Pontifícia Universidade Católica de Minas Gerais
Ciência da Computação

Problema da mochila valiosa

Aluno:
Gabriel Oliveira Campos

Junho
2018

Conteúdo

1 Resumo **1**

2 Definição **2**

3 Análise de Complexidade **3**

4 NP-Completo **4**

5 Heurística **5**

Bibliografia **6**

1 Resumo

Este trabalho aborda o problema da mochila valiosa(Knapsack problem), em que temos que decidir dentre os objetos disponíveis quais levar na mochila, sendo que não se pode ultrapassar o peso máximo da mochila e deve ser levado o maior número de objetos possíveis(objetos mais valiosos). Já existem diversas maneiras de resolver esse problema, indo desde o algoritmo por Força bruta até Heurísticas mais complexas, como o Guloso por peso.

2 Definição

No problema da mochila temos N itens que podem ser levados na mochila, onde cada um desses itens possui um peso P e um custo X, e a mochila tem uma capacidade C para carregar esses itens. Nesse trabalho temos então um problema de otimização combinatória, onde queremos carregar itens que somem o maximo possivel de utilidade sem estourar a mochila, então deve se escolher os itens que fariam a mochila carregar os itens que combinados serão os mais valiosos, ou seja, tornar a mochila mais valiosa possível.

3 Análise de Complexidade

O problema da mochila valiosa possui uma complexidade de O(N*P) onde N é a quantidade de itens e P é a capacidade Máxima da mochila.

Tendo que temos uma entrada com N pesos de itens,N custo de itens e 1 capacidade da mochila podemos afirmar que isso é 2n+1, porém não podemos escrever NP em função de 2n+1, nossa definição de tamanho deveria levar em conta o valor e não apenas a presença do parâmetro c. Poderíamos adotar o par (2n,c) como tamanho da instância, mas esse definição não é a ideal.

Considere uma questão mais básica: qual a definição mais razoável de tamanho de um número natural, como c por exemplo? Resposta: número de caracteres. Por exemplo, o tamanho do número 2568773 é 7 (e não 2568773). Como c é representado por aproximadamente lg c caracteres, o tamanho de uma instância (P1,...,Pn, C1,...,Cn, Capacidade) do problema é o par(2n, lg c).

Então podemos afirmar que a complexidade desse algoritmo é exponencial, O(n 2(elevado)lg C).

4 NP-Completo

O problema da mochila é NP-Completo pois podemos reduzir o problema CNF-SAT para o 3-CNF-SAT, depois podemos reduzir novamente o 3-CNF-SAT para o Subset-SUM (Dados números naturais p1, p2, . . . , pn e c, decidir se existe um subconjunto X de {1,2,. . . ,n} tal que p(X) = c), e por final podemos reduzir o Subset-SUM para o Problema da mochila valiosa. Como o problema do CNF-SAT é NP-Completo o problema da mochila também será.

5 Heurística

A heurística mais utilizada para a resolução do problema da mochila é a da programação dinâmica. A programação dinâmica tem a ideia é guardar em uma tabela, digamos t, as soluções das (sub)instâncias do problema.

A tabela é definida assim: para i = 0,1,...,n e b = 0,1,...,c,

t[i,b] é o valor de uma solução da instância (p1,...,pi, v1,...,vi, b) do problema.

A tabela t satisfaz a seguinte recorrência: para todo i menor que 1 e todo b,

t[i,b] = t[i-1,b] se pi maior que b e

max (t[i-1,b] , t[i-1,b-pi] + vi) se pi menor igual b .

(Observe que todos os números da forma b-pi são não negativos e portanto a posição (i-1,b-pi) da tabela está bem definida.)

Bibliografia

https://pt.slideshare.net/mcastrosouza/problema-da-mochina-01-knapsack-problem
https://www.ime.usp.br/pf/analise_de_algoritmos/aulas/mochila-bool.html
https://pt.wikipedia.org/wiki/Problema_da_mochila