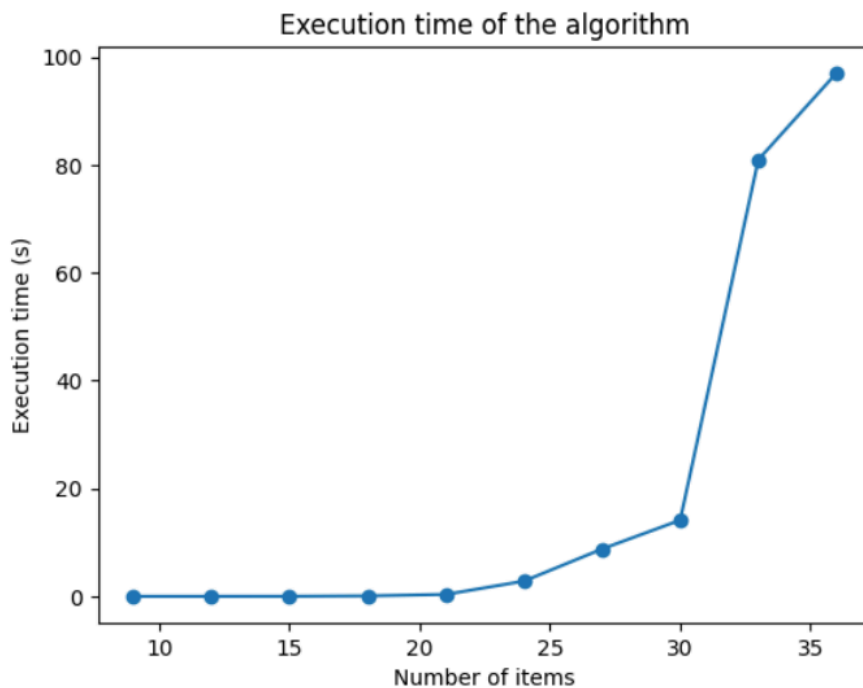


Supercomputação
Entrega - Atividade 6
Arthur Tamm

Introdução

O problema da mochila (Knapsack Problem) é um dos problemas clássicos de otimização combinatória. O objetivo é selecionar um subconjunto de itens com determinados valores e pesos, de forma que o valor total seja maximizado sem que o peso total exceda a capacidade da mochila. A solução exaustiva através de recursão, também conhecida como força bruta, consiste em explorar todas as combinações possíveis de itens para determinar a combinação que maximiza o valor. Neste relatório, iremos implementar e testar a solução recursiva de força bruta utilizando arquivos de entrada que representam diferentes instâncias do problema. Analisaremos o tempo de execução do algoritmo para cada entrada e discutiremos suas limitações, comparando-o com a abordagem iterativa e com uma versão recursiva otimizada com memoization.



Análise do Gráfico de Tempo de Execução

O gráfico acima ilustra o tempo de execução do algoritmo de força bruta recursivo em função do número de itens disponíveis para serem colocados na mochila. Observa-se que, para instâncias com menos de 25 itens, o tempo de execução permanece relativamente baixo, o que é esperado, dado que o número de combinações possíveis ainda é gerenciável. No entanto, à medida que o número de itens aumenta, o tempo de execução cresce exponencialmente. Este comportamento reflete a complexidade do algoritmo, que é $O(2^n)$, onde n é o número de itens. A partir de 30 itens, o tempo de execução sobe drasticamente, evidenciando a inviabilidade da solução exaustiva para grandes entradas.

Discussão sobre as Limitações do Algoritmo

Tempo de Execução

A principal limitação da solução exaustiva através de recursão é o tempo de execução exponencial. Cada item adiciona uma bifurcação no processo de decisão (incluir ou não incluir o item), o que gera 2^n combinações possíveis para n itens. Isso torna a abordagem impraticável para um número grande de itens, como visto no gráfico. Mesmo com otimizações de compilação ou hardware mais rápido, essa complexidade exponencial torna impossível resolver instâncias com muitos itens em tempo hábil.

Uso de Memória

Além do tempo de execução, o uso de memória é outra limitação significativa. A cada chamada recursiva, o algoritmo ocupa espaço na pilha de chamadas do programa. Para instâncias maiores, isso pode levar a um stack overflow, especialmente se o ambiente de execução tiver uma pilha limitada. Diferente da abordagem iterativa, que não enfrenta esse problema, a recursão pura pode causar problemas de memória em entradas grandes.

Comparações com a Abordagem Iterativa de Força Bruta

A abordagem iterativa de força bruta compartilha a mesma complexidade de tempo $O(2^n)$ que a abordagem recursiva. Entretanto, a iteração evita o problema de stack overflow, pois não utiliza recursão, e o uso de memória é controlado de forma mais eficiente. Porém, em termos de tempo de execução, ambas as abordagens são igualmente ineficazes para grandes entradas.

Comparações com a Abordagem Recursiva com Memoization

Ao introduzir memoization na solução recursiva, evitamos recalcular subproblemas que já foram resolvidos, reduzindo drasticamente o número de chamadas recursivas necessárias. Isso transforma a complexidade de tempo de $O(2^n)$ para $O(n \cdot W)$, onde W é a capacidade da mochila. Esta abordagem é muito mais eficiente em termos de tempo de execução, tornando viável resolver instâncias maiores do problema. No entanto, a memoization requer espaço adicional para armazenar os resultados dos subproblemas, o que aumenta o uso de memória, mas de uma maneira muito mais controlada e previsível do que o uso de memória na solução recursiva pura.

Conclusão

A solução recursiva de força bruta para o problema da mochila, embora garantida para encontrar a solução ótima, possui limitações significativas em termos de tempo de execução e uso de memória, especialmente para um grande número de itens. A abordagem iterativa de força bruta resolve o problema de uso de memória da recursão, mas não melhora o tempo de execução exponencial. Por outro lado, a introdução de memoization na abordagem recursiva transforma significativamente a complexidade de tempo, tornando-a uma opção muito mais viável para instâncias maiores do problema. Portanto, embora a solução recursiva pura seja uma boa ferramenta educacional para entender o problema, ela

é impraticável para aplicações reais em grande escala, onde abordagens mais sofisticadas são necessárias.