula da matriz ganhará um número que o identifica no grafo. Um mapa de exemplo Figura 1) será usado para apresentar com maior clareza o resultado de cada etapa da solução.

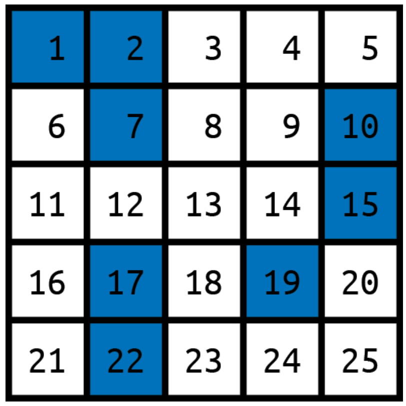

Figura 1. Mapa com identificação numérica dos lotes e representação dos lagos em azul. Produção própria.

Com esse mapa formamos um grafo em uma forma que se assemelha a uma árvore – não o é devido a possuir ciclos, apesar de não influenciar na solução – por meio da execução do algoritmo de busca em largura modificado para marcar a profundidade do nó alcançado, partindo de profundidade zero no vértice de origem. Se o mapa tiver partes desconexas então será necessário mais de uma chamada do algoritmo BFS. Com esse grafo similar a uma árvore fazemos a remoção das arestas cuja origem são vértices de profundidade ímpar, deixando assim o grafo totalmente bipartido (Figura 2).

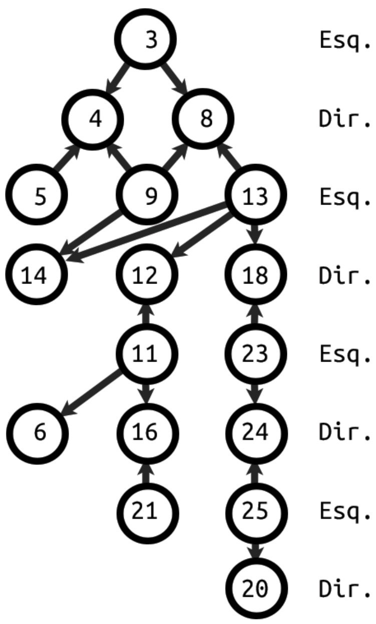

Figura 2. Grafo G_2 obtido com BFS partindo do nó 3. Produção própria.

O grafo formado (G_2, figura 2) é então reorganizado seguindo a paridade de sua profundidade – par vai pra esquerda, ímpar vai pra direita. Além disso, adiciona-se um novo vértice de origem e um novo vértice de destino para cálculo do emparelhamento por algoritmo de fluxo máximo. O novo vértice de origem é chamado s e o novo vértice de destino é chamado t. São adicionadas novas arestas partindo de s e chegando a todos os vértices da esquerda e partindo de todos os vértices da direita e chegando a t. Por fim, todas as arestas do grafo resultante recebem peso igual a um para representar a restrição de que um vértice ou terreno só pode parear-se com um outro vértice. Seguindo o exemplo dessa seção, o resultado é o grafo G_3 apresentado na figura 3.

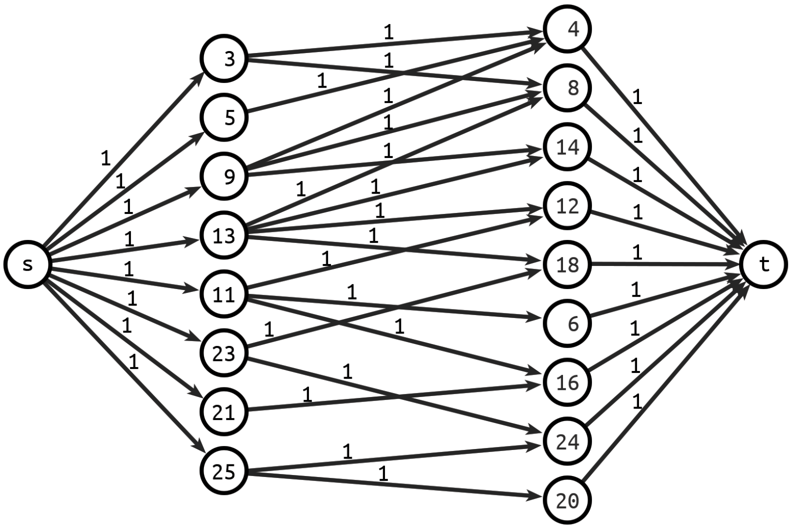

Figura 3. Grafo bipartido G_3 com novos origem e destino. Produção própria.

O passo seguinte trata da simples execução do algoritmo de Ford-Fulkerson sobre G_3 com origem em s e destino em t. O funcionamento do algoritmo não será detalhado aqui. Um possível resultado obtido pelo algoritmo é apresentado na figura 4.

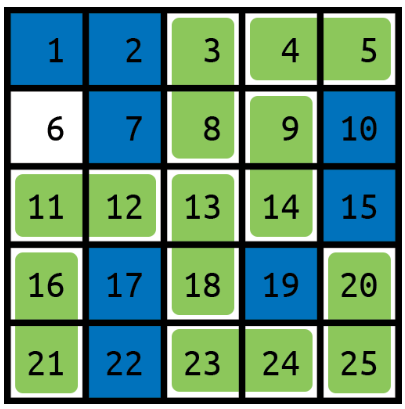

Figura 4. Em verde são as propriedades selecionadas pelo algoritmo Ford-Fulkerson. Produção própria.

O algoritmo encontrou fluxo máximo no grafo G_3 igual a 8. Como a modelagem do problema trata do fluxo máximo como o maior arranjo de propriedades compostas por dois terrenos vizinhos temos que o resultado do problema no exemplo dado é oito propriedades.

4. Conclusão

O algoritmo de Ford Fulkerson se mostrou útil também para cálculo de emparelhamento contanto que o grafo a emparelhar tenha sido bipartido, suas arestas sigam somente um sentido e tenham todas peso 1. Nestas condições o problema do emparelhamento pode ser comparado a um problema comum de fluxo máximo, sendo assim possível resolver com algoritmos do tipo.