

A02 Introdução à Otimização

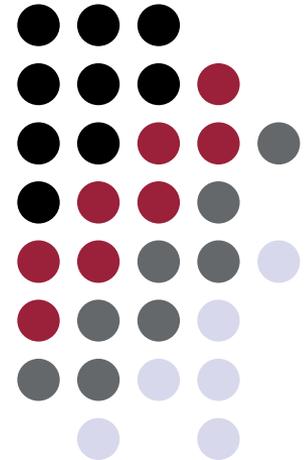


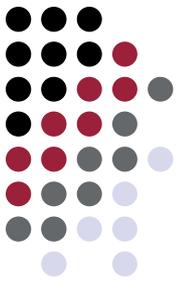
UFOP

Universidade Federal
de Ouro Preto

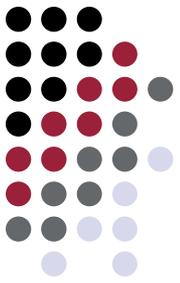
PEP300 Técnicas Metaheurísticas para Otimização Combinatória

Prof. Dr. George H. G. Fonseca
Universidade Federal de Ouro Preto

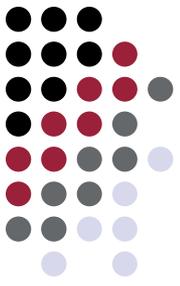




- Nesse curso estaremos interessados na resolução de problemas de Otimização Combinatória
- Métodos de resolução
 - Exatos
 - Heurísticos

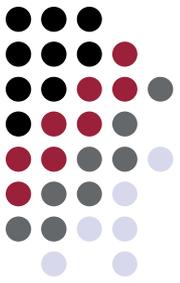


- Dado um conjunto S de soluções viáveis e uma função objetivo $f: S \leftarrow R$, que associa cada solução $s \in S$ a um valor real $f(s)$, a área de otimização consiste em desenvolver metodologias para encontrar a solução $s^* \in S$, dita ótima, para a qual $f(s)$ é **minima** (ou **máxima**)
- Grande parte desses problemas combinatórios são classificados como NP-difíceis
 - Não existe algoritmo que os resolva em tempo polinomial

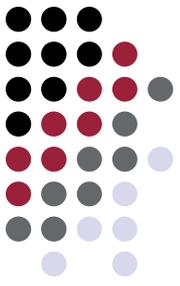


- Área da Pesquisa Operacional que utiliza o método científico para **apoiar** a tomada de decisões, procurando determinar como melhor projetar e operar um sistema, usualmente sob condições que requerem a alocação de recursos **escassos**
- Trabalha com modelos determinísticos
 - As informações relevantes são assumidas como conhecidas (sem incertezas)

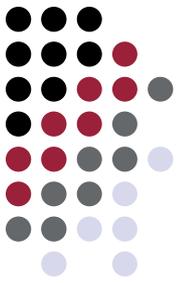
Aplicações Típicas



- Como aplicações típicas de otimização combinatória temos:
 - Roteamento de veículos
 - Programação de horários
 - Sequenciamento da produção
 - Escala de motoristas

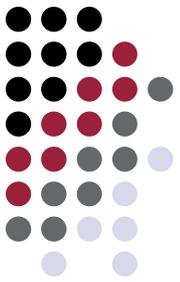


- Problema do Caixeiro Viajante
 - Tem-se um conjunto n de cidades e uma matriz de distâncias entre elas, o objetivo é sair de uma cidade, dita origem, visitar cada uma das $n - 1$ cidades restantes apenas uma única vez e retornar à cidade origem percorrendo a menor distância possível
 - O número total de rotas possíveis é $(n - 1)! / 2$
 - Mesmo para um n pequeno, $n = 20$ tem-se 6×10^{16} rotas possíveis
 - Assumindo que um computador leva 10^{-8} segundos para avaliar uma rota, ele levaria 19 anos para encontrar a melhor rota!



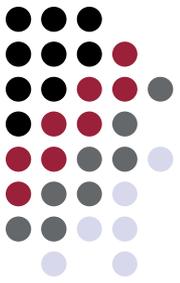
- É possível dar uma certa inteligência aos métodos de enumeração
 - Técnicas branch-and-bound e branch-and-cut reduzem drasticamente o número de soluções a serem avaliadas
 - Pode ser possível resolver problemas de dimensões mais elevadas
 - Mesmo assim, pode ser que muitas soluções tenham que ser analisadas, ou até mesmo todas no pior caso
 - Esse fato limita o uso desses métodos, ditos exatos, dado o tempo proibitivo para se encontrar a solução ótima

Métodos heurísticos

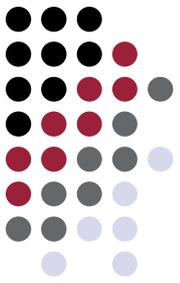


- Na prática, geralmente é o suficiente encontrar uma solução boa para os problemas, invés do ótimo global
- Por esse motivo pesquisadores têm concentrado esforços na utilização de heurísticas para solucionar problemas desse tipo
- Uma heurística é uma técnica inspirada em processos intuitivos que procura uma boa solução a um custo computacional aceitável
 - Não é capaz de garantir a otimalidade, nem garantir o quão próximo está da solução ótima

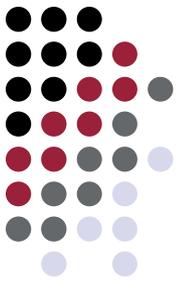
Métodos heurísticos



- Produzir soluções o mais próximas da ótima em tempo reduzido
- Heurísticas muito eficientes foram desenvolvidas para diversos problemas
- A maioria das heurísticas desenvolvidas é muito específica para um problema particular, não sendo eficientes na resolução de uma classe mais ampla de problemas

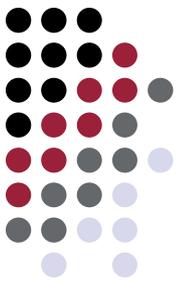


- A partir da década de 80 intensificaram-se os estudos para desenvolver procedimentos heurísticos com caráter mais geral
- Esses procedimentos reúnem conceitos das áreas de Otimização e Inteligência Artificial e são comumente conhecidos como metaheurísticas



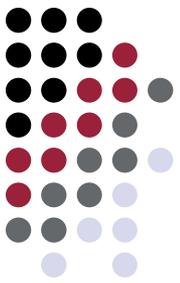
- Entre esses métodos destacam-se:
 - *Simulated Annealing*
 - Busca Tabu
 - Variable Neighborhood Search
 - Iterated Local Search
 - GRASP
 - Algoritmos Genéticos
 - Estratégias Evolutivas

Exemplo 1: Problema da Mochila



- Descrição
 - Imagine que um mochileiro precise viajar e nesta viagem deseja levar alguns objetos
 - Para guardar esses objetos ele pode usar uma mochila, que, no entanto, só pode levar b quilos
 - Cada objeto i tem um peso w_i
 - Cada objeto i proporciona um benefício c_i se for levado
 - O problema consiste em escolher os objetos que trarão o maior benefício possível sem ultrapassar a capacidade da mochila

Exemplo 1: Problema da Mochila

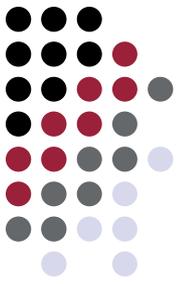


- Instância do Problema da mochila

Objeto	1	2	3	4	5	6	7	8
Benefício	4	3	2	6	2	3	5	4
Peso	5	4	3	9	4	2	6	7

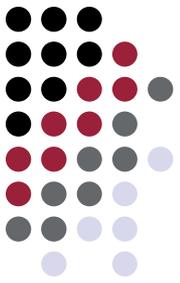
- Capacidade da mochila $b = 20$
- Qual a variável de decisão?

Exemplo 1: Problema da Mochila



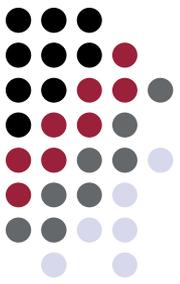
- Modelo para o Problema da mochila
 - $\max \sum_{i=1}^n c_i x_i$
 - Sujeito a:
 - $\sum_{i=1}^n w_i x_i \leq b$
 - $x_i \in \{0, 1\} \quad \forall i = \{1, \dots, n\}$

Exemplo 1: Problema da Mochila



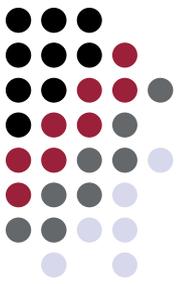
- Como podemos representar computacionalmente o problema da mochila?
- Como verificar se uma solução é viável?
- Como calcular o valor da função objetivo?

Exemplo 2: Problema das p -medianas



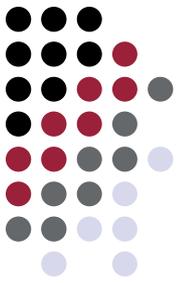
- Descrição
 - Localizar p nós (medianas, facilidades) de forma a minimizar a soma dos custos de ligação entre cada nó de demanda e sua mediana mais próxima
 - Aplicações na localização de
 - Fábricas
 - Depósitos
 - Hospitais
 - Escolas

Exemplo 2: Problema das p -medianas



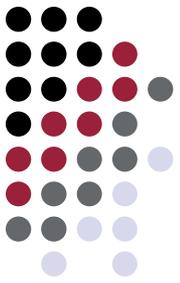
- Restrições:
 - Todas as regiões devem ser atendidas por uma facilidade
 - O número de facilidades instaladas deve ser igual a p
 - Uma região i só pode ser atendida por uma facilidade da região j se uma facilidade for instalada em j
- Parâmetros
 - n : número de localidades
 - p : número de facilidades a serem localizadas
 - c_{ij} : custo de ligação entre a localidade i e a localidade j
 - f_j : custo de instalação da facilidade j

Exemplo 2: Problema das p -medianas



- Qual(is) é(são) a(s) variáveis do problema?
- Como representar o problema computacionalmente?
- Como verificar se uma solução é viável?
- Como calcular o valor da função objetivo?

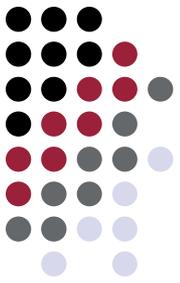
Mais exemplos



- Agendamento de Horários Educacionais
 - Alocar horários e recursos (professores, classes) a aulas respeitando diversas restrições



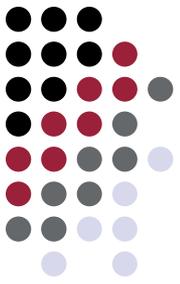
Mais exemplos



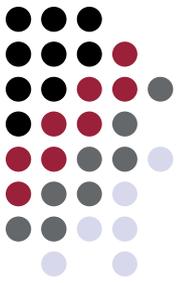
- Agendamento de Horários Educacionais
 - Qual horário você escolheria?

	Seg	Ter	Qua	Qui	Sex
13h30					
15h25		CSI557		CSI557	
18h50		CSI466		CSI466	
20h45		CSI466		CSI466	

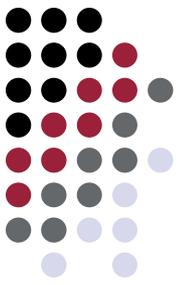
	Seg	Ter	Qua	Qui	Sex
13h30	CSI557				CSI557
15h25				CSI466	
18h50		CSI466			
20h45			CSI466		CSI466



- Agendamento de Horários Educacionais
 - Exemplos de restrições:
 - Não pode haver conflitos de horário
 - Indisponibilidade de professor
 - Espaçamento entre aulas
 - Janela de horário
 - Reduzir o número de dias com aulas, etc



- Agendamento de Horários Educacionais
 - Vamos pensar no ICEA..
 - 368 aulas a serem agendadas
 - 20 horários disponíveis
 - 38 classes e 67 professores
 - Cada aula pode ser alocada a qualquer um dos horários, o que leva a 20^{368} possíveis soluções...
 - Precisamos de bons algoritmos para encontrar soluções de qualidade nesse conjunto!

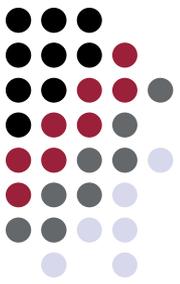


- Planejamento de Redes WLAN

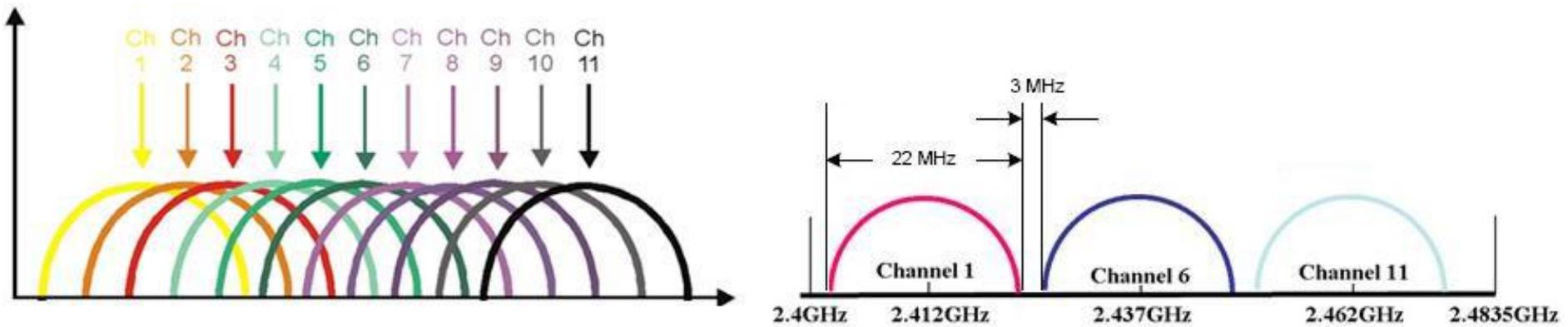
- Definir o posicionamento e a alocação de canais de pontos de acesso tal que a qualidade do sinal seja a melhor para os usuários
 - Cada AP tem um raio de alcance e deve operar em um canal

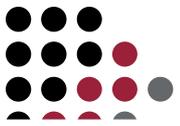


Mais exemplos

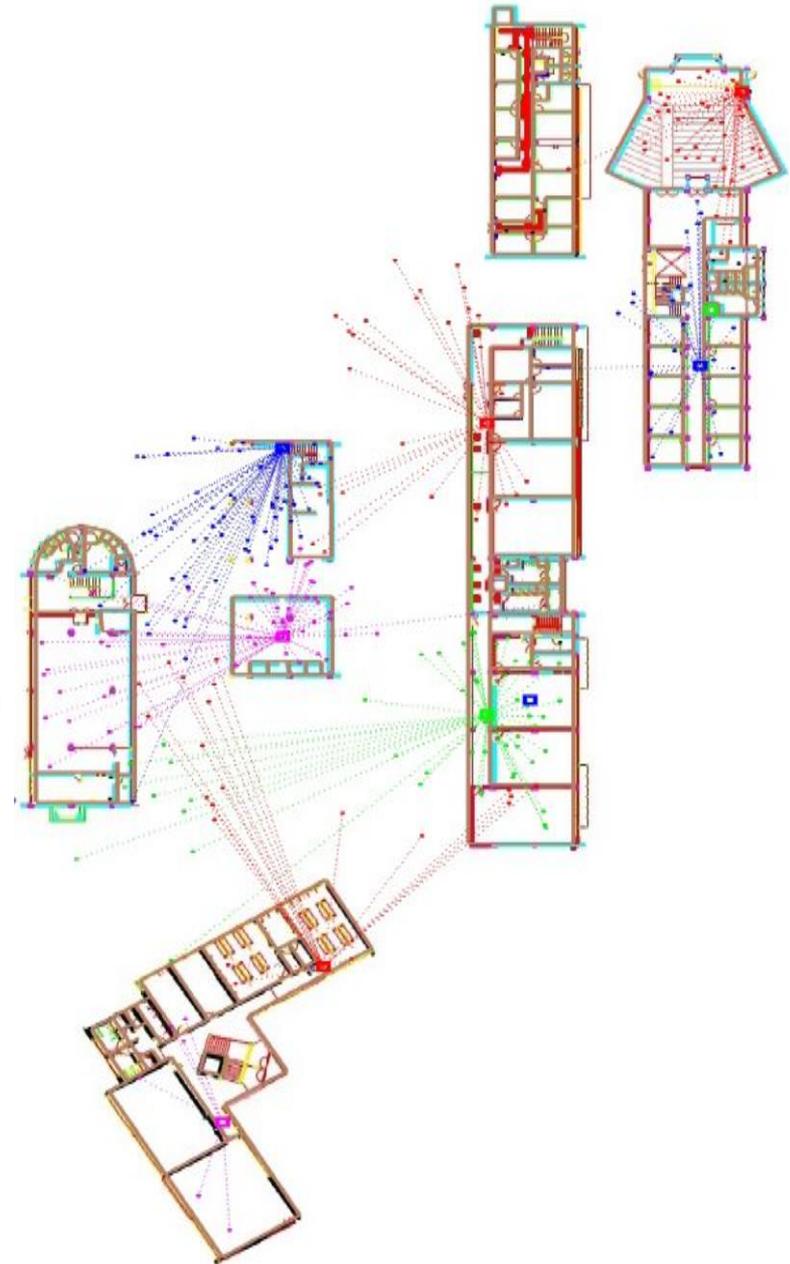


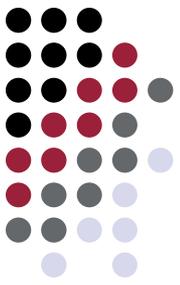
- Planejamento de Redes WLAN
 - Sobre os canais e interferência





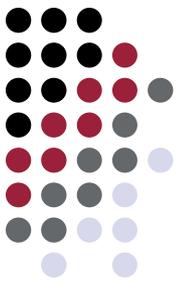
- Planejamento de Redes WLAN
 - MINHAUFOPWIFI
 - Em torno de 600 clientes
 - ~30 pontos de acesso
 - 7 prédios com maior concentração de demanda na cantina e no bloco administrativo





- Programação de Competições Esportivas
 - Para N times, programar $(2 * N) - 2$ rodadas de confrontos de modo que cada equipe enfrente todas as demais duas vezes, uma em casa e outra fora

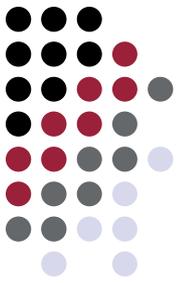




- Programação de Competições Esportivas
 - Objetivos variam de torneio para torneio, ex:
 - **Reduzir os custos e desgaste com viagens**
 - Equilibrar o campeonato
 - Maximizar o número de jogos televisoinados

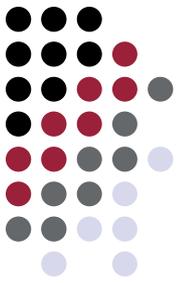
Time/Rodada	1	2	3	4	5	6	7	8	9	10
1	+ 6	- 2	+ 4	+ 3	- 5	- 4	- 3	+ 5	+ 2	- 6
2	+ 5	+ 1	- 3	- 6	+ 4	- 5	+ 6	- 4	- 1	- 5
3	- 4	+ 5	+ 2	- 1	+ 6	- 2	+ 1	- 6	- 5	+ 4
4	+ 3	+ 6	- 1	- 5	- 2	+ 1	+ 5	+ 2	- 6	- 3
5	- 2	- 3	+ 6	+ 4	+ 1	- 6	- 4	- 1	+ 3	+ 2
6	- 1	- 4	- 5	+ 2	- 3	+ 5	- 2	+ 3	+ 4	+ 1

Mais exemplos

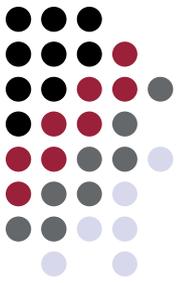


- Programação de Competições Esportivas
 - Qual sequência de jogos gera menos viagens para o Cruzeiro?

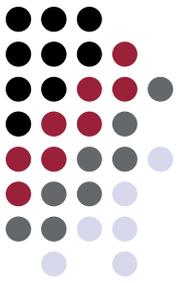




- Programação matemática
 - Fundamentação: na matemática
 - Vantagem: garante e solução ótima
 - Desvantagens
 - Pode gastar um tempo proibitivo para gerar a solução ótima
 - Nem sempre consegue produzir uma (boa) solução viável rapidamente
- Heurísticas
 - Fundamentação: na inteligência artificial
 - Vantagens:
 - De fácil implementação
 - Produzem boas soluções rapidamente
 - Desvantagem
 - Não garantem a otimalidade na solução obtida

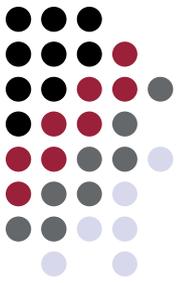


- Utilização ideal
 - O tempo gasto para resolver o problema de forma exata é proibitivo
 - A quantidade de memória necessária para se resolver o problema é maior que a quantidade de memória disponível
 - A quantidade de dados a serem processados é muito grande
 - Não se sabe como resolver o problema de forma exata

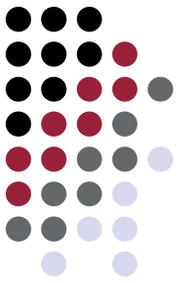


- **Classificação**
 - Métodos construtivos
Constroem uma solução passo a passo, elemento por elemento
 - Métodos de refinamento
Objetivam a melhorar a solução, através de modificações em seus elementos

Passo a passo para solução de problemas



- Entender o problema
- Saber como gerar uma solução inicial
- Saber como avaliar um solução gerada
- Saber como explorar o espaço de busca



- Souza, M. J. F. **Inteligência Computacional para Otimização**. Disponível em www.iceb.ufop.br/decom/prof/marcone/ico2009, acessado em Agosto de 2019.
- Martins, A. X. **Heurísticas e Metaheurísticas**. Material didático, 2018.