# Otimização Combinatória: Aplicações e desafio

Luidi Gelabert Simonetti
luidi@cos.ufrj.br
http://cos.ufrj.br/~luidi/

PESC - COPPE - UFRJ
Coordenador ECI - Escola Politécnica UFRJ

Seminário CEFET - RJ - 2019

# Otimização

?

# Otimização

- É o ato de obter o melhor resultado sobre uma dada circunstancia.
- Pode ser definida como o processo de achar as condições que dão o máximo e mínimo de uma função
- Os métodos de busca pelo ótimo são conhecidos como técnicas de programação matemática e normalmente estudados como parte da pesquisa operacional.

# Otimização - Pesquisa Operacional

- É um ramo interdisciplinar da matemática aplicada que faz uso de modelos matemáticos, estatísticos e de algoritmos na ajuda à tomada de decisões

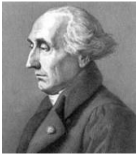
- É usada sobretudo para analisar sistemas complexos do mundo real, tipicamente com o objetivo de melhorar ou otimizar a performance

# Origem - Pesquisa Operacional

- A investigação operacional nasceu durante a II guerra mundial, quando os Aliados se viram confrontados com problemas (logística e militar) de grande dimensão e complexidade.
- Aplicaram o método científico aos problemas que lhes foram sendo colocados e criaram modelos matemáticos que lhes permitissem avaliar o resultado hipotético de estratégias ou decisões alternativas.
- Com o fim do conflito e sucesso obtido, os grupos de cientistas transferiram a nova metodologia na abordagem de problemas para as empresas.
- Com a evolução observada na informática criaram-se condições de concretização algorítmica e velocidade de processamento adaptados à imaginação dos profissionais da pesquisa operacional.

# Desenvolvimento Histórico - Otimização


**Isaac Newton** (1643 - 1727)
Cálculo Diferencial, ...



**Joseph-Louis Lagrange** (1736 - 1813)
Cálculo variacional, minimização funcional,
método de otimização para problemas restritos, ...

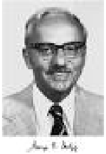
**Augustin-Louis Cauchy** (1789 - 1857)
Solução por substituição direta,
método do gradiente (ou método do máximo declive), ...

# Desenvolvimento Histórico - Otimização

**Leonhard Euler** (1707 - 1783)
Cálculo variacional, minimização funcional, ...

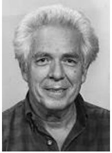
**Gottfried Leibnitz** (1646 - 1716)
Cálculo Diferencial, ...

**George Bernard Dantzig** (1914 - 2005)
Programação Linear, método Simplex (1947), ...



# Desenvolvimento Histórico - Otimização

**Harold William Kuhn** (1925 - 2014)
Condições necessárias e suficientes da solução ótima de problemas de programação matematica, teoria de jogos, ...



**Albert William Tucker** (1905 - 1995)
Condições necessárias e suficientes da solução ótima de problemas de programação matematica, Programação não linear, teoria de jogos (Orientou John Nash [PhD]), ...

# Desenvolvimento Histórico - Otimização

**Richard Bellman** (1920 - 1984)
Principio de otimalidade em programação dinâmica,
...

**John Von Neumann** (1903 - 1957)
Teoria dos jogos,
...

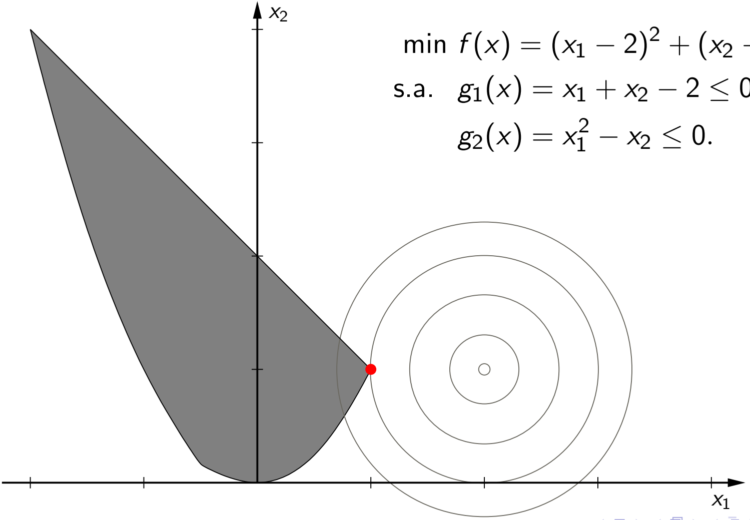
# Exemplo de Programação Não Linear Restrita

- O problema geral de otimização consiste em

$$\begin{aligned} \min\ & f(x) && \leftarrow \text{função objetivo} \\ \text{s.a.}\ & H(x) = 0, && \leftarrow \text{restrições} \\ & G(x) \leq 0. && \end{aligned}$$

- onde $f : \mathbb{R}^n \to \mathbb{R}$, $h_i : \mathbb{R}^n \to \mathbb{R}$, $i \in \mathcal{I}_H$, e $g_i : \mathbb{R}^n \to \mathbb{R}$, $i \in \mathcal{I}_G$.
- O conjunto viável $\Omega = \{x \in \mathbb{R}^n \mid H(x) = 0,\ G(x) \leq 0\}$.
- Normalmente temos um conjunto infinito de soluções viáveis.

# Exemplo Programação Não Linear Restrita




$$\begin{aligned} \min\ & f(x) = (x_1 - 2)^2 + (x_2 - 1)^2 \\ \text{s.a.}\ & g_1(x) = x_1 + x_2 - 2 \leq 0, \\ & g_2(x) = x_1^2 - x_2 \leq 0. \end{aligned}$$

# Definições

## Combinatória

- é um ramo da matemática que estuda coleções finitas de elementos que satisfazem critérios específicos determinados e se preocupa, em particular, com a "contagem" de elementos nessas coleções.

## Otimização Combinatória

- é um ramo da **ciência da computação**, **pesquisa operacional** e da **matemática aplicada** que estuda problemas de otimização em conjuntos finitos.

# Otimização Combinatória

- Em muitos problemas, a busca exaustiva não é tratável. (Mesmo tendo um conjunto de soluções finito)
- A otimização combinatória é um subconjunto da otimização matemática relacionada à pesquisa operacional, à algoritmos e à teoria da complexidade computacional.
- Possui aplicações importantes em vários campos, incluindo inteligência artificial, aprendizado de máquina, teoria de leilões e engenharia de software.

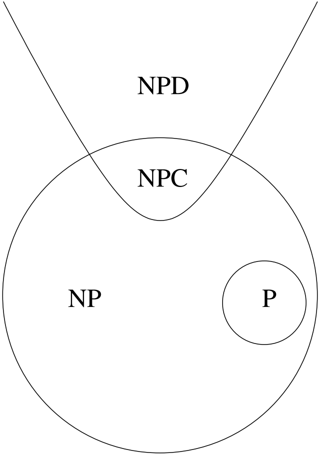
# História Combinatória

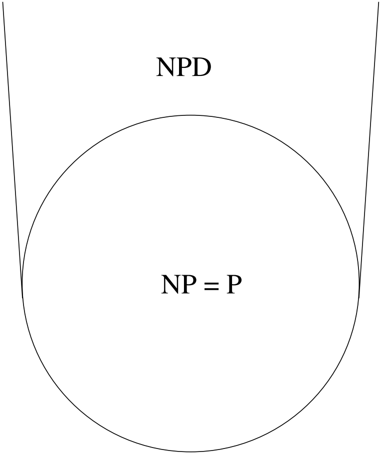
- Conceitos combinatórios básicos e resultados enumerativos apareceram em todo o mundo antigo.
- No século VI AC, o antigo médico indiano Sushruta afirma que 63 combinações podem ser feitas de 6 gostos diferentes, calculando todas as $2^6 - 1$ possibilidades.
- O historiador grego Plutarco discute uma discussão entre Crisipo (século III AC) e Hiparco (século 2 AC) de um problema enumerativo bastante delicado, que mais tarde se mostrou relacionado aos números de Schröder – Hipparchus.
- Arquimedes (século 3 AC) considera um quebra-cabeça de azulejos.
- O matemático indiano Mahāvīra (850) forneceu fórmulas para o número de permutações e combinações, e essas fórmulas podem ter sido familiares aos matemáticos indianos no início do século VI.
- ...

# História Combinatória

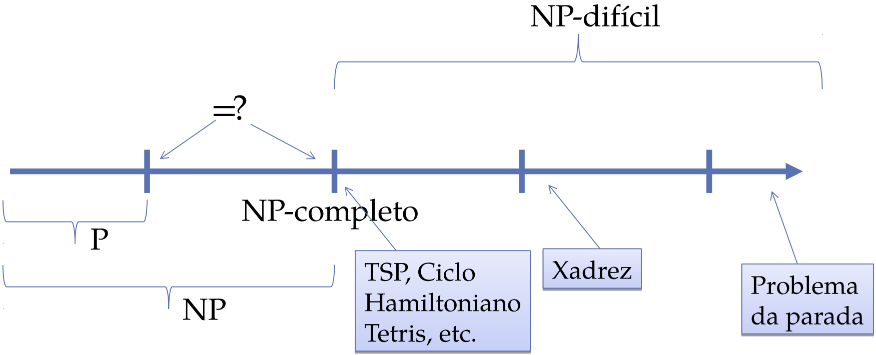
- Durante o Renascimento, junto com o resto da matemática e das ciências, a combinatória teve um renascimento. Os trabalhos de Pascal, Newton, Jacob Bernoulli e Euler tornaram-se fundamentais no campo emergente.
- A teoria dos grafos também teve uma explosão de interesse ao mesmo tempo, especialmente em relação ao problema das quatro cores.
- O crescimento rápido eliminaram as fronteiras entre combinatória e partes da matemática e da ciência da computação, mas ao mesmo tempo levou a uma fragmentação parcial do campo.

# Complexidade



# Dificuldade Computacional



# Otimização Combinatória - Força Bruta

- Tentando resolver por enumeração
  - Normalmente o número de soluções viáveis são $n!$ ou $2^n$.

| $n$ | $\log n$ | $n^2$ | $2^n$ | $n!$ |
|---|---|---|---|---|
| 10 | 3.32 | $10^2$ | $1.02x10^3$ | $3.6x10^6$ |
| 100 | 6.64 | $10^4$ | $1.27x10^{30}$ | $9.33x10^{157}$ |
| 1000 | 9.97 | $10^6$ | $1.07x10^{301}$ | $4.02x10^{2567}$ |

  - Num supercomputador atual ($10^{16}$ flops) seria necessário $2.95x10^{134}$ anos (100!) e $1.27x10^{2544}$ anos (1000!).

    Obs: o universo tem $13.82x10^9$ anos e $6x10^{79}$ átomos de hidrogênio.
  - Usando enumeração completa só revolvemos problemas para $n$ muito pequeno!!
  - **Obs: Número de soluções não é absoluto!**
    **Ex. Árvore geradora x Caminho hamiltoniano**

# Problema da Mochila 0-1

- Versão 1:
  - Um ladrão tem $n$ possíveis itens para roubar
  - Cada item $i$ tem um valor $c_i$ e peso $a_i$
  - O ladrão tem um limite máximo de peso que consegue levar $b$
  - Objetivo: selecionar os itens com maior valor
- Versão 2:
  - Uma empresa tem $n$ possíveis projetos para investir
  - Cada projeto tem um custo $a_i$ e um retorno $c_i$
  - O orçamento da empresa para investir é limitado $b$
  - Objetivo: selecionar um portfólio de projeto de máximo retorno
- Um problema pode ser "adaptado" para outros problemas "reais"

# Problema da Mochila 0-1

- Uma empresa tem $n$ possíveis projetos para investir
- Cada projeto tem um custo $a_i$ e um retorno $c_i$
- O orçamento da empresa para investir é limitado $b$
- Objetivo: selecionar um portfólio de projeto de máximo retorno
- Formulação Matemática

$$\max \left\{ \sum_{i=1}^{n} c_i x_i : x \in P \cap \mathbb{B}^n \right\}$$

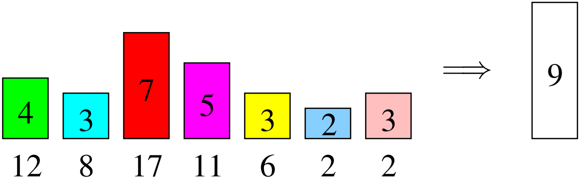
Onde $P$ é dado por:

$$\sum_{i=1}^{n} a_i x_i \leq b$$

$$0 \leq x_i \leq 1, \ \forall i \in \{1, \ldots, n\}$$

# Problema da Mochila 0-1

- Uma empresa tem $n$ possíveis projetos para investir
- Cada projeto tem um custo $a_i$ e um retorno $c_i$
- O orçamento da empresa para investir é limitado $b$
- Objetivo: selecionar um portfólio de projeto de máximo retorno
- Exemplo de Heurística Gulosa

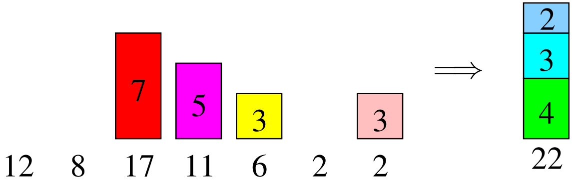



1. ordenar os $n$ elementos em ordem decrescente da razão $\frac{c_j}{a_j}$
2. adicionar os elementos nessa ordem até o limite ser estourado

- Solução heurística - custo 22
- Solução ótima - custo 23

# Problema da Mochila 0-1

- Uma empresa tem $n$ possíveis projetos para investir
- Cada projeto tem um custo $a_i$ e um retorno $c_i$
- O orçamento da empresa para investir é limitado $b$
- Objetivo: selecionar um portfólio de projeto de máximo retorno
- Exemplo de Heurística Gulosa

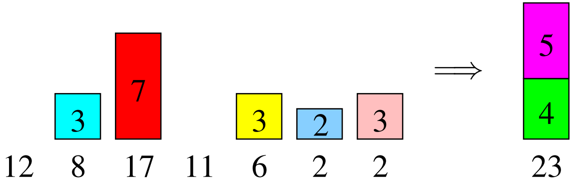



  1. ordenar os $n$ elementos em ordem decrescente da razão $\frac{c_j}{a_j}$
  2. adicionar os elementos nessa ordem até o limite ser estourado

- Solução heurística - custo 22
- Solução ótima - custo 23

# Problema da Mochila 0-1

- Uma empresa tem $n$ possíveis projetos para investir
- Cada projeto tem um custo $a_i$ e um retorno $c_i$
- O orçamento da empresa para investir é limitado $b$
- Objetivo: selecionar um portfólio de projeto de máximo retorno
- Exemplo de Heurística Gulosa




  1. ordenar os $n$ elementos em ordem decrescente da razão $\frac{c_j}{a_j}$
  2. adicionar os elementos nessa ordem até o limite ser estourado

- Solução heurística - custo 22
- Solução ótima - custo 23

# Problema com múltiplas Mochilas

- $n$ itens e $m$ mochilas
- Cada item $i$ um retorno $c_i$
- Cada item $i$ tem um custo $c_{ij}$ para ser adicionado a mochila $j$
- Cada mochila $j$ tem uma capacidade $b_j$
- Objetivo: selecionar os itens que maximizem o retorno
- Essas múltiplas mochilas podem ser necessárias para o problema ou podem garantir um grau de diversidade.

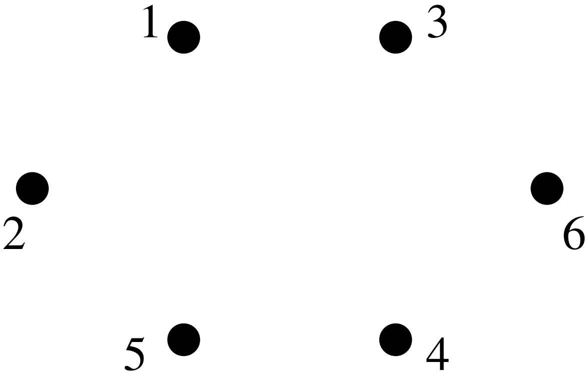
# Problema da Mochila quadrática

- Um estudante tem $n$ possíveis itens para escolher
- Cada item tem um peso $a_i$ e um retorno $c_i$
- A capacidade da mochila é $b$
- Além do retorno de cada item temos o retorno de escolher dois itens conjugados
- Objetivo: selecionar os itens que levem ao máximo retorno

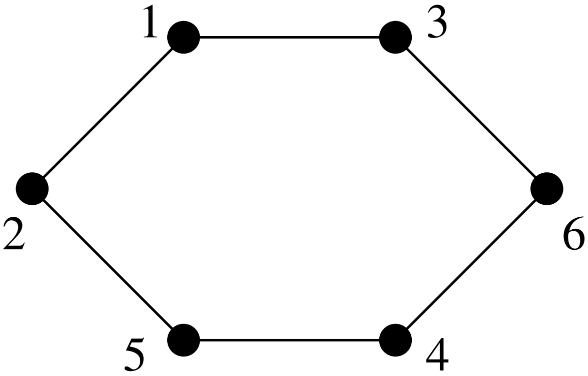
# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

- É o mais famoso problema em Otimização Inteira e Combinatória!
- Um conjunto de cidades $N = \{1, \ldots, n\}$ é dado.
- O tempo de viagem entre quaisquer duas cidades é conhecido.
- Um caixeiro viajante deve visitar cada cidade uma única vez e retornar ao ponto de partida.
- Objetivo: Definir uma ordem de visita de forma a retornar ao ponto de partida o mais rápido possível.
- uma solução viável do TSP é chamado de um tour e, no grafo, corresponde a um ciclo hamiltoniano.

# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

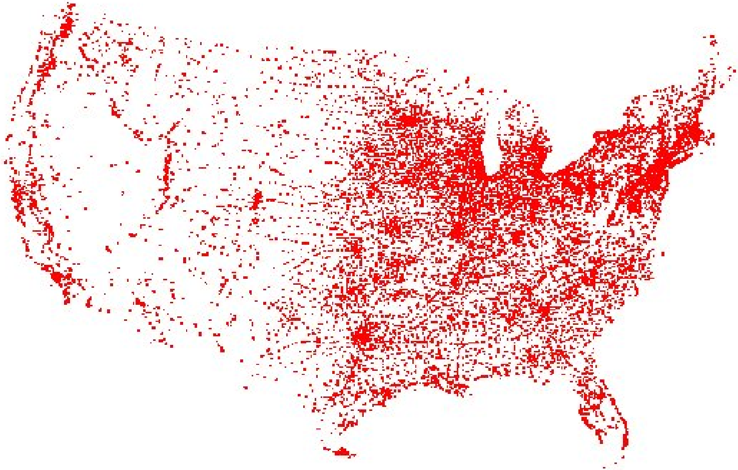


# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)



# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

USA - 13509 cidades

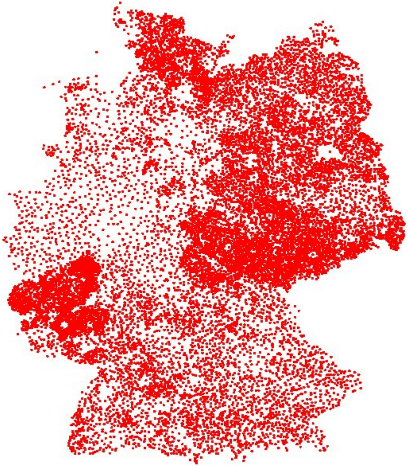


# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

USA - 13509 cidades

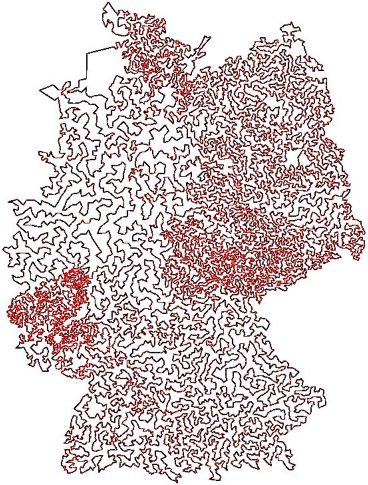
# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

## Alemanha - 15112 cidades



# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

## Alemanha - 15112 cidades

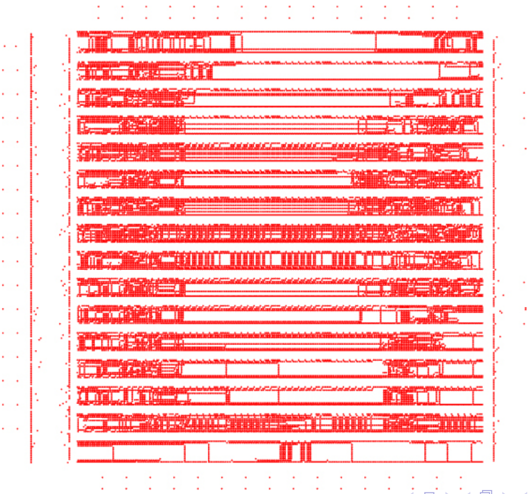


# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

- Por que estudar um problema que não tem mais tanta aplicação prática?
- Um método desenvolvido para um problema "normalmente" pode ser adaptado para outros
- Na verdade o TSP tem diversas aplicações atuais
  - A principal aplicação vem de subproblemas em várias aplicações de transporte e logística (Ônibus escolar, equipamento agrícola para teste de solo, visita agendada em firma de cabo, delivery de refeição para pessoas "restritas" em casa, planejamento de retroescavadeira em armazéns, etc)
  - planejamento de uma máquina de furar uma placa de circuito
  - Otimizar a sequencia de imagens de objetos celestes
  - Coleta de moedas em telefones públicos
  - ...

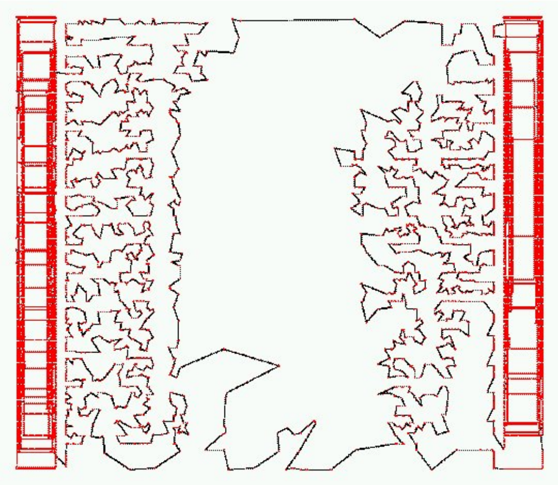
# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

## Placa de circuito - 7397 pontos



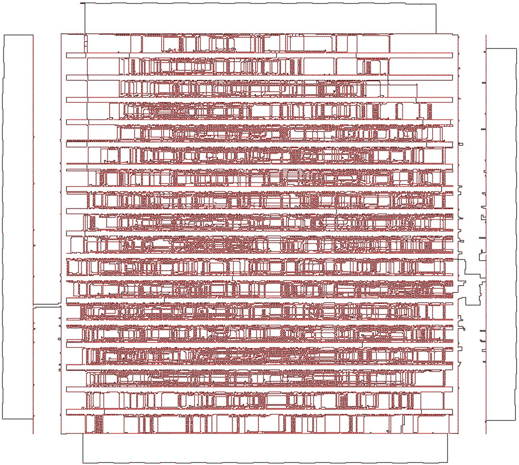
# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

## Placa de circuito - 7397 pontos

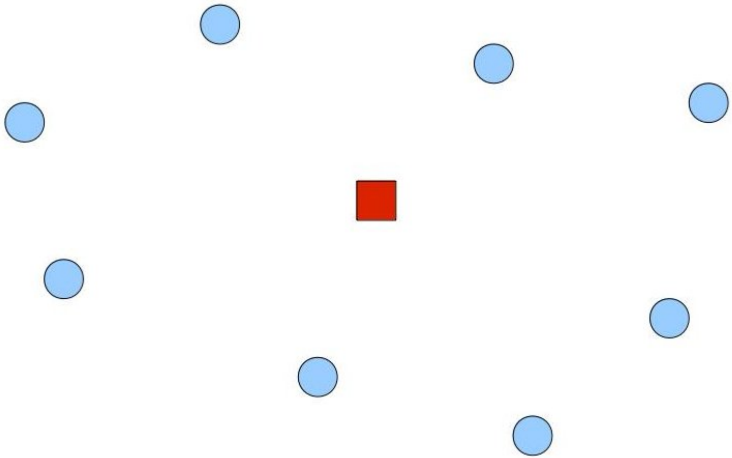


# Problema do Caixeiro Viajante (Traveling Salesman Problem, TSP)

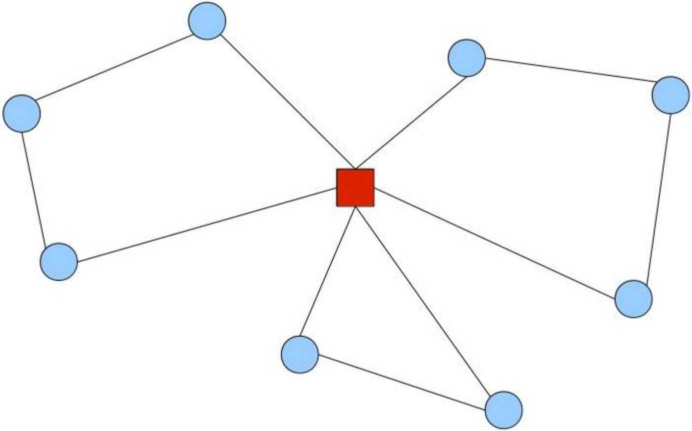
## Placa de circuito - 85900 pontos



# Exemplo: Roteamento de veículos

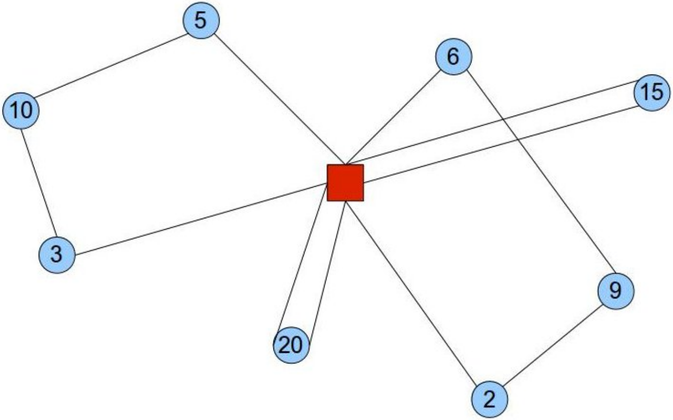


# Exemplo: Roteamento de veículos



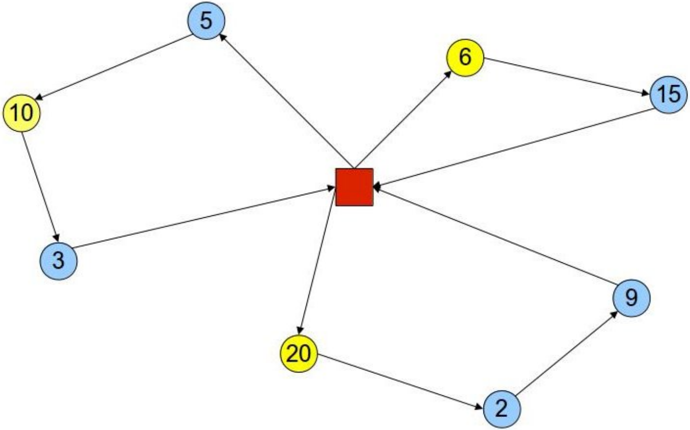
# Exemplo: Roteamento de veículos

## Variações - Capacitado - Ex: 20



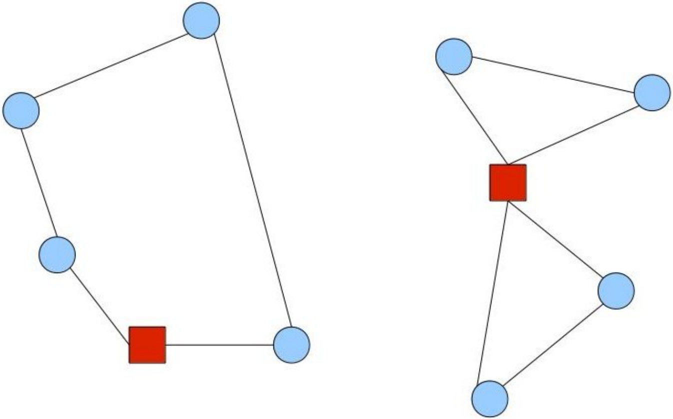
# Exemplo: Roteamento de veículos

## Variações - Coleta e Entrega - Ex: 20



# Exemplo: Roteamento de veículos

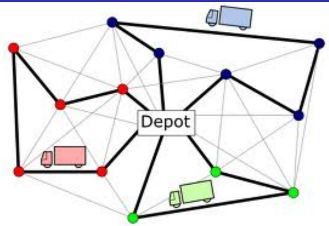
## Variações - Múltiplos Depósitos

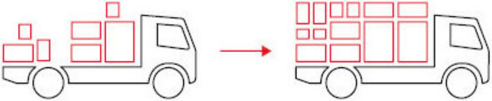


# Roteamento de veículos

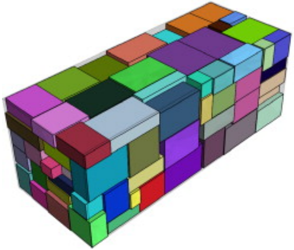
## Realidade

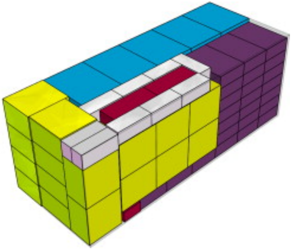
- Janela de tempo
- Ordem dos itens no caminhão
- Empacotamento (2d ou 3d)
- Trafego
- ...

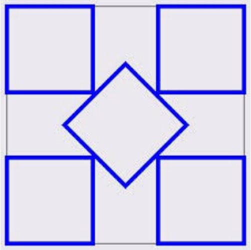


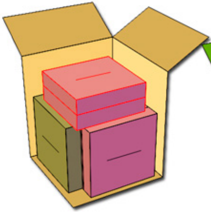



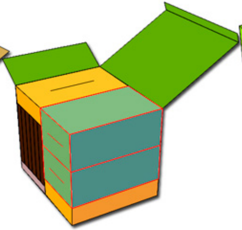
# Corte e Empacotamento

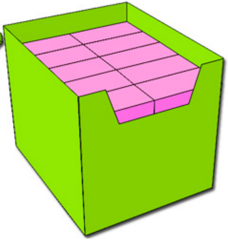




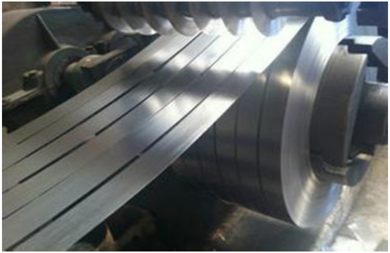


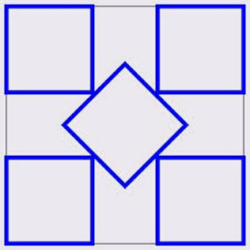


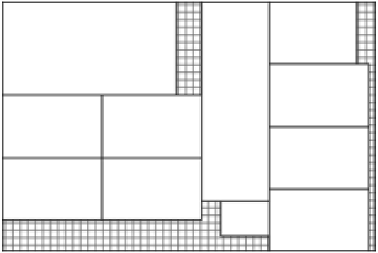




# Corte e Empacotamento







**Dado**

Um grafo $G = (V, E)$

**Achar**

A árvore geradora $T$ do grafo com maior número de folhas

**Folha**

Vértice adjacente a um único vértice

# Árvore Geradora com número máximo de folhas

## Definição

**Dado**

Um grafo $G = (V, E)$

**Achar**

A árvore geradora $T$ do grafo com maior número de folhas

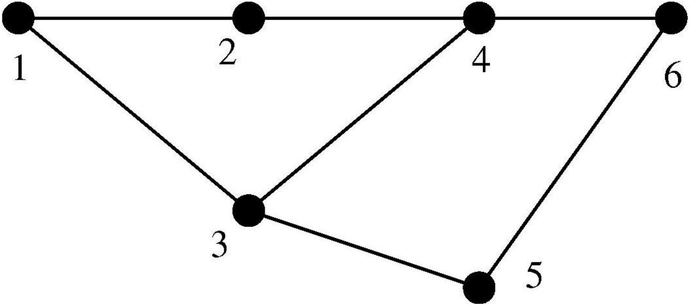
**Folha**

Vértice adjacente a um único vértice

# Árvore Geradora com número máximo de folhas

## Exemplo

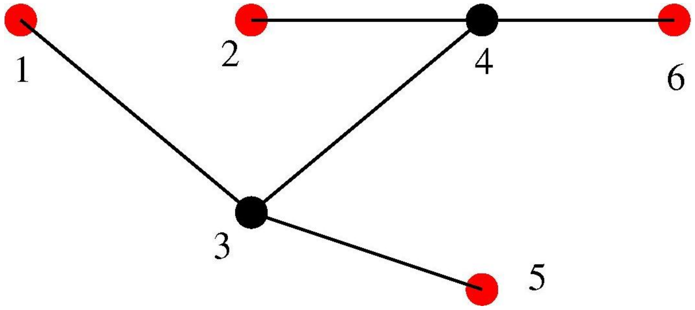
$G(V, E)$



# Árvore Geradora com número máximo de folhas

## Exemplo

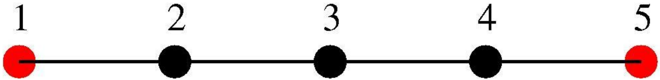
### Solução (4 folhas)



# Árvore Geradora com número máximo de folhas

## Exemplo

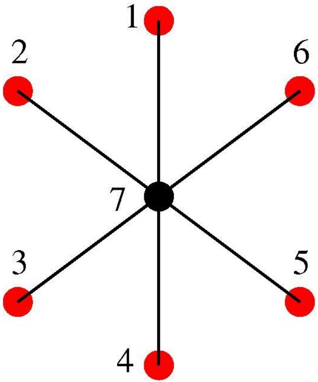
### Pior caso



# Árvore Geradora com número máximo de folhas

## Exemplo

### Melhor caso

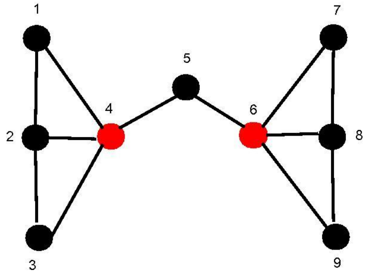


## Conjunto dominante Mínimo de $G$: $C \subset V$ tal que

- para todo $i \in V \setminus C$: existe uma aresta com uma extremidade em $i$ e outra em $C$.
- encontrar o conjunto dominante de $G$ com o mínimo número de vértices possível.

## Mínimo conjunto dominante conexo

- Implica numa árvore de $G$ sem os vértices folhas $\mathcal{T}$.
- Fácil gerar a árvore geradora com maior número de folhas de G.



# Árvore Geradora com número máximo de folhas

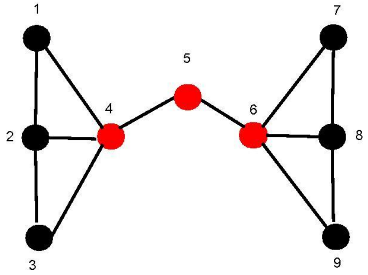
## Equivalente ao problema

### Conjunto dominante Mínimo de $G$: $C \subset V$ tal que

- para todo $i \in V \setminus C$: existe uma aresta com uma extremidade em $i$ e outra em $C$.
- encontrar o conjunto dominante de $G$ com o mínimo número de vértices possível.

### Mínimo conjunto dominante conexo

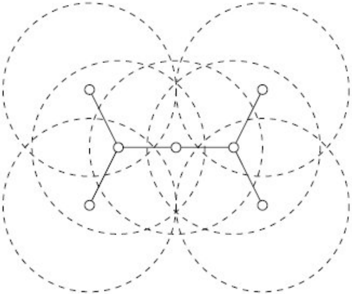
- Implica numa árvore de $G$ sem os vértices folhas $\mathcal{T}$.
- Fácil gerar a árvore geradora com maior número de folhas de G.



# Mínimo conjunto dominante conexo

## Motivação

- Modela rede de telecomunicação
  Ex: Wireless Ad Hoc Networks
- Layoult de circuitos

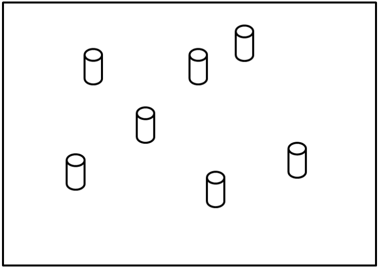


# Problema de Alocação de Frequências de celulares

- Problema de alocar frequências a antenas de transmissão
  - Antenas de transmissão
  - Zona de cobertura (antenas possuem interseções entre as zonas)
  - Antenas com interseção nas zonas de cobertura devem receber frequências respeitando as distâncias para não gerar interferências)
  - Objetivo: 1) Minimizar a faixa de frequências utilizadas 2) Minimizar a interferência

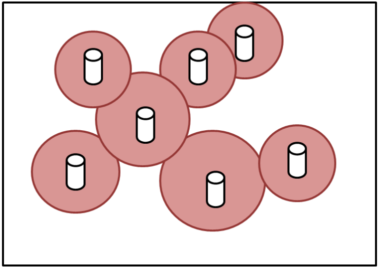
# Problema de Alocação de Frequências de celulares

- Problema de alocar frequências a antenas de transmissão
  - Antenas de transmissão
  - Zona de cobertura (antenas possuem interseções entre as zonas)
  - Antenas com interseção nas zonas de cobertura devem receber frequências respeitando as distâncias para não gerar interferências)
  - Objetivo: 1) Minimizar a faixa de frequências utilizadas 2) Minimizar a interferência



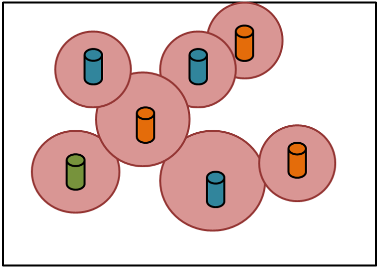
# Problema de Alocação de Frequências de celulares

- Problema de alocar frequências a antenas de transmissão
  - Antenas de transmissão
  - Zona de cobertura (antenas possuem interseções entre as zonas)
  - Antenas com interseção nas zonas de cobertura devem receber frequências respeitando as distâncias para não gerar interferências)
  - Objetivo: 1) Minimizar a faixa de frequências utilizadas 2) Minimizar a interferência



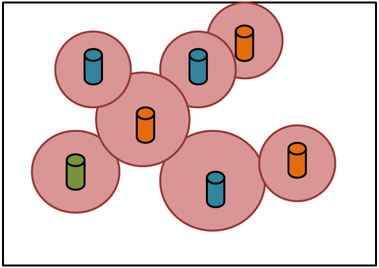
# Problema de Alocação de Frequências de celulares

- Problema de alocar frequências a antenas de transmissão
  - Antenas de transmissão
  - Zona de cobertura (antenas possuem interseções entre as zonas)
  - Antenas com interseção nas zonas de cobertura devem receber frequências respeitando as distâncias para não gerar interferências)
  - Objetivo: 1) Minimizar a faixa de frequências utilizadas 2) Minimizar a interferência



# Problema de Alocação de Frequências de celulares

- Problema de alocar frequências a antenas de transmissão
  - Antenas de transmissão
  - Zona de cobertura (antenas possuem interseções entre as zonas)
  - Antenas com interseção nas zonas de cobertura devem receber frequências respeitando as distâncias para não gerar interferências)
  - Objetivo: 1) Minimizar a faixa de frequências utilizadas 2) Minimizar a interferência

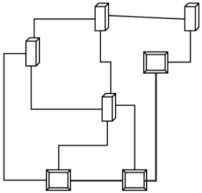


# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência

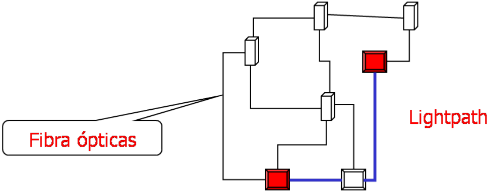
# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência



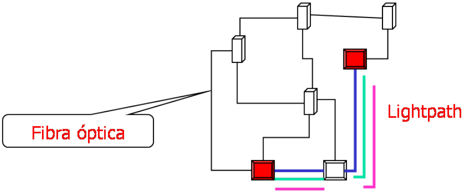
# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência



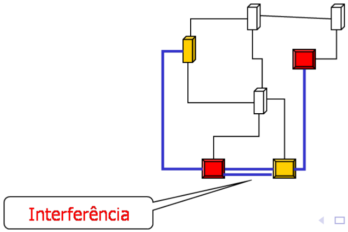
# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência



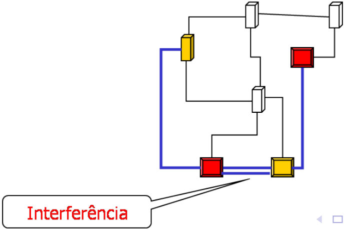
# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência

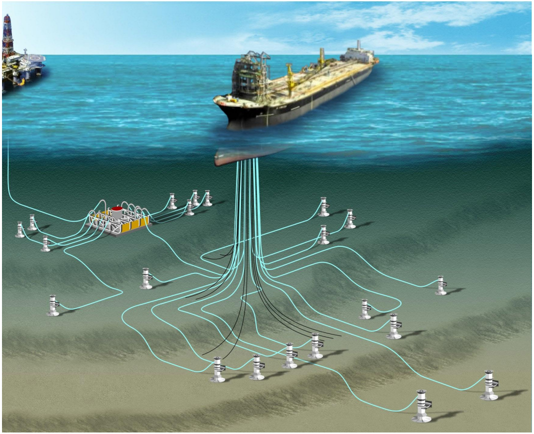


# RWA (Routing and Wavelength Assignment)

- Em redes de computadores de alto desempenho, dados são transmitidos entre terminais
- para haver comunicação entre 2 terminais temos 1) criação de um caminho 2) estabelecer qual frequencia os dados serão transmitidos
- Uma fibra óptica pode transmitir dados em várias frequencias diferentes
- Interferência
- Objetivo : 1) Roteamento das comunicações a serem realizadas 2) Atribuição de frequências sem interferência



# Arranjo Submarino



# Aplicações

- Em todas as áreas de industria e serviços
- Os problemas clássicos ainda são usados como parte de soluções de problemas mais complexos ou com uma simplificação
- Novos problemas surgem com novas tecnologias, industria ou serviços
  - IoT
  - IA
  - Industria 4.0 (5.0)
  - Aplicativos (serviços)
  - Analytics

# Obrigado