

# Inteligência Artificial - Metaheurísticas

Produção e Fundamentos da Indústria 4.0

Carlos Diego Rodrigues

Federação das Indústrias do Estado do Ceará - FIEC

4 de junho de 2022

# Pesquisa Operacional



# Agenda

Representação de soluções

Construindo soluções

Busca e vizinhança

Metaheurísticas de busca local

Algoritmo guloso

Busca tabu

Recozimento simulado

Metaheurísticas evolutivas

Algoritmo genético

Enxame de partículas

# Problemas reais... soluções reais

- Os problemas reais pedem soluções reais.
- Contudo temos um desafio de lidar com situações extremamente complexas, combinatórias que requerem o apoio do computador.
- Precisamos, portanto, frequentemente, representar o problema no computador antes de resolvê-lo, assim como suas soluções.
- Esse processo de tradução do problema real em um problema matemático, resolução do problema e tradução da solução para soluções reais é o objeto de estudo da Pesquisa Operacional.

# Soluções no computador

- Um dos principais desafios para resolver um problema é saber enunciá-lo.
- Outro desafio é entender como se configuram as soluções daquele problema.
- Não estamos falando exclusivamente da melhor solução, mas das soluções do problema de forma geral.
- Soluções podem ser
  - Inviáveis.
  - Viáveis.
  - Ótimas.
  - *"Quase" ótimas.*
- As soluções são representações das **variáveis** mais importantes do problema.
- Objetivo do problema.

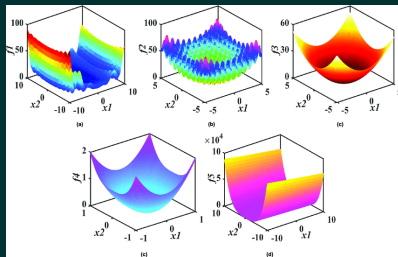
# Restrições

- As relações entre as variáveis são chamadas de restrições.
- Frequentemente essas relações servem para limitar as possibilidades de valoração das variáveis.
- Restrições podem ser
  - Duras: uma solução viável deve respeitar essas restrições.
  - Suaves: uma solução ótima deve respeitar essas restrições.
- Pesquisa operacional está interessada em:
  - Função objetivo.
  - Variáveis de decisão.
  - Restrições do problema.

# Contínuo *versus* Combinatório

## ■ Contínuo

- Encontrar os melhores valores em um intervalo.
- Minimização de funções complexas.
- Equações diferenciais.



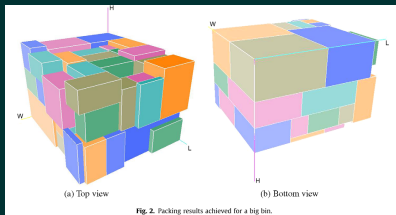
# Contínuo *versus* Combinatório

- Combinatório

- Escolher entre opções.

- Roteamento.

- Empacotamento.





## *P versus NP*

- A dificuldade dos problemas se dividem em classes.
- Problemas na classe *P* são problemas para os quais são conhecidos um algoritmo de tempo polinomial.
- Problemas na classe *NP* podem ter várias dificuldades.
- Mas existe uma subclasse de *NP* chamada *NP – completo* para os quais não se conhece um algoritmo polinomial para qualquer problema nessa classe.
- *P versus NP*: consistem em encontrar ou mostrar que não existe um algoritmo polinomial para qualquer problema desta classe.
- <https://www.youtube.com/watch?v=YX40hbAHx3s>

Este problema está em aberto.

# Boas soluções

- Em nosso curso vamos elaborar metodologias para encontrar boas soluções, mas que não necessariamente são as melhores soluções.
- Boas estratégias são chamadas *heurísticas*. Elas sempre são muito bem-vindas na pesquisa operacional!
- Metaheurísticas são estratégias para gerar heurísticas: arcabouços gerais que podem funcionar para qualquer problema.

# Agenda

- Representação de soluções

- Construindo soluções

  - Busca e vizinhança

- Metaheurísticas de busca local

  - Algoritmo guloso

  - Busca tabu

  - Recozimento simulado

- Metaheurísticas evolutivas

  - Algoritmo genético

  - Enxame de partículas

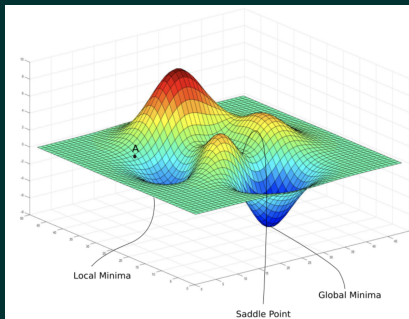
# Heurísticas construtivas

- Uma primeira tentativa para obter soluções para um problema é tentar construir a partir de um conhecimento prévio.
- Estas estratégias nas quais as variáveis são valoradas sequencialmente seguindo esse conhecimento são chamadas heurísticas construtivas.
- Muitas vezes as metaheurísticas funcionam melhor quando partem de uma boa solução.
- Alguns problemas (especialmente aqueles que apresentam restrições duras) podem impor grande dificuldade na tentativa de elaborar-se heurísticas construtivas.

# Exemplo

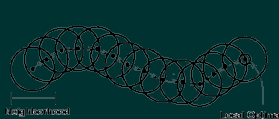
# Busca local *versus* busca global

- Idealmente queremos encontrar a melhor solução de todo o problema.
- Contudo é difícil indicar um caminho razoável para o computador calcular a rota para atingir o **ótimo global**.
- Nosso procedimento de busca pode nos levar a encontrar **ótimos locais**: soluções para as quais a vizinhança não oferece uma melhor alternativa.

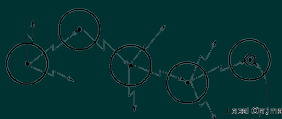


## Passando de uma solução para outra

- Uma vez que temos uma solução inicial  $s$  podemos elaborar operadores que nos permitam construir uma nova solução a partir daquela.
- Este processo é chamado busca.
- As soluções candidatas, enumeráveis a partir de uma determinada solução  $s$ , são chamadas de vizinhança de  $s$ .
- Mais de um operador pode ser utilizado para construir uma vizinhança.
- Este processo pode ser *determinístico* ou *estocástico*.
- Conhecido como busca local ou busca por vizinhança.



(a) Basic Iterative Improvement



(b) Variable Depth Search

# Exemplo



# Técnicas de busca local

- Algoritmo guloso
- Busca tabu
- Têmpera simulada

# Técnicas de população

- Algoritmo genético
- Otimização de sistemas de partículas

# Agenda

- Representação de soluções

- Construindo soluções

  - Busca e vizinhança

- Metaheurísticas de busca local

  - Algoritmo guloso

  - Busca tabu

  - Recozimento simulado

- Metaheurísticas evolutivas

  - Algoritmo genético

  - Enxame de partículas

# Algoritmo guloso

- Consiste em buscar sempre a melhor solução entre os vizinhos.
- Nunca retroceder!
- Tende a ser um algoritmo rápido, mas para no primeiro mínimo local.
- É usado como base de comparação ou solução inicial para uma heurística mais elaborada

# Busca Tabu

- Realiza uma busca local guardando sempre a melhor solução.
- Caso encontre um mínimo local, aceita soluções piores.
- Para evitar ciclos, guarda uma lista de movimentos proibidos, conhecida como lista tabu.
- Originalmente aplicável para problemas combinatórios, mas possui adaptações para problemas contínuos.
- O tamanho da lista tabu é um parâmetro do método.

# Recozimento simulado

- Realiza uma busca local guardando sempre a melhor solução.
- Possui uma variável do método chamada temperatura.
- Caso encontre um mínimo local, aceita soluções piores com uma probabilidade que depende da temperatura.

se  $r \leq e^{\frac{\Delta E}{T}}$  então aceita a solução.

onde  $\Delta E$  é a diferença entre a melhor solução encontrada e a melhor solução entre os vizinhos.

- O decaimento da temperatura faz o método tender a um mínimo local.
- Difícil parametrização.

# Agenda

- Representação de soluções

- Construindo soluções

  - Busca e vizinhança

- Metaheurísticas de busca local

  - Algoritmo guloso

  - Busca tabu

  - Recozimento simulado

- Metaheurísticas evolutivas

  - Algoritmo genético

  - Enxame de partículas

# Algoritmo genético

- Constrói-se uma população inicial aleatória.
- Sobre esta população incide um processo evolutivo que consiste de dois operadores.
  - **Cruzamento**: dois ou mais elementos aleatórios da população são selecionados e são gerados novos elementos que conservam as características (variáveis) dos pais, mas combinados entre si.
  - **Mutação**: um elemento da população é selecionado e uma característica (variável) deste elemento é modificada.
- Com o advento dos elementos gerados, uma seleção é feita entre os indivíduos para a manutenção do tamanho original da população, de forma que apenas os elementos com melhor avaliação da função objetivo são conservados.
- O processo itera por um número pré-determinado de gerações ou até que as mudanças na função objetivo não sejam mais sensíveis (por um certo número de gerações).



# Otimização de enxame de partículas

- Constrói-se uma população inicial aleatória.
- Cada indivíduo possui um vetor de velocidade: uma direção de mudança e um valor
- A cada iteração opera-se a mudança proposta por cada indivíduo e avalia-se a nova população.
- A direção e a intensidade de cada vetor de velocidade é então atualizada por uma pequena porção aleatória que garante a exploração do espaço de soluções. sensíveis (por um certo número de gerações).