

Universidade Federal do Pará
Instituto de Tecnologia -ITEC
Disciplina: IA Bio-Inspirada e Otimização.

Introdução às metaheurísticas

Profa: Jasmine Araújo.
Prof: Glauco.

Sumário

- Introdução
 - Breve Histórico
- Conceitos básicos
 - Nature Inspired Algorithms
- Considerações finais
- Bibliografia

Introdução



- Problemas de otimização são recorrentes na academia, bem como na indústria.(as meta-heurísticas tem se tornado muito populares, especialmente as de inteligência de enxame)
- Podemos dizer sem exagero que a otimização está presente em tudo do projeto de engenharia ao planejamento de um negócio, e também do roteamento da Internet ao planejamento de um feriado.
- Sempre tentando atingir certos objetivos ou otimizar alguma coisa como lucro, qualidade e tempo.
- Sempre que possível, deve-se buscar solucionar um problema de otimização empregando um método exato.
- Problemas de otimização combinatória geralmente são de difícil resolução.

Introdução

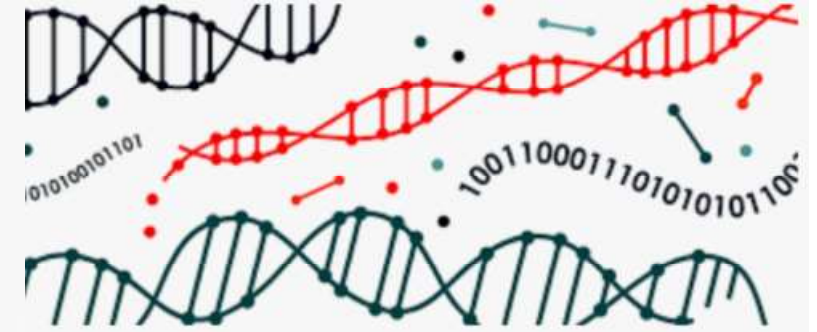


- Problemas de médio e grande porte podem não serem solucionados em tempo computacional admissível por um método exato.
- Natureza multiobjetivo dos problemas de tomada de decisão.
- Resolvedores de otimização (*softwares* comerciais) possuem alto custo.
- A inserção de novas restrições pode ser uma dificuldade para a aplicação de um modelo matemático, bem como para a adoção de métodos exatos para sua resolução.
- A importância de técnicas heurísticas.

Introdução

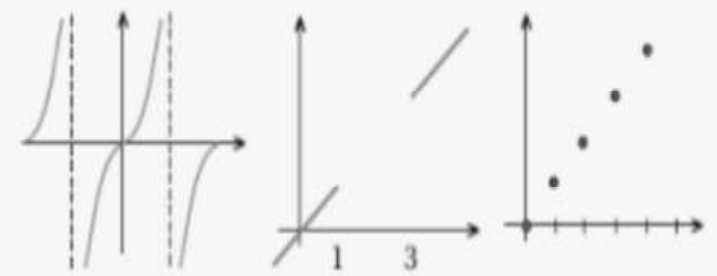
- Breve histórico
- Nossa experiência diária de aprendizagem é com heurística(tentativa e erro)
- Alan Turing foi provavelmente o primeiro a usar heurística para quebrar os códigos alemães na segunda guerra mundial.
- Turing chamou seu método de busca heurística (heuristic search) .
- Não se tinha garantia de encontrar a solução correta, mas foi um sucesso.
- Turing foi então trabalhar no NPL (National Physical Lab) com ideias inovadoras em aprendizado inteligente, redes neurais e algoritmos evolucionários.

Introdução



- Breve histórico
- As décadas de 60 e 70, foram importantes para o desenvolvimento de algoritmos evolucionários.
- John Holland e seus colaboradores desenvolveram na Universidade de Michigan os algoritmos genéticos.
- Em 1962, ele estudou sistema adaptativo e foi o primeiro a usar crossover e recombinação.
- O livro com o resultado da pesquisa com algoritmos genéticos foi publicado em 1975.

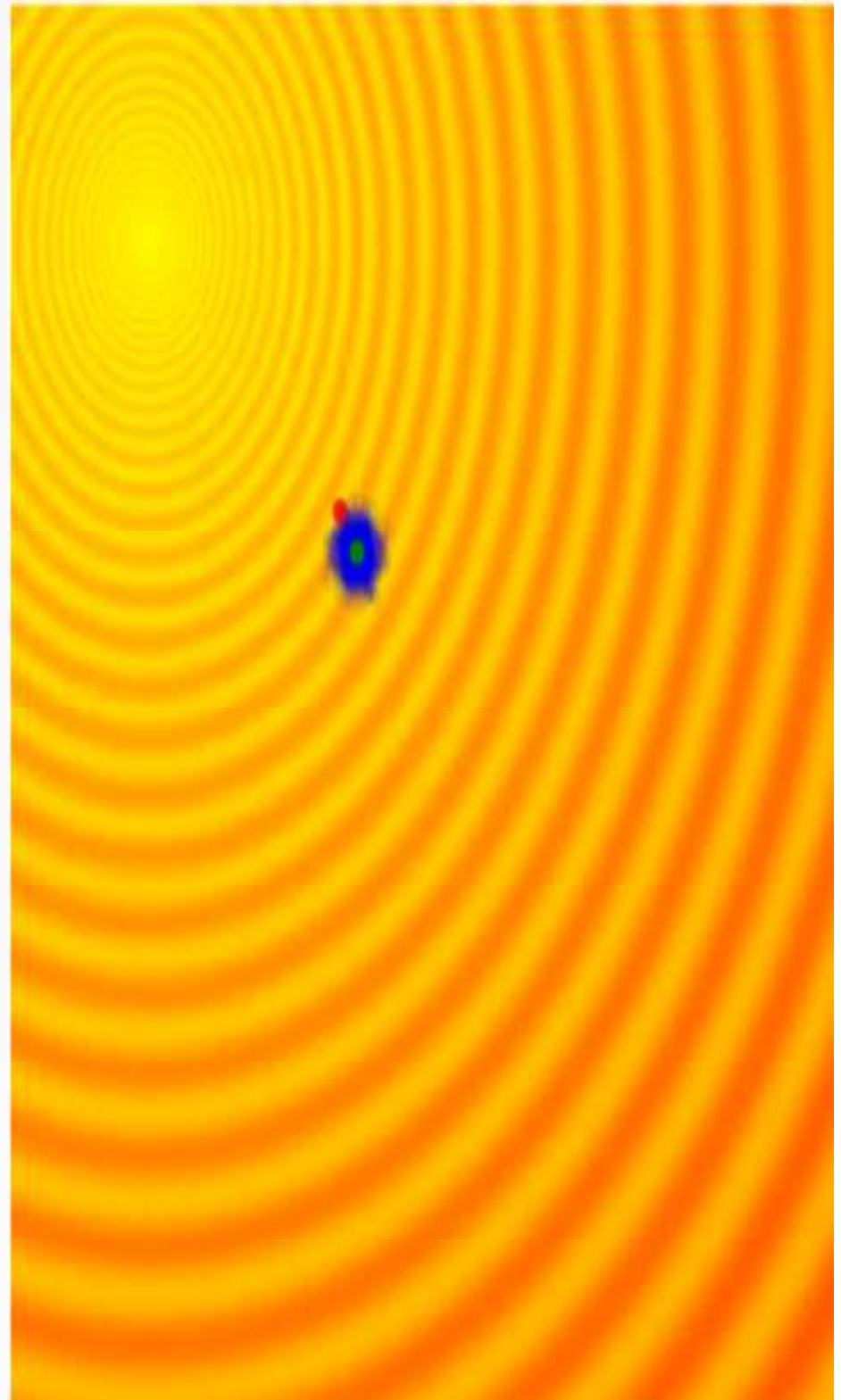
Introdução



- Breve histórico
- No mesmo ano De Jong publicou um artigo demonstrando a capacidade dos AGs de tratar uma grande quantidade de funções objetivos, com ruído, descontínuas e multimodais.
- O AG usa método de busca baseado na abstração da evolução Darwiniana e na seleção natural de sistemas biológicos e representa através de operadores matemáticos: crossover ou recombinação, mutação, fitness e seleção do melhor fitness.
- Desde então se tornaram um sucesso em resolver uma grande quantidade de problemas de otimização.
- Existem milhares de artigos publicados e centenas de livros

Introdução

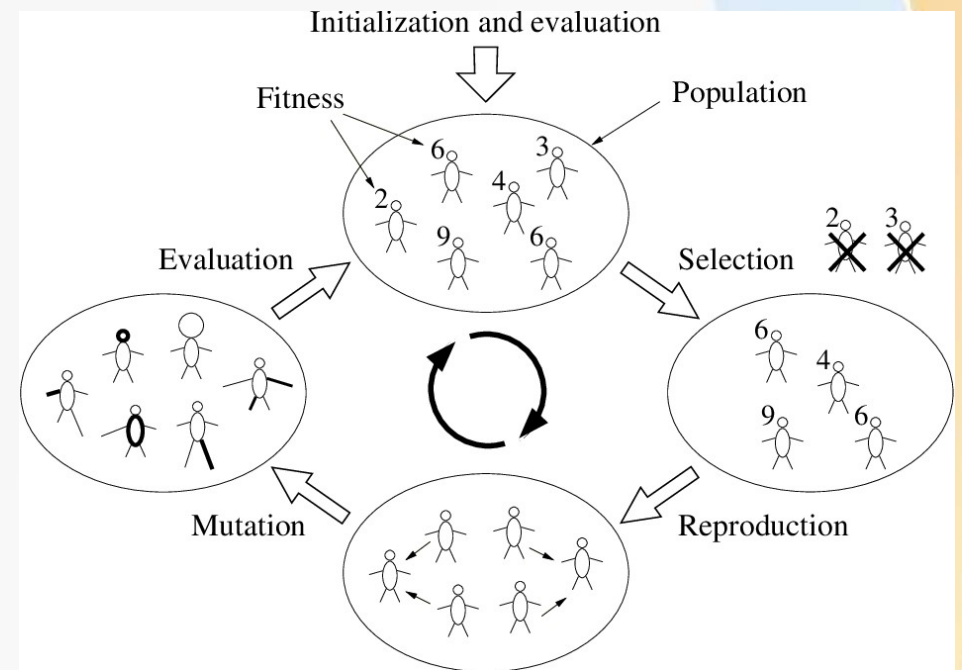
- Breve histórico
- Durante o mesmo período, em 1963, dois pesquisadores: Ingo Rechenberg e Hans-paul Schwefel desenvolveram uma técnica de busca aplicada a engenharia aeroespacial no Universidade Técnica de Berlim chamada estratégia evolucionária.
- Peter Bienert juntou-se a eles e iniciou a construção de um experimentador automático usando regras simples de mutação e seleção (sem crossover).
- Não havia crossover, apenas mutação e a melhor solução era mantida a geração seguinte.
- Isso é essencialmente o algoritmo hill climbing com randomização.



Introdução

- **Breve histórico**

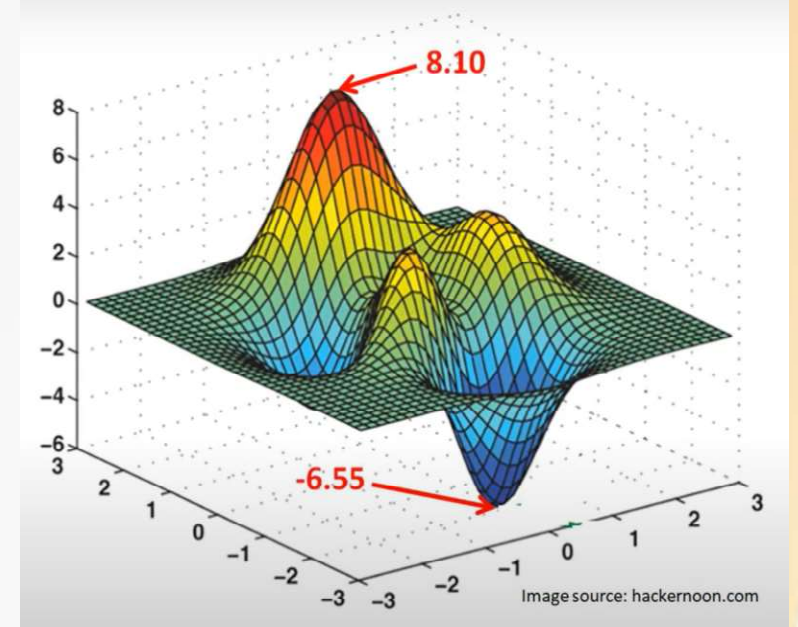
- Em 1960 o pesquisador Fogel usou como processo de aprendizado a evolução simulada como uma ferramenta para estudar a inteligência artificial.



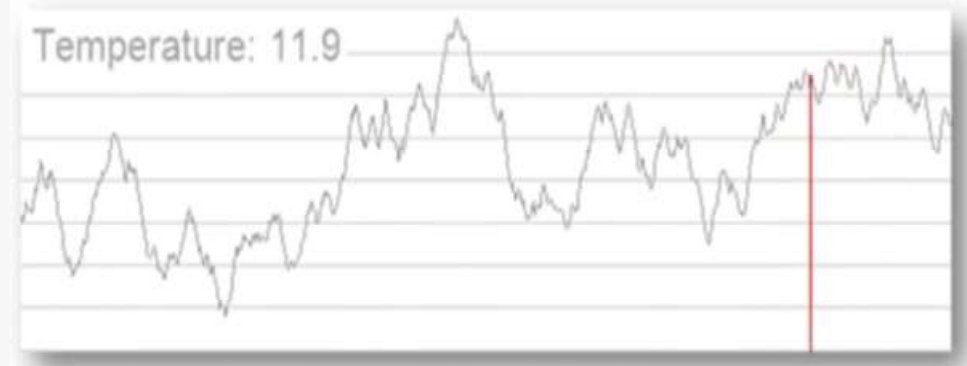
- Em 1966, Fogel, Owen e Walsh desenvolveram a técnica de programação evolucionária para representar soluções como máquinas de estado finitas e aleatoriamente mudando uma dessas máquinas.
- Tudo evoluiu para uma disciplina maior chamada algoritmos evolucionários ou computação evolutiva.
- Outras técnicas podem ser classificadas como técnicas de otimização heurísticas: redes neurais e SVM e etc..

Introdução

- Breve histórico
- Essas técnicas minimizam o erro de aprendizado e/ou erros de predição via iterações de tentativas e erros.
- As décadas de 80 e 90 foram as mais decisivas para as metaheurísticas: o simulated annealing foi proposto em 1983 por Kirkpatrick, Gellat e Vecchi inspirados pelo processo de recozimento de metais.
- O simulated annealing é um algoritmo de busca baseado em trajetória com uma solução inicial aleatória ou definida por tentativa em uma alta temperatura e gradualmente o sistema vai sendo resfriado.
- Um movimento ou uma nova solução é aceita se for melhor.



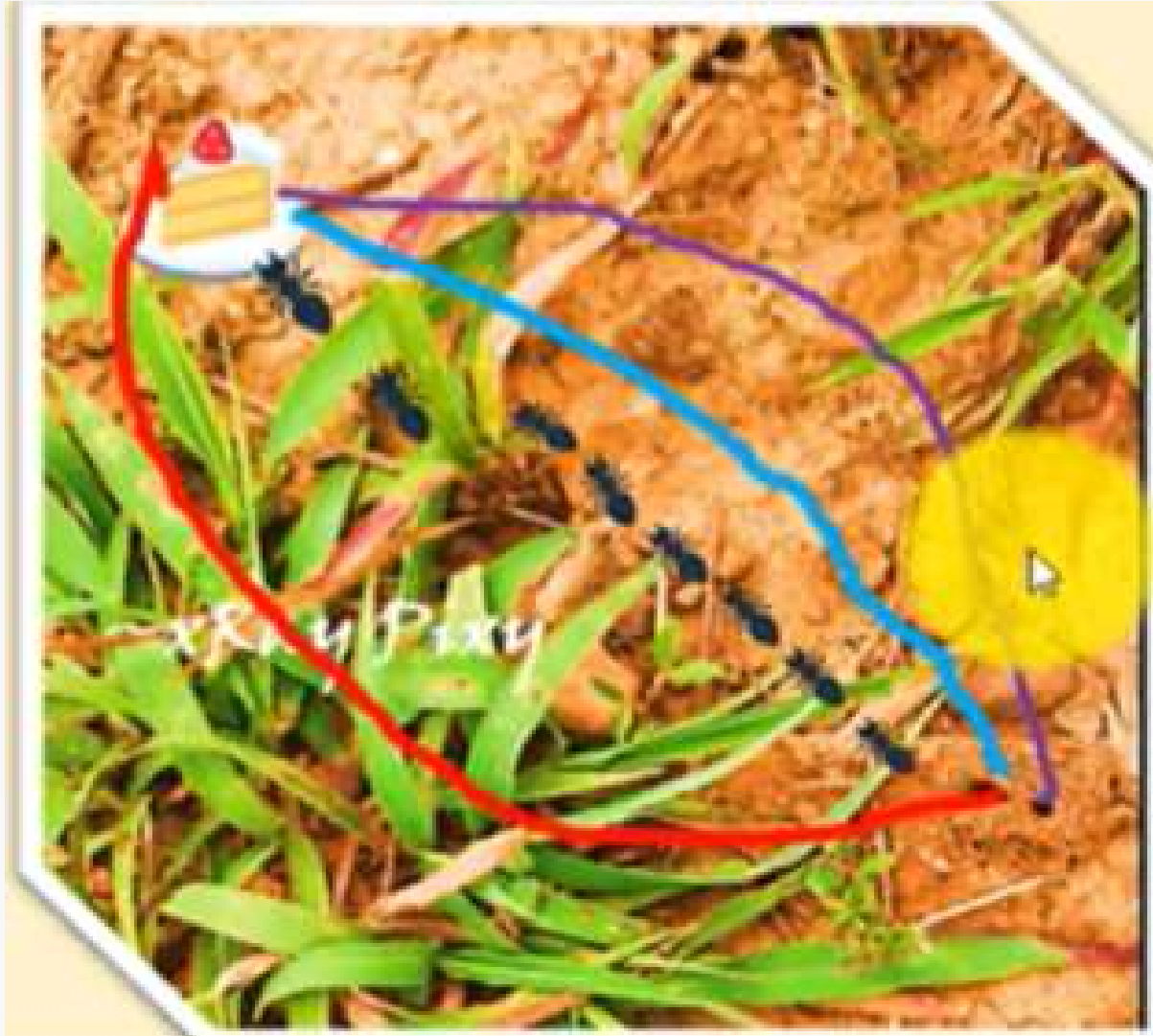
Introdução



- Breve histórico
- De outra maneira, essa solução é aceita mas com um valor de probabilidade associado.
- Essa decisão faz com que o algoritmo tenha grande chance de escapar de ótimos locais já que não descarta totalmente a solução.
- É esperado então que ao resfriar lentamente o sistema, um valor ótimo global seja então encontrado.
- Em 1992, Marco Dorigo terminou sua tese de doutorado propondo o algoritmo ACO (Ant Colony Optimization).
- O ACO é baseado no comportamento social inteligente de enxame das formigas usando feromônio como mensageiro químico.

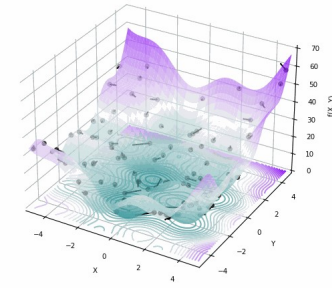
Introdução

- Breve histórico



Introdução

[1/100] w:0.800 - c₁:3.500 - c₂:0.500



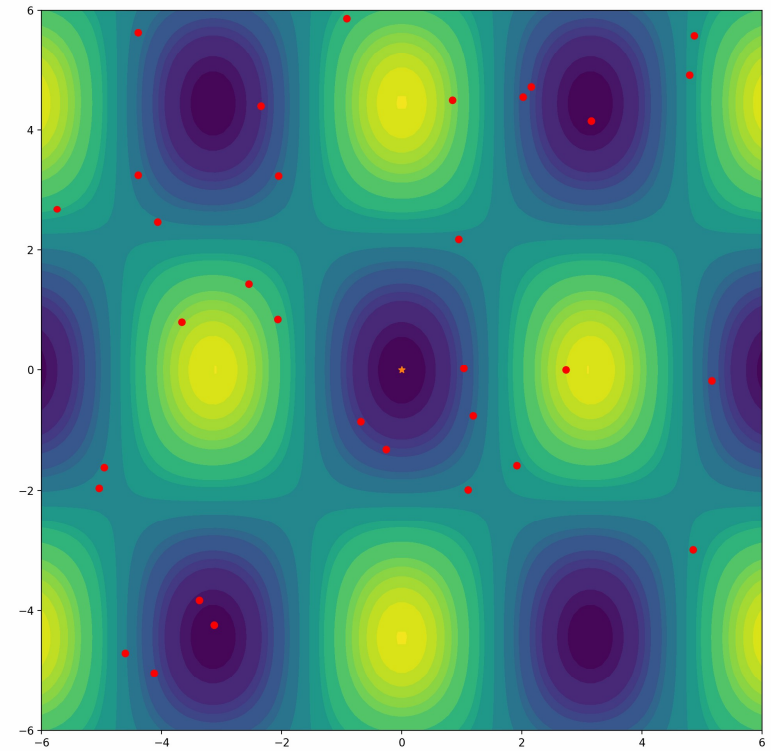
- Breve histórico
- Um pouco antes em 1988, Koza fez sua primeira patente com programação genética. E em 1992 publicou um tratado sobre o assunto.
- O tratado de Koza lançou as bases para as modernas técnicas de aprendizado de máquina.
- A ideia básica era o princípio da genética para criar programas de computador para produzir os melhores programas para um dado tipo de problema.
- Em 1995, um psicólogo americano James Kennedy e um engenheiro Russell C. Eberhart desenvolveram outra importante contribuição que foi o PSO.

Introdução

- Breve histórico
- O PSO é um algoritmo de otimização baseado na inteligência de enxame de peixes e aves e também no comportamento humano.
- Os múltiplos agentes chamados partículas, voam o espaço de busca partindo de valor aleatório inicial.
- O enxame comunica a melhor valor atual e compartilha o melhor global então caminha para soluções que tenham melhor qualidade.
- Há mais de 20 tipos de PSO propostos.
- Existem evidências que o PSO é melhor que os algoritmos de busca tradicionais e em alguns trabalhos melhor resposta que o AG.

Introdução

- Breve histórico
- Em 1996/97 Storn e Price publicaram o differential evolution (DE) e também demonstrou eficiência na resolução de vários problemas melhor que o AG.
- Não existe algoritmo que sirva para todos os problemas.
- A pesquisa consiste em encontrar algoritmos mais eficientes e melhores para um dado problema.
- Como consequência disto precisamos projetar algoritmos melhores para a maior parte dos problemas, não para todos os problemas.
- A pesquisa ainda está ativa nesta área.



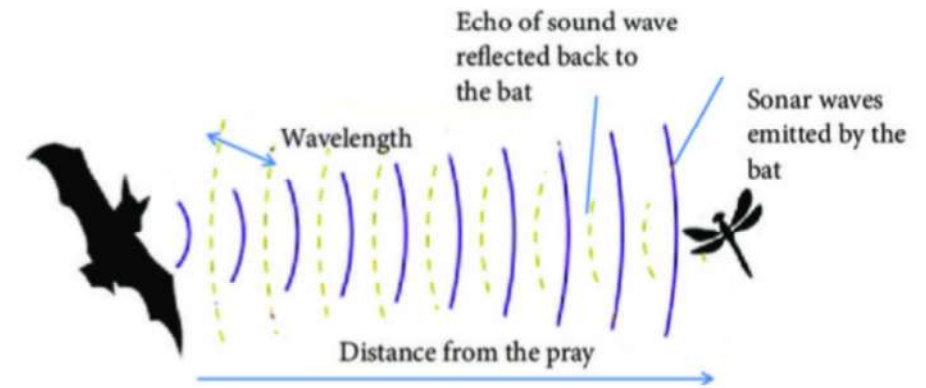
Introdução



- Breve histórico
- Em 2001, foi publicado o Harmony Search (HS) que tem sido aplicado em distribuição de água, modelagem e e agendamento de transportes.
- Em 2004, Nakrani e Tovey publicaram o algoritmo Honey Bee que atendeu bem aos problemas de otimização de centros de Internet e evoluiu em 2005 para um novo algoritmo de abelha e também o pesquisador publicou o ABC (Artificial Bee Colony).
- Em 2008 foi publicado o Firefly Algorithm (FA) por X S Yang.
- Em 2009, Xin She Yang e Suash Deb publicaram o Cuckoo Search (CS) demonstrou ser mais eficiente que muitos algoritmos, inclusive o PSO.

Introdução

- Breve histórico
- Em 2010, Xin S. Yang publicou o BAT Algorithm.
- Em 2012, X S Yang publicou o flower pollination (FPA).
- Outros algoritmos estão sendo publicados até hoje.
- Discutiremos nesta disciplina alguns deles como o Simulated Annealing, PSO e FPA.



Conceitos básicos

- *Heuriskein* (grego) → descobrir.
- Conforme Gigerenzer (2011), heurísticas são processos cognitivos empregados em decisões não racionais, sendo definidas como estratégias que ignoram parte da informação com o objetivo de tornar a escolha mais fácil e rápida.
- Técnicas baseadas na experiência e no bom senso.
Métodos aproximados ou aproximativos (não-exatos).
- Segundo Goldberg *et al.* (2016), espera-se que, em muitos casos, as heurísticas alcancem os valores ótimos da solução de problemas NP-Hard ou mais complexos, especialmente nas ocasiões em que partem de uma solução próxima do valor ótimo.

Conceitos básicos

- Segundo Blum e Roli (2003), uma metaheurística é um conjunto de conceitos os quais podem ser usados para definir métodos heurísticos que podem ser aplicados a uma ampla gama de diferentes problemas.
- Conforme Goldberg *at al.* (2016), uma meta-heurística trata-se de uma arquitetura geral de regras que, formada a partir de um tema em comum, pode servir de base para o projeto de uma ampla gama de heurísticas computacionais.
- Uma metaheurística é um algoritmo modular que pode ser aplicado em diferentes problemas de otimização com poucas modificações a serem realizadas na adaptação a um problema específico.

Conceitos básicos

- Conforme Reeves e Beasley (1995), uma heurística é uma técnica que pesquisa boas soluções (ou seja, próximas do ótimo) com um custo computacional razoável, sem estar apta a garantir a otimalidade, ou a distância à otimalidade de uma dada solução.
- Segundo Goldberg *et al.* (2016), uma heurística é uma técnica computacional aproximativa, que visa alcançar uma solução avaliada como aceitável para um dado problema que pode ser representado em um computador, utilizando um esforço computacional que pode ser considerado como aceitável. Sendo capaz de garantir, em determinadas condições, a viabilidade ou a otimalidade da solução encontrada.

Conceitos básicos

- As Heurísticas usam funções especiais para procurar o espaço de buscar de maneira inteligente.
- Os algoritmos Meta-heurísticos são processo de geração iterativa o qual guia uma heurística subordinada para explorar e intensificar a busca no espaço de busca.
- As estratégias de aprendizado nas meta-heurísticas ajudam a encontrar soluções próximas do ótimo eficientes.

Conceitos básicos

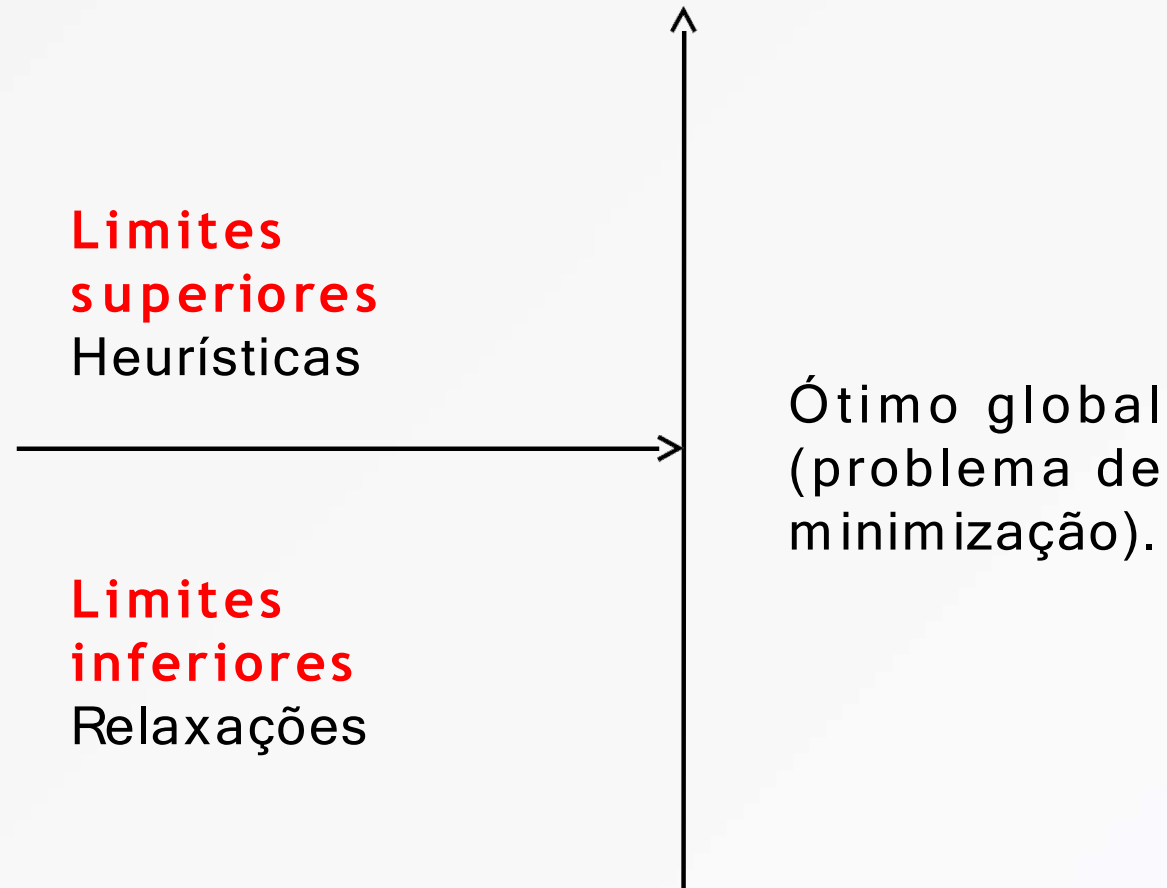


Figura 1: Representação esquemática de um problema de minimização.

Conceitos básicos

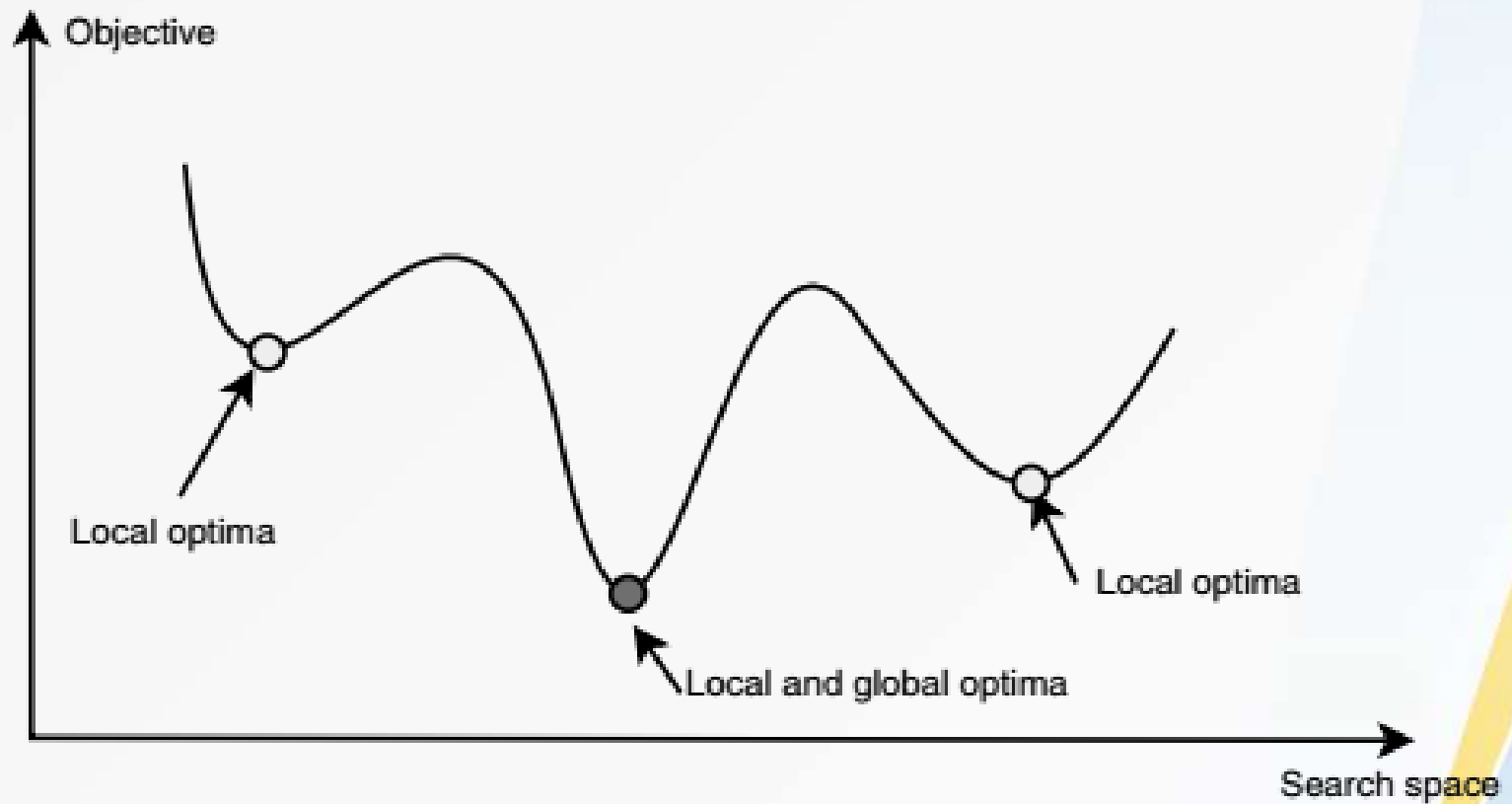


Figura 2: Ótimo local e ótimo global no espaço de busca.

Conceitos básicos

- Otimização pode ser matematicamente escrita como:

minimizar $f_i(x)$, ($i = 1, 2, \dots, M$)

$x \in R^n$

sujeito a $h_j(x) = 0$, ($j = 1, 2, \dots, J$),

$g_k(x) \leq 0$, ($k = 1, 2, \dots, K$)

Onde $f_i(x)$, $h_j(x)$ e $g_k(x)$ são funções do vetor de solução

$$x = (x_1, x_2, \dots, x_n)^T$$

Os componentes de x são chamadas de variáveis de decisão e podem ser contínuas, discretas ou uma mistura das duas.

Conceitos básicos

- As funções $f_i(x)$ são chamadas de função custo ou função objetivo.
- Quando M for 1 então significa que apenas há 1 objetivo.
- O espaço que abrange as variáveis de decisão é chamado de espaço de projeto ou espaço de busca R^n e o espaço formado pelos valores das funções objetivo é chamado de espaço de solução ou espaço de resposta.
- As funções h e g são as restrições. As inequações podem ser também maiores que zero e o problema pode ser maximizado também.
- Uma maneira de classificar os algoritmos de otimização é se eles tem um ou mais objetivos.

Conceitos básicos

- A otimização multiobjetivo é também chamada de otimização multicritério ou otimização de multiatributos.
- Os problemas do mundo real são em sua maioria multiobjetivos.
- Também podemos classificar a otimização de acordo com o número de restrições:
- Se $J+K=0$ então é chamado de problema de otimização sem restrição (unconstrained).
- Se $K=0$ e J maior ou igual a 1 então é um problema restrito a igualdade (equality constraint).
- Se $J=0$ e K maior ou igual a 1 então é um problema restrito a desigualdade (inequality constraint)

Conceitos básicos

- As funções objetivo podem ser lineares ou não-lineares.
- Em geral é tudo não linear.
- Para se trabalhar com problemas envolvendo restrições, existem três principais alternativas:
 - Penalização de soluções inviáveis.
 - Correção de soluções inviáveis.
 - Geração apenas de soluções viáveis.

Conceitos básicos

Outra forma de classificação:

- **Algoritmos evolucionários**

Trabalham com populações de soluções, e, em geral, são bio-inspirados.

Exemplos: Algoritmos Genéticos, Colônia de Formigas, Evolução Diferencial, Enxame de Partículas(PSO), Polinização de Flores (FPA) Algoritmos Meméticos.

- **Algoritmos de busca de vizinhança**

Trabalham com uma única solução e investigam a sua vizinhança por meio de mecanismos de busca local.

Exemplo: Simulated Annealing.

Conceitos básicos

- **Conceito de vizinhança de uma solução**
- **Codificação de um problema de otimização:**
 - Como representar um problema;
 - A cada solução viável, deve ser relacionado um valor de função objetivo.
- **Solução vizinha:** solução acessível a partir de uma dada solução, por meio de um movimento.
- **Perturbação:** alteração em uma dada solução para que se visite outra região do espaço de busca.
- **Busca local:** investigação da vizinhança de uma solução obtida por soluções acessíveis, em geral, por um movimento.

Conceitos básicos

- Busca por Otimalidade
- Depois que um problema é formulado corretamente, a tarefa principal é encontrar soluções ótimas por algum procedimento de solução usando as técnicas matemáticas certas.
- Vamos imaginar uma caça ao tesouro em uma ilha dentro de um tempo limite.
- Em um extremo estamos cegos sem nenhum guia.

Conceitos básicos

- O processo de busca é essencialmente uma busca pura aleatória.
- Como podemos esperar não é eficiente.



Conceitos básicos



- No outro extremo, se nos dissessem que o tesouro está no mais alto pico de uma região conhecida, nós subiríamos até o penhasco mais íngreme e tentaríamos atingir o mais alto pico e esse cenário corresponde a técnica clássica de escalada (hill-climbing).
- Em muitos casos nossa busca é entre esses extremos.
- Nós não estamos de olhos vendados mas não sabemos onde procurar.
- Não é uma boa ideia procurar em cada metro quadrado de uma determinada região para encontrar um tesouro perdido

Conceitos básicos

- O cenário mais provável é que façamos um passo aleatório procurando alguma dica ou sugestão para onde ir.
- Um lugar é procurado quase aleatoriamente e depois verifica-se a próxima posição de maneira plausível e então assim por diante.
- Esse passeio aleatório é a principal característica dos modernos algoritmos de busca.
- Pode-se fazer essa caça ao tesouro sozinho, então todo o caminho é uma pesquisa baseada em trajetória.
- O simulated annealing é um desses algoritmos que é baseado em pesquisa de trajetória.

Conceitos básicos

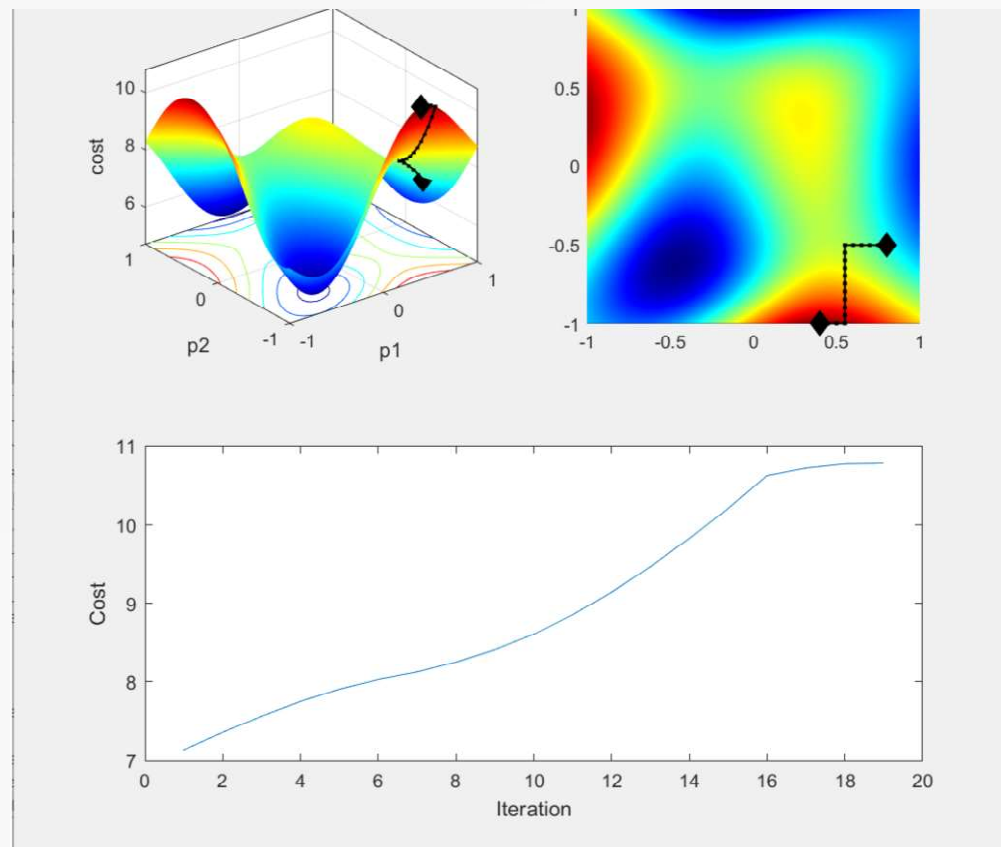
- De outra forma, um grupo de pessoas pode procurar o tesouro e trocar informações entre si, isso corresponde a inteligência de enxame.
- O PSO é um desses algoritmos.
- Se o tesouro é realmente importante e se a área é grande, o processo de busca levará um longo tempo.
- Então senão existe limite de tempo e todas as regiões são acessíveis, então é teoricamente possível encontrar o tesouro (a solução ótima global).
- Uma outra maneira mais eficiente de fazer isso seria manter somente os melhores caçadores, avaliando a competência individual em combinação com a história do sistema. Essa é a base dos modernos algoritmos : AG por exemplo.

Conceitos básicos

- Uma outra maneira de classificar os algoritmos é se são determinísticos ou estocásticos.
- Os determinísticos consideram o caminho e as variáveis de projeto e funções repetíveis.
- Por exemplo, o hill climbing é um algoritmo determinístico e tanto faz rodar ele agora ou mais tarde ou amanhã ele iniciará do mesmo ponto e seguirá o mesmo caminho.
- Os algoritmos estocásticos tem sempre alguma aleatoriedade.
- Nos algoritmos genéticos as variáveis iniciam de maneira diferente a cada rodada, os resultados serão diferentes (como os números na verdade são pseudo aleatórios em algum momento vão se repetir) mas os caminhos não serão muito parecidos.

Conceitos básicos

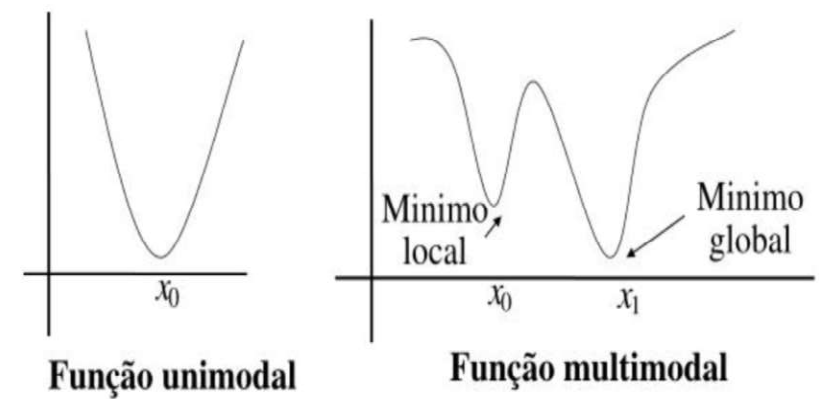
- Pode-se ainda misturar os dois como o hill climbing com início aleatório mas no final sempre será classificado como estocástico por possuir o componente aleatório.



Conceitos básicos

- Nature Inspired Algorithms

- Retornando a classificação dos algoritmos para os determinísticos os que usam informação de gradiente são chamados de gradient based algorithms.
- Ex: Newton-Raphson usa valores de função e suas derivadas e dá resultados satisfatórios para problemas unimodais.
- Exceção: se há alguma descontinuidade na função não trabalha bem.
- Existem os gradient free algorithms: Nelder-Mead (usam valores das funções e não as derivadas)



Conceitos básicos

- Nature Inspired Algorithms
- Voltando ao conceito de heurística e metaheurística sob o ponto de vista de algoritmos estocásticos:
 - Heurísticos: encontram ou descobrem por tentativa e erro.
 - soluções de qualidade para uma otimização difícil podem ser encontradas em um período de tempo razoável, mas não existe garantia de ter encontrado soluções ótimas.
 - É esperado também que funcionem bem a maior parte do tempo. São satisfatórios quando precisa-se de boas soluções e não ótimas.
 - Metaheurísticos: meta significa além ou nível superior, geralmente tem melhores resultados que os heurísticos.

Conceitos básicos

- **Nature Inspired Algorithms**
- Todos os algoritmos metaheurísticos usam certa compensação entre aleatoriedade e busca local.
- Há algumas tendências em chamar algoritmos metaheurísticos aqueles algoritmos estocásticos com aleatoriedade e busca local.
- A aleatoriedade provê um mecanismo de sair de uma busca local para uma busca de escala global.
- Isso leva a conclusão que muitos algoritmos metaheurísticos intencionam ser satisfatórios para a otimização global.

Conceitos básicos

- Nature Inspired Algorithms

Heurísticos são uma maneira por tentativa e erro para produzir soluções aceitáveis para um problema complexo em um tempo prático razoável.

A complexidade do problema de interesse torna ele impossível de procurar cada possível solução ou combinação, o objetivo é encontrar soluções satisfatórias em um prazo aceitável.

Não existe garantia que as melhores soluções serão encontradas. A ideia é ter um algoritmo eficiente mas prático que trabalhará a maior parte do tempo e é capaz de produzir boas soluções.

Entre as soluções de qualidade encontradas é esperado encontrar algumas delas próxima do ótimo (embora não exista garantia disso)

Conceitos básicos

- Nature Inspired Algorithms

Dois componentes dos algoritmos metaheurísticos são:

Intensificação e Diversificação ou “exploitation” e “exploration”.

Diversificação significa gerar soluções diversas como explorar o espaço de busca de forma global.

Intensificação significa focar a busca em uma região local explorando a informação de que uma boa solução atual é encontrada nesta região.

Esse processo trabalhando junto com a seleção das melhores soluções.

Conceitos básicos

- Nature Inspired Algorithms

A seleção das melhores assegura que convergirão para a otimalidade, enquanto a diversificação via aleatoriedade evita que as soluções caiam em um ótimo local, ao mesmo tempo que aumenta a diversidade das soluções.

A boa combinação entre esses dois algoritmos assegurará que a otimalidade global é atingida.

Considerações Finais

- Técnicas capazes de gerar soluções de alta qualidade em tempo computacional admissível.
- Não existe garantia da otimalidade da solução, bem como noção de distância da solução obtida em relação ao ótimo global.
- Existe um *trade-off* diversificação (exploração) e intensificação (exploração).
- Parâmetros devem ser calibrados, os quais afetam substancialmente a qualidade das soluções.
- Classificação usual: algoritmos evolucionários e algoritmos de busca de vizinhança.

Bibliografia

- BLUM, C.; ROLI, A. Metaheuristics in Combinatorial Optimization: Overview and conceptual comparison. **ACM Computing Surveys**, v. 35, p. 268-308, 2003.
 - GIGERENZER, G.W. Heuristic Decision Making. *Annual Review of Psychology*. v. 62 (1), p. 451-482, 2011.
 - GOLDBARG, M.C.; GOLDBARG, E.G.; LUNA, H.P.L. **Otimização combinatória e meta-heurísticas: algoritmos e aplicações**, Rio de Janeiro: Campus, 2016.
 - REEVES, C.R. (Ed.) **Modern heuristic techniques for combinatorial problems**, London: McGraw-Hill, 1995.
 - TALBI, E.G. **Metaheuristics: from design to implementation**. New Jersey: John Wiley & Sons, 2009
- Xin She Yang. *Nature Inspired Metaheuristics Algorithm*, Luniver Press, 2010..