MC558 - Análise de Algoritmos II - Turmas A e B
Laboratório 6 - O Problema do Troco
Marcelo Azevedo Gonçalves dos Santos RA:106140

O Problema

Dado n tipos de moeda, cada uma com peso p_j e valor v_j (j = 1...n), descobrir qual a combinação de moedas que tem valor total igual a V, troco a ser dado, com o menor peso possível.

A Redução

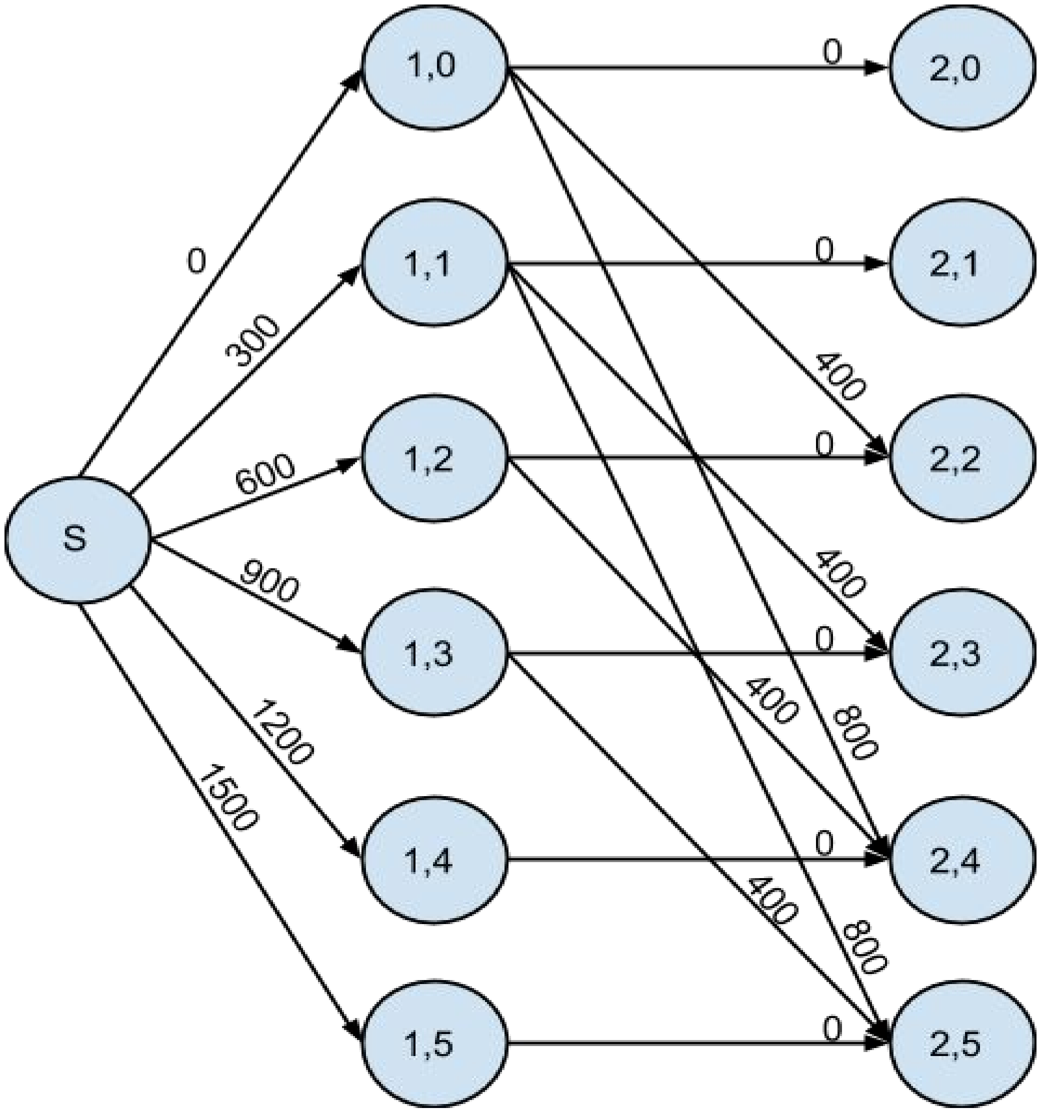
Primeiramente ordena-se as moedas em ordem crescente de valor O(nlogn). Em seguida, cria-se um grafo com n*(V + 1) + 1 vértices, ou seja, um vértice fonte e (V+1) vértices para cada moeda O(n). O grafo a ser construido será uma DAG com n camadas, sendo que o vértice de número k está na camada k % (V + 1) . O fluxo do grafo segue a lógica de que para cada camada i, fizemos todas as considerações das moedas de indíces 0 até i para todas os valores de 0 até V. Para facilitar a compreensão, vamos denotar um vértice como um par ordenado (i,j) em que j é o valor total até o vértice da moeda de indíce i. Assim, um vértice da forma (i,j) conecta-se com outro (i+1,j') se:

1 - j' >= j
2 - j' = j + x * ($v_{(i+1)}$) com x pertencente aos inteiros.
3 - j' <= V

Se todas essas condições forem atendidas então o cria-se uma aresta de (i,j) -> (i+1,j') com peso de $p_{(i+1)}$ * x. Com relação ao vértice fonte, esse conecta-se com todos os vértices (1,j) com peso j*p_0 .
Vamos exemplificar a redução com um exemplo. Seja as moedas em ordem crescente de valor:

Indice	**1**	**2**	**3**	**4**
Valor	1	2	3	4
Peso	300	400	450	500

Vamos agora considerar o vértice (1,0). Esse vértice somente irá conectar-se aos vértices (2,0) com peso 0,(2,2) com peso 400 e (2,4) com peso 800. Em seguida o vértice (1,1) irá conectar-se com o vértice (2,1) com peso 0, (2,3) com peso 400 e (2,5) com peso 800.
A figura abaixo exemplifica uma parte do processo:



Repetindo esse processo para todos os vértices, o objetivo é encontrar o menor caminho da fonte ao vértice (n,V), no caso do exemplo acima seria o vértice (4,5).

Complexidade

Para construir o grafo, temos que criar n*(V + 1) + 1 vértices e conectar os vértices das camadas i com as da camada i + 1 (quando possível) O(n*V).
Com o DAG construida, roda-se o algoritmo de dijkstra O(V+E).Por fim, basta percorrer o caminho mínimo gerado e atualizar a quantidade de moedas utilizadas agora considerando o indíce de entrada, não o indíce das moedas em ordem crescente de valor O(n).No total, temos que o algoritmo roda em tempo O(n*V).Como V não representa o tamanho da entrada, o algoritmo roda em tempo O(n*2^b) em que b = logV, ou seja, bits necessarios para representar o número V. Dessa maneira, a redução não nos permite concluir nada sobre o problema do troco, pois a solução roda em tempo exponencial.