

UNIVERSIDADE TUIUTI DO PARANÁ

Winderson Jose Barboza dos Santos

PROBLEMA DA MOCHILA

**CURITIBA
2018**

Winderson Jose Barboza dos Santos

PROBLEMA DA MOCHILA

Trabalho apresentado ao curso de Ciência Da Computação da Universidade Tuiuti do Paraná com requisito avaliativo do 2º Bimestre da disciplina de Inteligência Artificial ministrada pela professora Chaua.

CURITIBA

2018

SUMÁRIO

INTRODUÇÃO	4
DESENVOLVIMENTO	5
Descrição do problema	5
Descrição do algoritmo	5
Problema Mochila	5
Heurística	6
Hill Climbing	7
Tabu search	8
Resultados experimentais	8
Hill Climbing	8
Tabu search	9
CONCLUSÃO	10
REFERÊNCIAS	11

INTRODUÇÃO

Esse trabalho tem o objetivo de demonstrar e falar sobre duas técnicas bastante conhecidas se tratando de problema de otimização matemática, são elas Hill Climbing e Tabu search. Veremos como elas se aplicam na resolução do problema da mochila utilizando a heurística GRASP.

DESENVOLVIMENTO

Descrição do problema

O problema da mochila consiste em um problema de otimização combinatória, onde a partir de um modelo de uma situação em que temos um número limitado de itens para colocar em uma mochila levando em consideração o peso de cada item e a capacidade do recipiente, e queremos maximizar o valor total de utilidade, sendo que a cada item é associado o seu valor de utilidade.

Descrição do algoritmo

A estrutura do algoritmo consiste em:

Problema Mochila

O algoritmo do problema da mochila foi implementado em cima de um exemplo simples onde é descrito os dados como o tipo, o peso e o valor de três itens.

Tipo	Peso (Kg)	Valor
1	3	40
2	5	100
3	2	50

A partir desses dados foi definido a quantidade disponível de cada item para ser colocado na mochila. O item 1 contendo a quantidade disponível de 3 unidades, item 2 com a quantidade disponível de 2 unidades e o item 3 com 5 unidades.

Tipo	Peso (Kg)	Qtd. Disponível	Total (Kg)
1	3	3	9
2	5	2	10
3	2	5	10

Dada essa definição de valores obtivemos um total de 29 kg se levado em consideração a soma dos pesos de todos os itens disponíveis.

Após a definição dos itens bem como suas quantidades disponíveis e seus pesos totais, foi estabelecido a capacidade de 25 kg para a mochila.

Heurística

Na definição da heurística foi implementado uma estratégia com aplicação de probabilidade, onde é analisado cada item bem como sua quantidade e definidos as possíveis combinações. Se pegarmos o item do tipo 1, obtemos o peso de 3 kg por item, sendo eles com valor de 40 cada, deste item temos 3 unidades disponíveis. Se partindo desses dados conseguimos levantar algumas informações como a quantidade de possibilidades de uso desse item que teremos na geração de um estado, que nesse caso serão 4 possibilidades.

Possibilidade	Quantidade de itens
1	0 (zero)
2	1
3	2
4	3

Se aplicarmos a probabilidade e a probabilidade acumulada em cima de cada uma dessas possibilidades obtemos os seguintes dados.

Possibilidade	Qtd. itens	Probabilidade	P. Acumulada
1	0 (zero)	0,25	0,25
2	1	0,25	0,50
3	2	0,25	0,75
4	3	0,25	1,0

Se analisarmos é possível concluir que para cada possibilidade teremos 0,25 (25%) de chance de umas das 4 opções serem escolhidas, sabendo disso foi definido que será gerado um valor aleatório que representa o percentual definido para aquele item, para obtermos o valor deste percentual, será feita a divisão por 100. Com o valor obtido teremos que comparar em qual probabilidade acumulada esse valor se encaixa, por exemplo, o número definido é o 40, esse número dividido por 100 tem como resultado 0,4 que se comparado com a probabilidade acumulada das opções ele entra como maior 0,25 e menor que 0,50, consequentemente a opção escolhida para aquele estado daquele item será a possibilidade 2.

Hill Climbing

O algoritmo de implementação da técnica de Hill Climbing consiste em sua inicialização necessitar de parâmetros que podem ser usados como critérios de parada, são eles:

- Número máximo de iterações
- Número máximo de iterações sem melhora
- Tempo máximo
- Atingiu o objetivo

Na implementação foi usado o número de iterações e o número de iterações sem melhora.

O algoritmo consiste na geração de estados vizinhos utilizando o problema da mochila e verificando a cada interação se houve melhora, esse processo dura enquanto o número de interações máxima ou sem melhoras não foi atingido. Após cada geração é verificado o custo atual e custo vizinho para ocorrer a comparação se o estado vizinho for um estado válido.

Tabu search

O algoritmo de implementação da técnica de Tabu search consiste na mesma ideia da técnica de Hill Climbing em sua inicialização necessitar de parâmetros que podem ser usados como critérios de parada, são eles:

- Número máximo de iterações
- Número máximo de iterações sem melhora
- Tempo máximo
- Atingiu o objetivo

Na implementação foi usado o número de iterações e o número de iterações sem melhora e o tamanho da lista tabu.

O algoritmo consiste na geração de estados vizinhos utilizando o problema da mochila e verificando a cada interação se houve melhora, esse processo dura enquanto o número de iterações máxima ou sem melhoras não foi atingido. Após cada geração é verificado o custo atual e custo vizinho para ocorrer a comparação e também a verificação da existência de um estado igual ao gerado na lista tabu, essa lista fica responsável por armazenar os estados já válidos já avaliados, se a lista chegar ao seu tamanho máximo os itens com menos frequência serão removidos.

Resultados experimentais

A tabela dos itens disponíveis e seus dados:

Tipo	Peso	Valor	Qt. disponível
1	3	40	3
2	5	100	2
3	2	50	5

Hill Climbing

O algoritmo nas interações com 100, 1000 e 10000 apresentou evolução logo nas primeiras iterações depois disso continuou constante em seus estados.

VL:450 - MOCHILA:| TIPO:1 | QT:0 || TIPO:2 | QT:2 || TIPO:3 | QT:5 | KG: 20

VL:400 - MOCHILA:| TIPO:1 | QT:0 || TIPO:2 | QT:2 || TIPO:3 | QT:4 | KG: 18

VL:390 - MOCHILA:| TIPO:1 | QT:1 || TIPO:2 | QT:1 || TIPO:3 | QT:5 | KG: 18

VL:380 - MOCHILA:| TIPO:1 | QT:2 || TIPO:2 | QT:2 || TIPO:3 | QT:2 | KG: 20

Tabu search

O algoritmo em um 100, 10000 e 100000 apresentou bastante variações em suas iterações porém a média de melhor estado sempre ficou a mesma ou aproximada. Nas iterações sempre apresentou o melhor estado também nas primeiras iterações, porém foram mais rápidas do que a técnica de Hill Climbing.

Tendo como referência esses dados definidos os melhores resultados obtidos foram:

VL:450 - MOCHILA:| TIPO:1 | QT:0 || TIPO:2 | QT:2 || TIPO:3 | QT:5 | KG: 20

VL:400 - MOCHILA:| TIPO:1 | QT:0 || TIPO:2 | QT:2 || TIPO:3 | QT:4 | KG: 18

VL:390 - MOCHILA:| TIPO:1 | QT:1 || TIPO:2 | QT:1 || TIPO:3 | QT:5 | KG: 18

VL:380 - MOCHILA:| TIPO:1 | QT:2 || TIPO:2 | QT:2 || TIPO:3 | QT:2 | KG: 20

CONCLUSÃO

Conclui-se que ambas as técnicas Hill Climbing e Tabu search foram aplicadas e trouxeram resultados satisfatórios em relação a estratégia usada como Heurística Grasp para a resolução do problema da mochila, que tratou de forma a encontrar melhores estados de acordo com a probabilidade de cada item disponível para ser utilizado na busca de um estado válido.

Como resultado foi percebido uma diferença em quantidade de iterações para se chegar no melhor estado onde a técnica de Tabu search foi a melhor, porém é levado em consideração que a técnica implementa uma estratégia que leva vantagem a técnica de Hill Climbing, mas por outro lado ambas ficaram com resultados parecidos.

REFERÊNCIAS

Mochila booleana. Acesso em 24 de Novembro de 2018 https://www.ime.usp.br/~pf/analise_de_algoritmos/aulas/mochila-bool.html

Universidade Estadual de Campinas Instituto de Matemática, Estatística e Computação Científica. Problema da Mochila. Rubens Carvalho. Acesso em 24 de Novembro de 2018 <http://www.ime.unicamp.br/~eabreu/Projeto/RubensCarvalhoRA122181-MS777.pdf>

Programação dinâmica - o problema da mochila. Acesso em 24 de Novembro de 2018 <https://www.youtube.com/watch?v=F-wu-wkdv-M>

Clever Algorithms: Nature-Inspired Programming Recipes. Jason Brownlee PhD. Acesso em 24 de Novembro de 2018 http://www.cleveralgorithms.com/nature-inspired/stochastic/tabu_search.html