Algoritmo de Kuhn-Munkres (Hungarian Algorithm)

Fernando Rezende Domingues, Otávio Augusto Gonçalves Silva, Rafael Moreira de Melo

UFU

O problema

O Problema de Atribuição

Imagine que você tem um grupo de trabalhadores e um conjunto de tarefas. Cada trabalhador tem um custo diferente para realizar cada tarefa. O desafio é atribuir cada trabalhador a uma única tarefa de forma que o custo total seja o menor possível.

Esse é um problema clássico de otimização combinatória, conhecido como "Problema de Atribuição".

Uma breve história

As origens do algoritmo

O algoritmo foi desenvolvido e publicado por **Harold Kuhn** em 1955.

Kuhn o batizou de "Método Húngaro" em homenagem a dois matemáticos húngaros, **Dénes Kőnig** e **Jenő Egerváry**, cujos trabalhos foram a base para o algoritmo.

Curiosamente, mais tarde foi descoberto que **Carl Gustav Jacobi** já havia resolvido o problema de atribuição no século XIX, com sua solução publicada postumamente em 1890.

O que é o Algoritmo Húngaro?

É um algoritmo de otimização que resolve o problema de atribuição em tempo polinomial.

Ele encontra a atribuição de custo mínimo entre dois conjuntos de elementos (por exemplo, trabalhadores e tarefas).

A ideia central é transformar a matriz de custos original em uma nova matriz, onde a atribuição ótima pode ser facilmente identificada.

Como o Algoritmo Húngaro funciona?

A matriz de custos

O algoritmo opera em uma matriz de custos, onde as linhas representam um conjunto de elementos (ex: trabalhadores) e as colunas representam o outro conjunto (ex: tarefas).

Cada célula da matriz contém o custo de atribuir um elemento de um conjunto a um elemento do outro.

Passo a passo do algoritmo

1. **Preparação da Matriz:**
   Crie uma matriz de custos n x n. Se a matriz não for quadrada, adicione linhas ou colunas com custo zero.

2. **Redução de Linhas:**
   Para cada linha, encontre o menor elemento e subtraia-o de todos os elementos daquela linha.

3. **Redução de Colunas:**
   Após a redução das linhas, faça o mesmo para as colunas: encontre o menor elemento de cada coluna e subtraia-o de todos os elementos da coluna.
4. **Atribuição inicial:**
   Faça o máximo de atribuições possíveis, e cubra as colunas que contém alguma atribuição.

Passo a passo do algoritmo

5. **Teste de Otimização:**
   - Procure por zeros não cobertos, se não for encontrado nenhum zero, pule para etapa **6**:
     - Se o zero achado está na mesma linha que um zero atribuído, marque-o, descubra a coluna do zero atribuído e cubra a linha correspondente.
     - Caso não haja atribuições na mesma linha, iremos procurar por um **caminho de aumento**, seguindo os seguintes sub-passsos:
       1. Encontre um zero atribuído na coluna correspondente. Se haver algum pule para a sub-etapa **2**, senão, pare.
       2. Encontre um zero marcado na linha correspondente. Pule novamente para a sub-etapa **1**.
     - Para todos os zeros encontrados no caminho, atribua os zeros marcados, e desatribua os zeros atribuídos. Desmarque todos os zeros e descubra todas as linhas.

Passo a passo do algoritmo

5. **Teste de Otimização (continuação):**
    - Depois de acharmos o caminho de aumento, e tratá-lo, iremos cobrir as colunas com zeros atribuídos (já que mudamos eles). E em seguida repetir o passo **5** - até que não se ache zeros descobertos, e assim pulamos para o passo **6**.
6. **Ajuste da matriz:**
    - Se o número de zeros atribuídos é igual a n (ordem da matriz), então terminamos;
    - Senão, iremos procurar o menor elemento não coberto. Iremos subtrair todas as posições não cobertas por esse menor elemento, e somá-lo nas posições em que a linha e coluna está coberta. E voltamos a etapa **5**.

Conclusão

Resumo

- **Eficiência:** Encontra a solução ótima para o problema de atribuição de forma eficiente (complexidade $O(n^3)$). O que se fossemos fazer no método da força bruta seria $O(n!)$.

- **Garantia de Otimalidade:** O algoritmo sempre encontra a atribuição de menor custo total.

Obrigado a todos!

Perguntas ?