

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
Programa de Graduação em Engenharia de Software

André Fernandez Mendes
Leonardo Augusto Pereira do Carmo

RELATÓRIO DO TRABALHO PRÁTICO

Belo Horizonte
2024

André Fernandez Mendes
Leonardo Augusto Pereira do Carmo

RELATÓRIO DO TRABALHO PRÁTICO

Trabalho apresentado à disciplina
Fundamentos de Projeto e Análise de
Algoritmos da Graduação de Engenharia
de Software da Pontifícia Universidade
Católica de Minas Gerais

Professor: João Caram

Belo Horizonte
2024

RESUMO

Este trabalho trata do estudo de problemas intratáveis, tipicamente pertencente às classes NP, e nas técnicas de projeto de algoritmos que podem nos ajudar a encontrar soluções de compromisso adequadas.

São duas as tarefas do grupo:

- Projetar e implementar uma solução para o problema apresentado utilizando backtracking. A solução deve incluir uma estratégia de poda para soluções não promissoras. Gerar conjuntos de teste de tamanho crescente, a partir de 10 interessadas e incrementando de 1 em 1, até atingirmos tamanho T que não consiga ser resolvido em até 30 segundos pelo algoritmo. Na busca do tempo limite de 30 segundos, faça o teste com 10 conjuntos de cada tamanho, contabilizando a média das execuções.
- Projetar e implementar uma solução para o problema apresentado utilizando programação dinâmica. O grupo deverá decidir se vai utilizar o método demonstrado em aula ou outro à escolha. Para este teste, utilize os mesmos conjuntos de tamanho T encontrados no backtracking. Em seguida, aumente os tamanhos dos conjuntos de T em T até atingir o tamanho 10T, sempre executando 10 testes de cada tamanho para utilizar a média

SUMÁRIO

Nenhuma entrada de sumário foi encontrada.

1. SOLUÇÃO DO PROBLEMA UTILIZANDO BACKTRACKING

O backtracking é uma técnica de projeto de algoritmos que consiste no refinamento da busca exaustiva/ força bruta, pois não testa todas as soluções, eliminando algumas sem examinar. A técnica consiste em realizar uma busca em profundidade e, quando a busca falha ou não tem como continuar, retorna pelo mesmo caminho percorrido com a finalidade de encontrar soluções alternativas.

Para aplicação da técnica em questão, é necessário a implementação de uma estrutura de dados de controle.

Conforme estabelecido no problema, foram cadastrados ...

A estratégia de poda utilizada consiste na comparação dos valores obtidos em cada combinação de venda, podando quando a soma dos megawatts disponíveis ultrapassa a quantidade existente.

Assim, evita-se a venda o cálculo de combinações que ultrapassam a quantidade de megawatts que pode ser vendida.

Após medir o tempo de execução de conjuntos de tamanho crescente, iniciando com 10 interessadas e aumentando o número de interessadas até que o tempo médio do algoritmo excedesse 30 segundos, chegou-se a conclusão que podem ser combinadas

2. SOLUÇÃO DO PROBLEMA UTILIZANDO PROGRAMAÇÃO DINÂMICA

Modelagem do Problema:

- O problema foi modelado de forma semelhante ao problema clássico da mochila, onde cada empresa representa um item, a quantidade de megawatts representa o peso do item, e o valor da oferta representa o valor do item.

Estrutura de Dados:

- Utilizamos um array dp de tamanho capacidade + 1 para armazenar o valor máximo que pode ser obtido para cada capacidade de megawatts, onde capacidade é o total de megawatts disponíveis para venda.
- Utilizamos uma matriz taken para rastrear quais empresas foram selecionadas na solução ótima.

Atualização dos Estados:

- Para cada empresa, atualizamos o array dp de forma que dp[j] represente o valor máximo obtido ao considerar a inclusão ou exclusão da oferta da empresa atual.

FUNCIONAMENTO DO ALGORITMO

Inicialização:

- Inicializamos o array dp com zeros, representando o valor inicial para cada capacidade de megawatts.

Processamento das Empresas:

- Para cada empresa, percorremos o array dp de trás para frente (para evitar a reutilização dos mesmos itens na mesma iteração) e atualizamos o valor máximo que pode ser obtido para cada capacidade de megawatts.

Rastreamento das Empresas Selecionadas:

- Após preencher o array dp, utilizamos a matriz taken para rastrear quais empresas foram selecionadas na solução ótima.

Construção da Solução Ótima:

- Percorremos a matriz taken para construir a lista de empresas selecionadas e calcular o valor total obtido.

Comparação de Resultados

Tempo de Execução

A programação dinâmica se mostrou eficiente em termos de tempo de execução, especialmente quando comparada ao backtracking. Ao evitar a exploração de soluções não promissoras e utilizando subproblemas já resolvidos, a programação dinâmica reduziu significativamente o tempo necessário para encontrar a solução ótima.

Qualidade do Resultado

A qualidade do resultado obtido pela programação dinâmica foi excelente, atingindo o valor máximo possível de forma consistente. A matriz taken garantiu que a solução encontrada fosse rastreável e verificável, o que é uma vantagem em termos de clareza e confiabilidade.

Considerações Finais

A programação dinâmica provou ser uma técnica eficaz para resolver o problema de venda de energia, oferecendo um balanço ideal entre tempo de execução e qualidade do resultado. A implementação cuidadosa das estruturas de dados e a abordagem sistemática na atualização dos estados garantiram que a solução fosse tanto eficiente quanto precisa.