

Algoritmos Aproximados para Problemas NP Completos

Parte 2

Introdução

- Problemas intratáveis ou difíceis são comuns na natureza e nas áreas do conhecimento.
 - “Fáceis” → resolvidos por algoritmos polinomiais
 - Busca Binária: $O(\log n)$, Pesquisa Sequencial: $O(n)$, Ordenação por Merge Sort: $O(n \log n)$
 - “Difíceis” → resolvidos por algoritmos exponenciais

Problemas Exponenciais

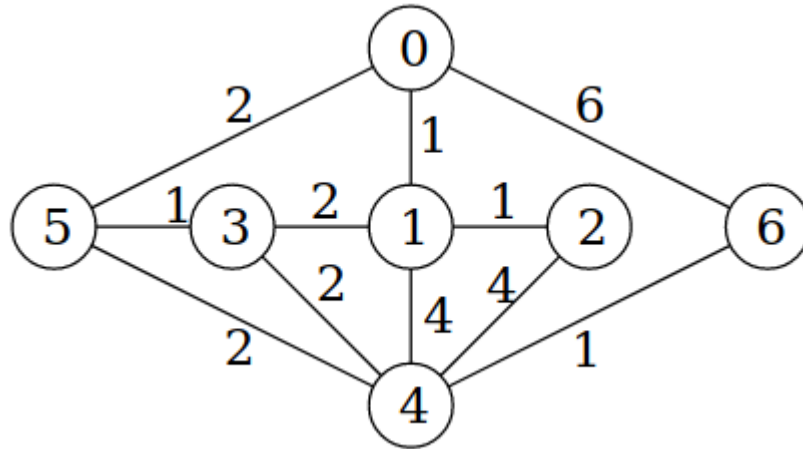
- É desejável resolver instâncias grandes de problemas de otimização em tempo razoável.
 - Para um algoritmo que execute em tempo proporcional a 2^N , não é garantido obter resposta para todos os problemas de tamanho $N \geq 100$.

O que fazer?

- Usar algoritmos exponenciais “eficientes” aplicando técnicas de tentativa e erro.
- Usar algoritmos aproximados.
 - Acham uma resposta que pode não ser a solução ótima, mas é garantido ser próxima dela.
- Concentrar no caso médio.
 - Buscar algoritmos melhores que outros neste quesito e que funcionem bem para as entradas de dados que ocorrem usualmente na prática.

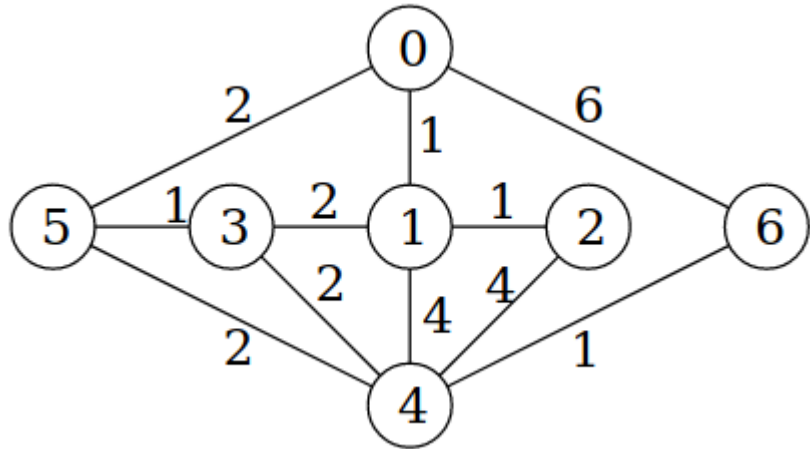
Ciclo Hamiltoniano

- Caminho que permite passar por todos os vértices de um grafo G , não repetindo nenhum.



Ciclo Hamiltoniano

- A rigor, o melhor algoritmo conhecido resolve o problema tentando todos os caminhos possíveis.



Existem duas respostas:

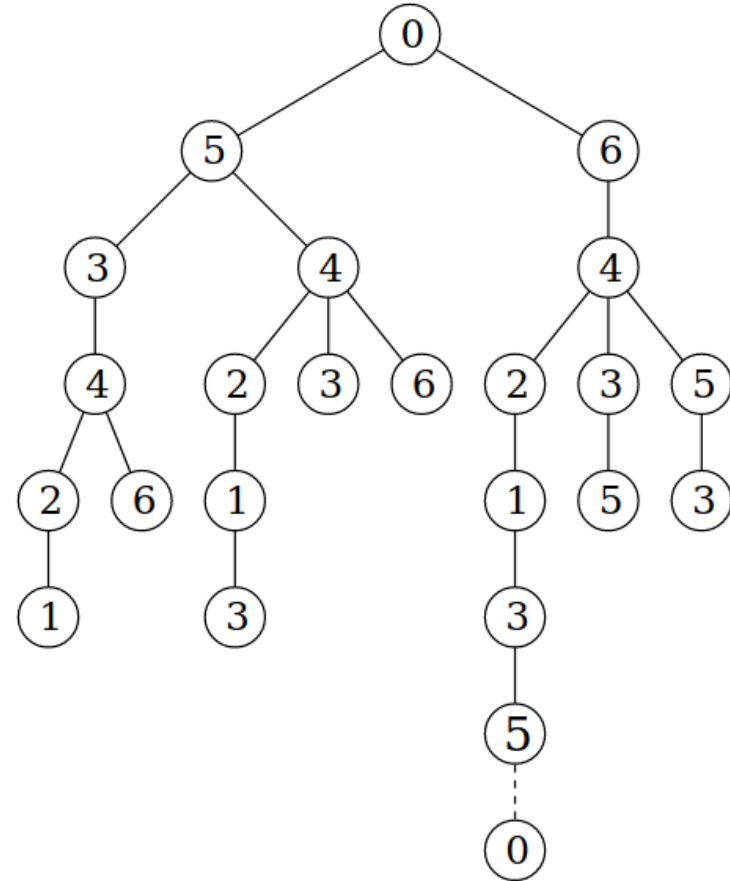
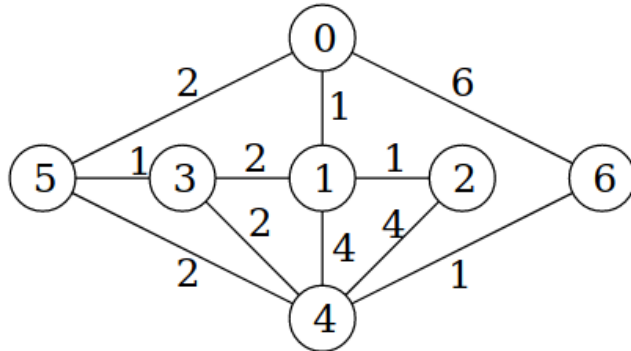
0 5 3 1 2 4 6 0
0 6 4 2 1 3 5 0

Ciclo Hamiltoniano

- Diminuir número de chamadas fazendo “**poda**” na árvore de caminhamento.
 - No exemplo anterior, cada ciclo é obtido duas vezes, caminhando em ambas as direções.

Ciclo Hamiltoniano

- Insistindo que o nó 2 apareça antes do 0 e do 1, não precisamos chamar Visita para o nó 1 a não ser que o nó 2 já esteja no caminho.

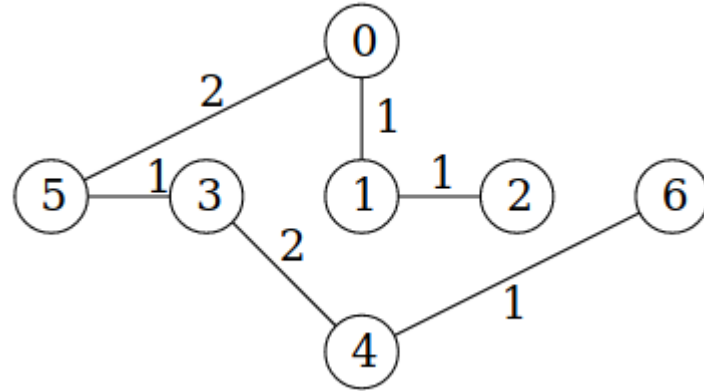
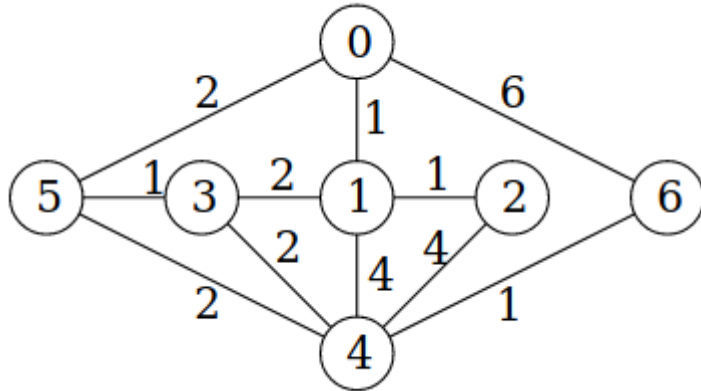


Algoritmo Aproximado

- Não precisa encontrar a solução ótima
- Necessário monitorar a solução encontrada
 - Razão de aproximação $R_A(I) = \frac{S(I)}{S^*(I)}$,
 - Limite inferior e superior

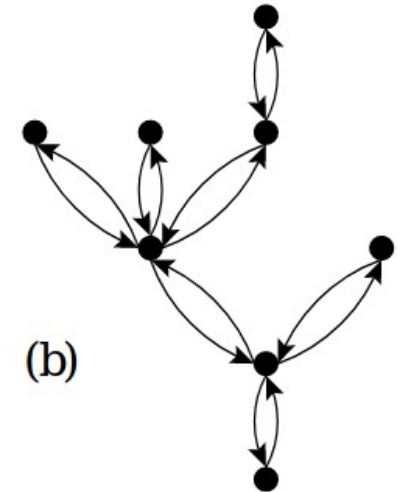
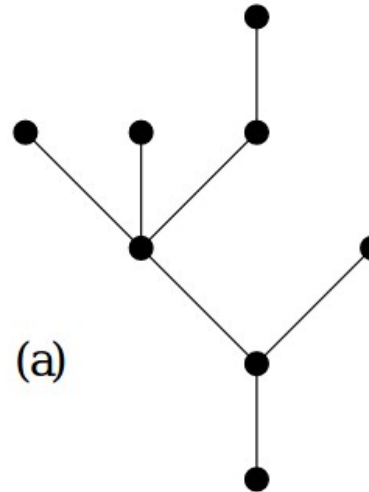
Exemplo

- Problema do Caixeiro Viajante (PCV)
- Limite inferior?
 - Algoritmo de Dijkstra (árvore geradora mínima - AGM)



Algoritmo Aproximado

- Problema do Caixeiro Viajante (PCV)
- Limite superior?
 - Busca em Profundidade na AGM
 - Visita todos os vértices
 - Nenhuma aresta é visitada mais que 2 vezes



Algoritmo Aproximado

- Inicie em uma folha da AGM, mas sempre que a busca em profundidade for voltar para uma cidade já visitada, salte para a próxima ainda não visitada.

