

Área de Otimização Combinatória

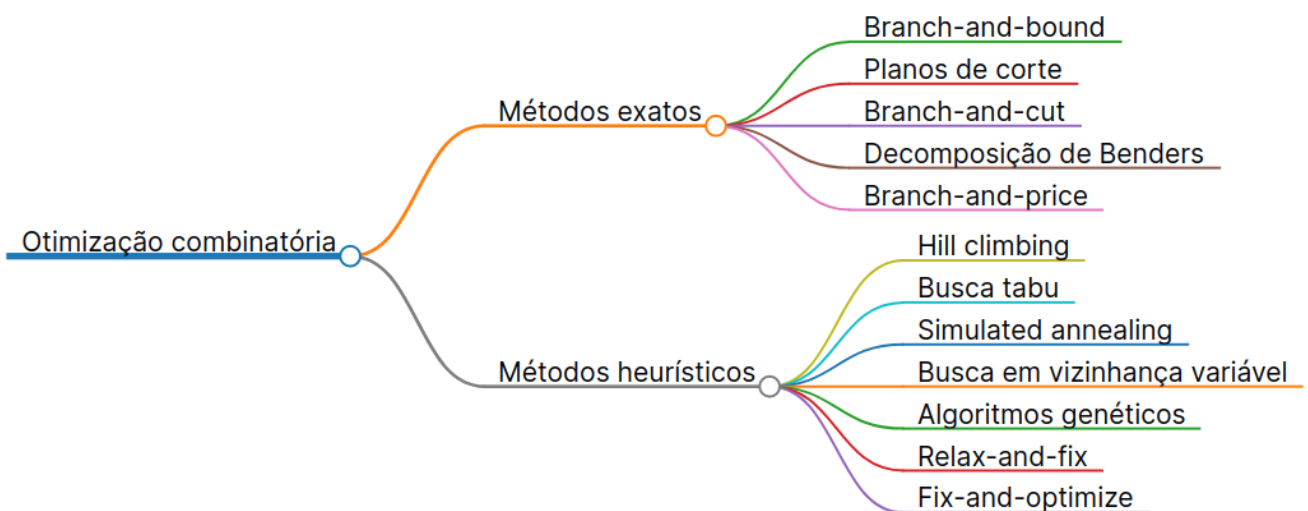
Problemas combinatórios consistem em encontrar um (ou vários) elementos de um conjunto que satisfaz (em) restrições pré-definidas. Existem alguns tipos de problemas diferentes:

1. Encontrar uma solução satisfazendo um conjunto de restrições.
2. Encontrar todas as soluções satisfazendo um conjunto de restrições.
3. Encontrar uma solução ótima de acordo com uma função objetivo.

Problemas de Otimização Combinatória são problemas combinatórios que potencialmente admitem mais de uma solução e essas soluções são diferenciadas por uma medida de qualidade, a função objetivo.

Diversos problemas de otimização combinatória são NP-difíceis. Simplificadamente, isso faz com que o tempo de resolução cresça exponencialmente com o tamanho do problema. Por isso, algoritmos baseados na enumeração explícita de cada solução viável conseguem resolver apenas instâncias pequenas de tais problemas. Por isso, frequentemente, instâncias de interesse prático, demandam por estratégias mais sofisticadas de solução.

Duas dessas estratégias são: métodos exatos e métodos heurísticos. Os métodos exatos, como o branch-and-cut, são métodos que garantem a otimalidade da solução encontrada desde que uma demanda por recursos computacionais (tempo e memória) possa ser atendida. Métodos heurísticos são métodos que relaxam a busca por soluções ótimas em prol da obtenção de uma solução boa em tempo computacional reduzido.



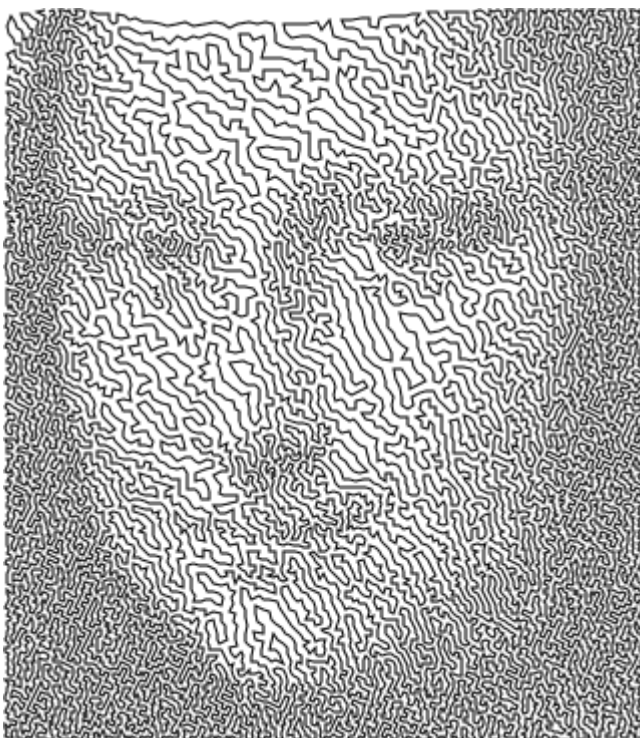
Dentre os métodos exatos está o arcabouço conhecido como branch-and-cut. A implementação de algoritmos de branch-and-cut para um problema costuma ser não trivial. Para garantir a eficiência de tais algoritmos é necessário combinar estratégias de pré-processamento, exploração de árvores de branch-and-bound, planos de corte, entre outros. Nesse caso, é necessário construir problemas auxiliares para a obtenção de limitantes duais para o problema a ser resolvido. O arcabouço de branch-and-cut é de tamanha importância que há diversos pacotes computacionais que fornecem uma implementação pronta como Gurobi, SCIP, CPLEX e Cbc. Para utilizar tais pacotes é necessário conseguir escrever o problema utilizando o paradigma de Programação Linear Inteira.

Métodos de solução heurísticos relaxam a garantia de otimalidade da solução em prol de reduzir os recursos computacionais necessários, em geral, deseja-se obter uma solução de boa qualidade (não necessariamente ótima) mais rapidamente. Existem diversas estratégias heurísticas, meta-heurísticas, que podem ser aplicadas a inúmeros problemas. Entre eles estão algoritmos genéticos, algoritmos de colônias de formigas, otimização por enxame de partículas, busca tabu, simulated annealing, iterated local search, variable neighborhood search, e muitos outros.

Exemplos de problemas de Programação Linear Inteira

O problema do caixeiro viajante

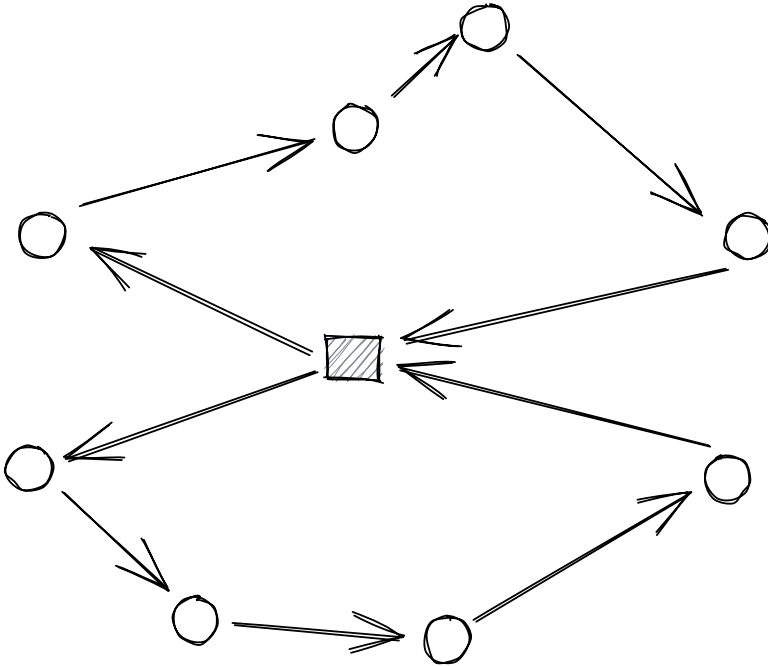
Um vendedor deseja visitar as cidades $0, 1, 2, \dots, n - 1$, partindo de uma delas. O custo de viajar de uma cidade i para uma cidade j é c_{ij} , $i, j = 0, 1, \dots, n - 1$. O problema de determinar uma rota com o menor custo que visita todas as cidades exatamente uma vez é conhecido como Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP). O TSP é um dos problemas mais estudados da computação.



(Fonte https://wiki.evilmadscientist.com/TSP_art)

O número de soluções viáveis do TSP é $O(n!)$ e algoritmos ingênuos de enumeração explícita, conseguem resolver instâncias com poucas dezenas de cidades em tempo viável. Com algoritmos de Programação Inteira, é possível encontrar soluções garantidamente ótimas para instâncias com [dezenas de milhares de pontos](#).

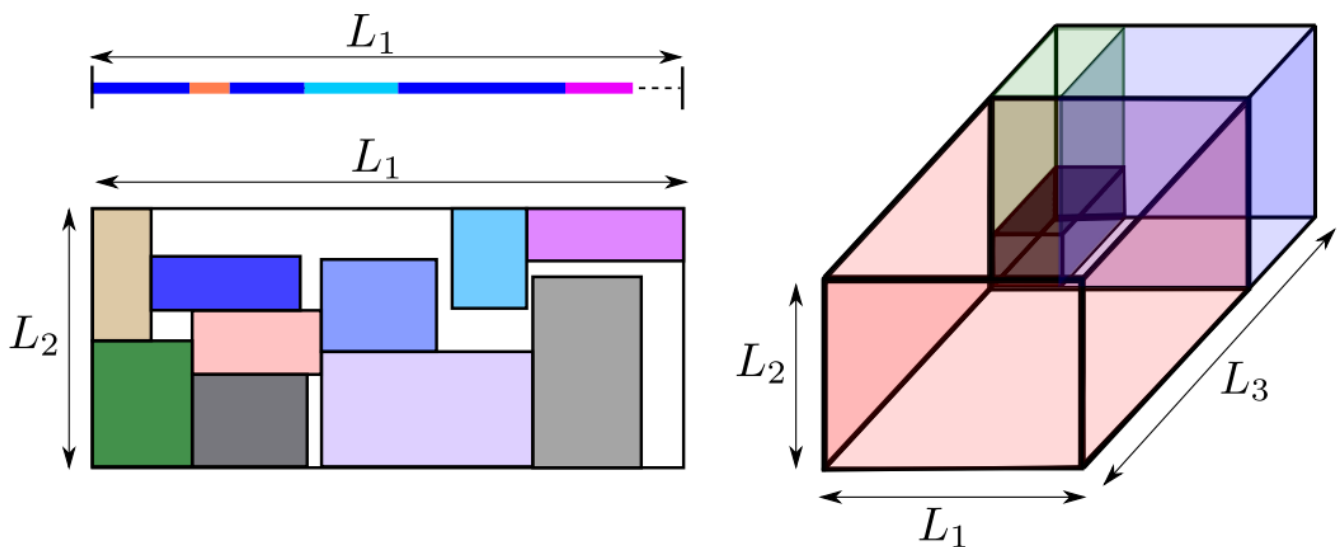
O TSP possui diversas aplicações ([inclusive artísticas](#)) e pode ser estendido para considerar veículos com limite de capacidade, janelas de tempo para visitar as cidades, demandas variáveis em cada cidade, entre outros. De forma geral, tais problemas são chamados de Problemas de Roteamento de Veículos.



Problemas de corte e empacotamento

Considere I objetos retangulares, cada objeto é definido por sua largura e altura, (a_i, b_i) , respectivamente. É necessário escolher um subconjunto $S \subseteq I$ de retângulos e posicionar cada elemento de S em um recipiente retangular com dimensões (L_1, L_2) de forma a minimizar a área desocupada do recipiente.

Há variações do problema que são unidimensionais, tridimensionais ou multidimensionais, com objetos e/ou recipientes não retangulares. Problemas de corte e empacotamento são valiosos para indústrias de papel, vidro, cerâmica, móveis entre outras.



Fonte: BELIN CASTELLUCCI, Pedro. **Consolidation problems in freight transportation systems: mathematical models and algorithms**. 2019. Tese de Doutorado.

Quero fazer um Trabalho de Conclusão de Curso ou Iniciação Científica na área, o que preciso?

Tipicamente, é necessária uma curiosidade pelas áreas de algoritmo, programação e matemática. De uma forma geral, um projeto na área é feito para estudar um problema de Otimização Combinatória. Inicialmente, é feita uma revisão da literatura para identificação de abordagens interessantes. Então, são desenvolvidas soluções, realizados experimentos computacionais e redigidos relatórios para documentação e divulgação das atividades.

Caso tenha interesse, me escreva para conversarmos!

{Nome}.{Sobrenome} (at) ufsc.br

Alguns links interessantes

- Palestra do Professor [Marco Lubbecke](#) no TEDx:
<https://www.youtube.com/watch?v=Dc38La-Xvog>
- Aula do Professor [Pedro Munari](#) sobre o que é Pesquisa Operacional:
<https://www.youtube.com/watch?v=KeJBy8Jbcg4>
- Aula do Professor [Mário Felice](#). O que é Otimização Combinatória:
https://www.youtube.com/watch?v=ebB_U7z6Alo
- Seminário de coisas legais: <https://www.youtube.com/watch?v=nPHiTBJCeK0>