Algoritmos de Busca Local

Slides adaptados de material didático dos professores Adriana Alvim (UniRio), Celso Ribeiro (UFF) e Diego Luchi (egresso Doutorado UFES)

labtim
Laboratório de Otimização - UFES

UNIVERSIDADE FEDERAL DO ESPIRITO SANTO
DOCETE OMNES GENTES

Busca Local

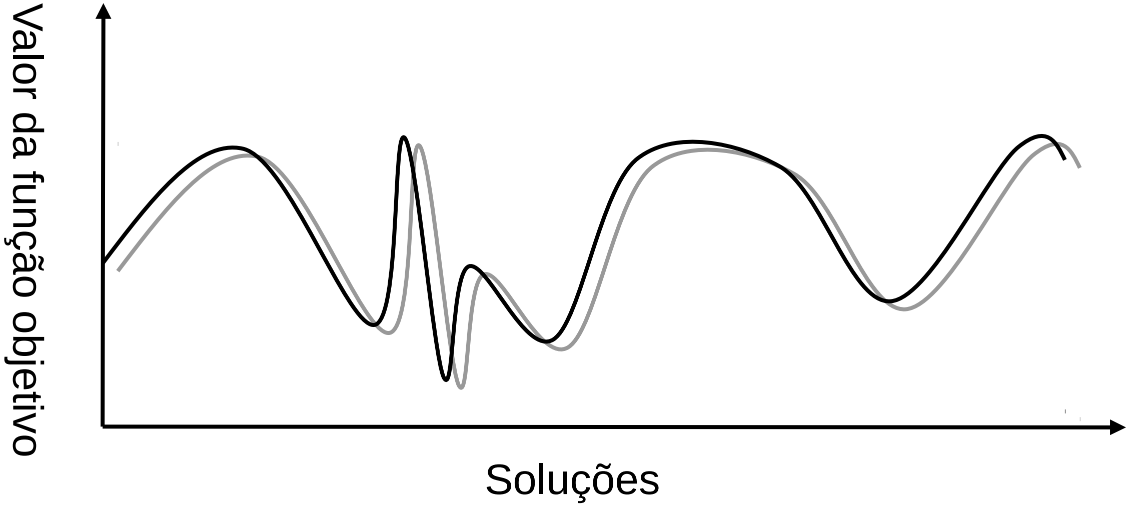
- Técnica de exploração do espaço de soluções
- Componentes da técnica
  - espaço de busca
    - conjunto finito de soluções candidatas
  - conjunto de soluções
    - soluções viáveis
  - relação de vizinhança
  - solução inicial
  - movimento (passo da busca)
  - critério de parada
- Busca local estocástica
  - alguns desses componentes podem ser obtidos de forma aleatória

Busca local

- Busca local
  - inicia em algum ponto do espaço de busca
  - move iterativamente de um posição para outra na vizinhança
    - cada posição possui um número relativamente pequeno de vizinhos
    - cada movimento é determinado por uma decisão baseada simplesmente em conhecimento local
  - tipicamente são algoritmos incompletos
    - não garante que eventualmente encontrará soluções ótimas
    - não consegue determinar com certeza que não existe solução ótima
    - pode visitar a mesma solução mais de uma vez

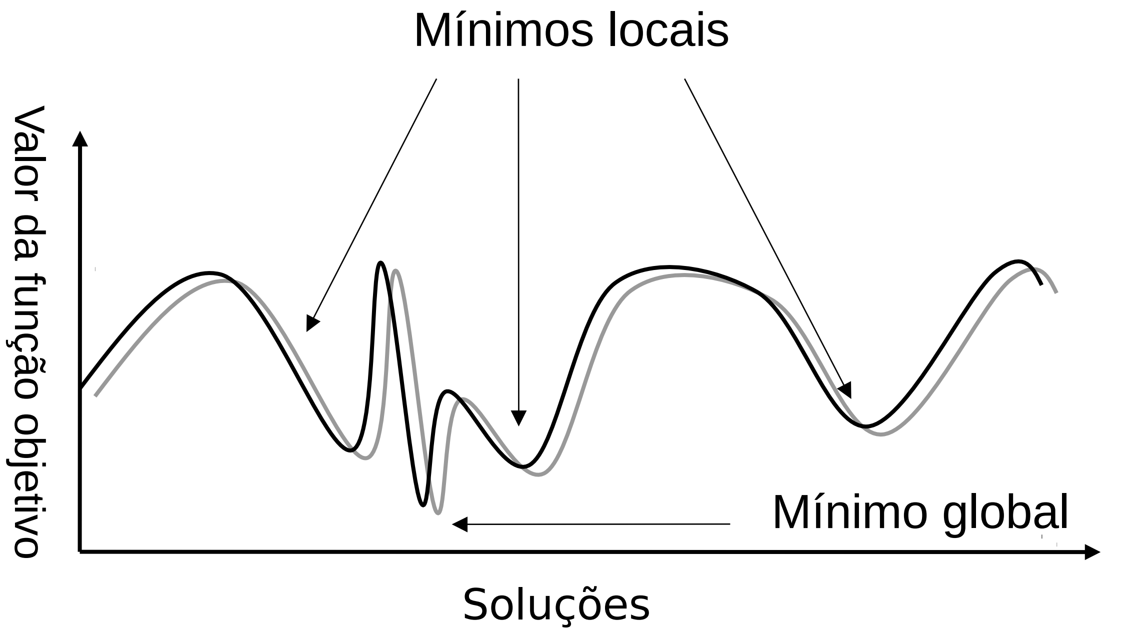
Busca local

- Espaço de soluções



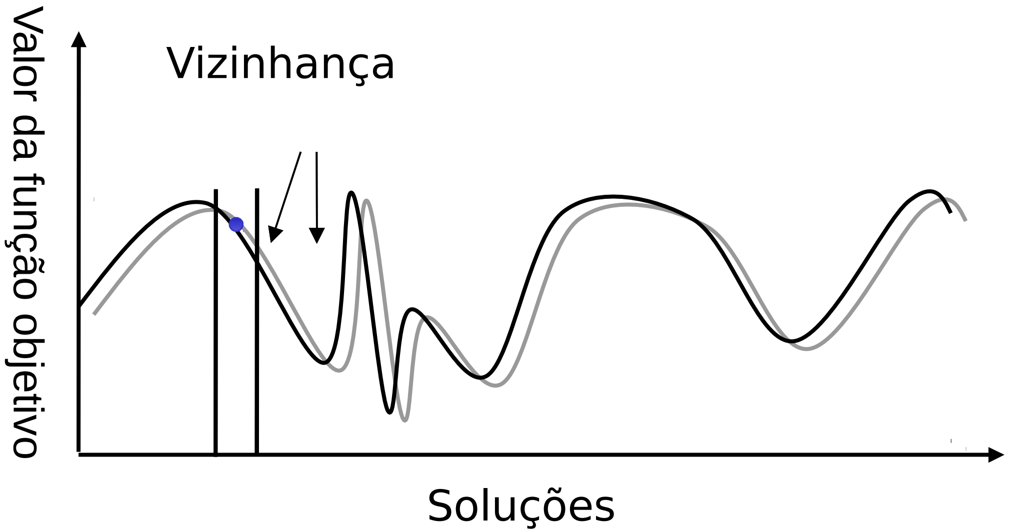
Busca local

- Espaço de soluções



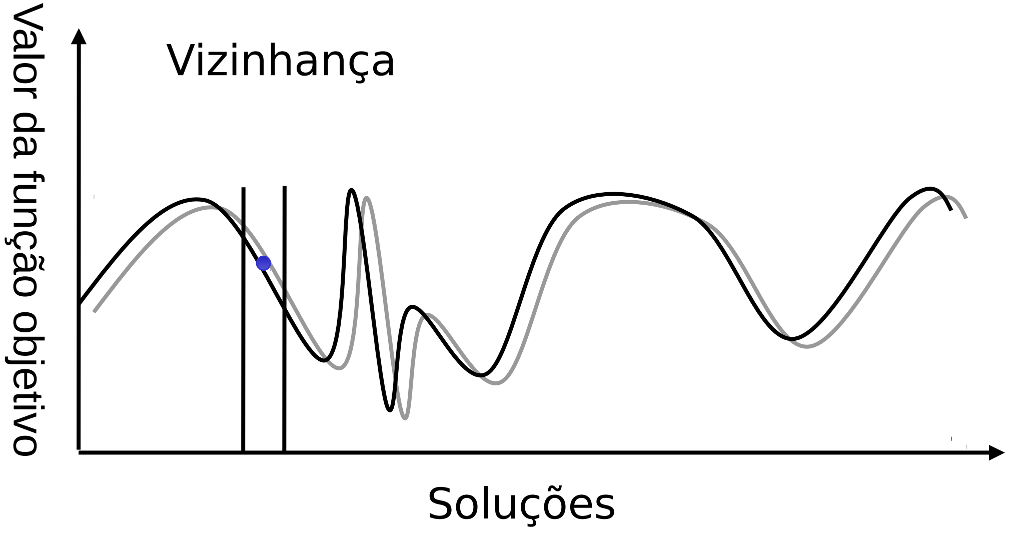
Busca local

- Espaço de soluções



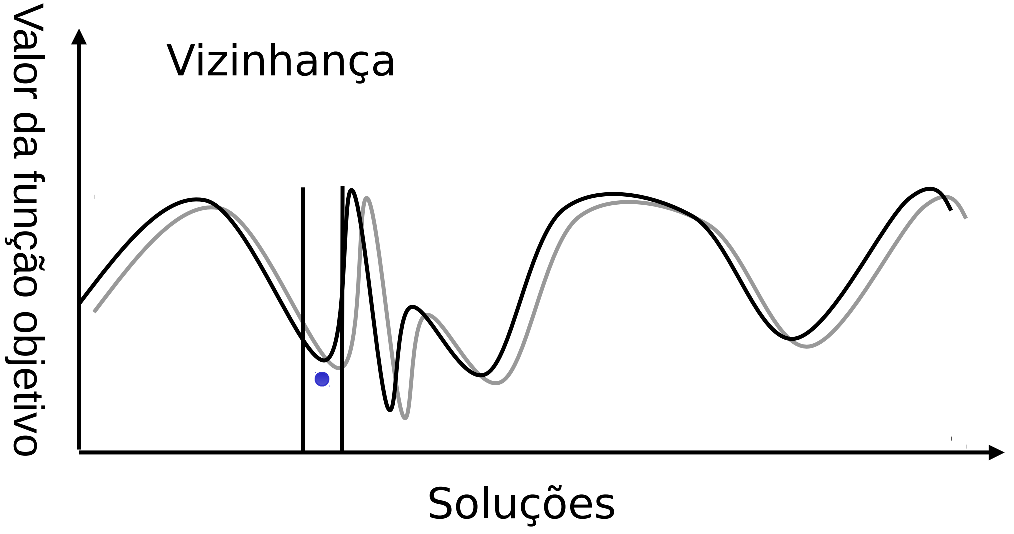
Busca local

- Espaço de soluções



Busca local

- Espaço de soluções



Busca local

- Problema de otimização combinatória:

  $f(s^*) = \text{mínimo}\ \{f(s): s \in S\}$

  S é um conjunto discreto de soluções

- Vizinhança: elemento que introduz a noção de proximidade entre as soluções de S.

- Uma vizinhança é um mapeamento que associa cada solução s a um subconjunto de soluções (vizinhos).

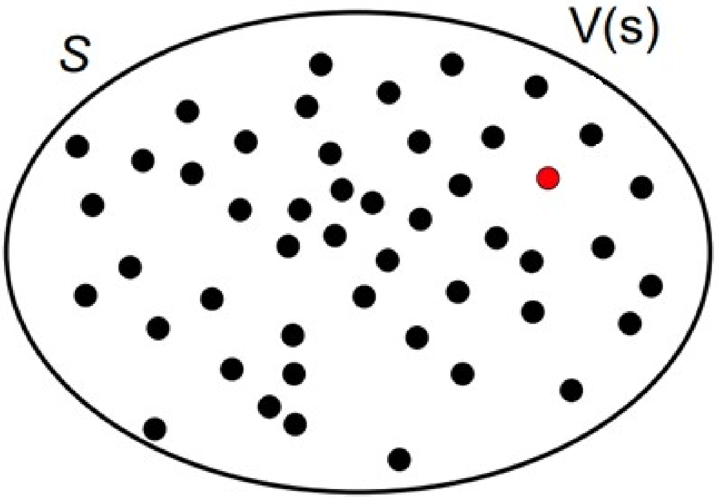
  $N(s) = \{s_1, s_2, \ldots, s_k\}$: soluções vizinhas de s

Busca local

- Boas vizinhanças permitem representar de forma compacta o conjunto de soluções vizinhas de uma solução s qualquer e percorrer (visitar, explorar) de maneira eficiente o conjunto de soluções.

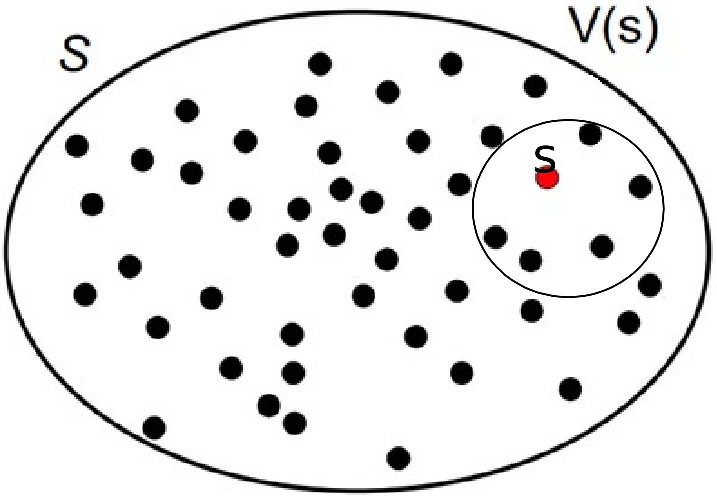
Busca local

- Vizinhança:



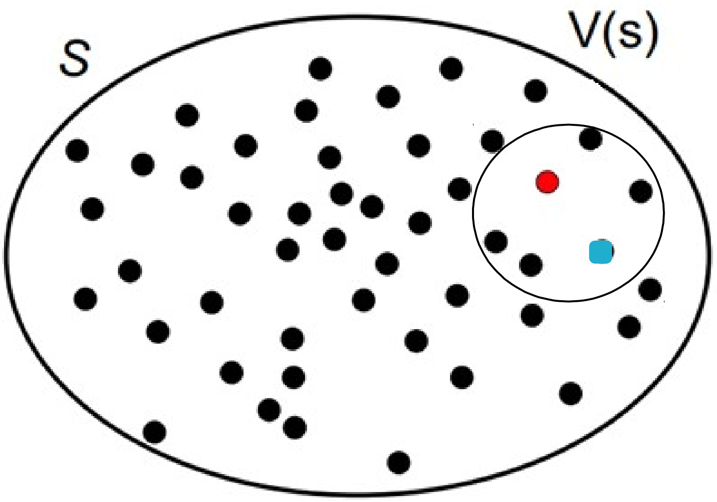
Busca local

- Vizinhança:

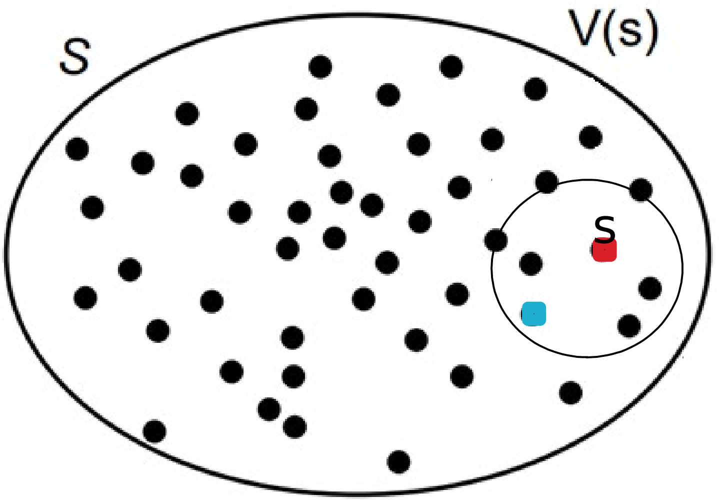


Busca local

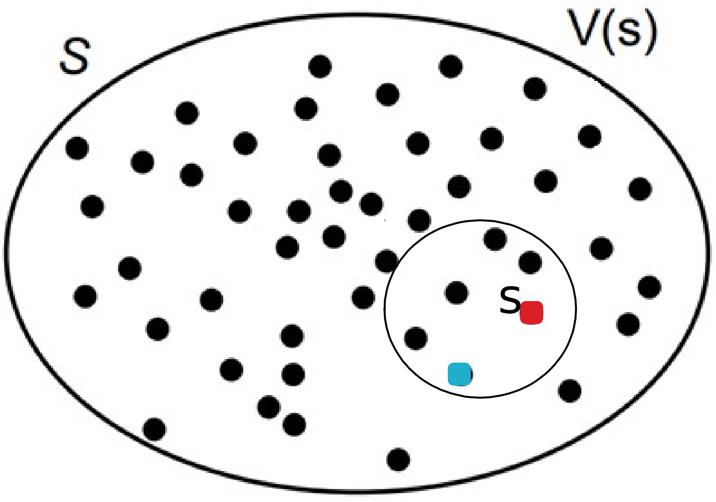
- Vizinhança:



Busca local



Busca local



Vizinhança

- Problema combinatório

$$c(s^*) = \text{mínimo } \{c(s): s \in S\}$$

  S é um conjunto discreto de soluções

- Vizinhança
  - elemento que introduz a noção de proximidade entre as soluções em S
- Uma vizinhança é um mapeamento

$$N: S \rightarrow 2^{|S|}$$

que define para cada solução $s \in S$ um conjunto $N(s) \subseteq S$ de soluções que estão, de certa forma, próximas de s

Vizinhança

- Boas vizinhanças permitem representar de forma compacta/eficiente o conjunto de soluções vizinhas a qualquer solução s
  - a escolha de uma boa vizinhança é fundamental para o sucesso de uma heurística
- O conjunto $N(s) = \{ s_1, s_2, \ldots, s_k \}$ é chamado de vizinhança de s, e cada s_i, $i = 1, \ldots, k$ de vizinho de s
  - em outras palavras, a vizinhança $N(s)$ da solução s é um conjunto de soluções que podem ser atingidas a partir de s através de operações simples

Vizinhança

- Qualquer relação de vizinhança induz um grafo no espaço de busca
  - os vértices são as soluções e
  - existem arestas entre pares de vértices associados a soluções vizinhas

Vizinhança

- Espaço de busca: definido pelo conjunto de soluções S e por uma vizinhança N
- Representação por um grafo:
  - Nós → soluções
  - Arestas → soluções vizinhas

Vizinhança

- Muitas propriedades importantes da relação de vizinhança é refletida no grafo
  - muitas relações de vizinhança padrão são
    - simétricas
  - o grau de cada vértice corresponde ao número de vizinhos
  - em muitas vizinhanças todos os vértices possuem o mesmo grau
    - o grafo é regular

Vizinhança

- Existem padrões de relações de vizinhança que formam a base de muitas heurísticas de sucesso
  - incluir um elemento na solução
    - a vizinhança possui $O(n)$ vizinhos
  - retirar um elemento da solução
    - a vizinhança possui $O(n)$ vizinhos
  - incluir um elemento na solução e retirar um elemento da solução
    - a vizinhança possui $O(n^2)$ vizinhos
  - trocar a posição de dois elementos na solução (swap)
  - alterar a posição de um elemento (shift)

Vizinhança

Vizinhanças no espaço de permutações

Solução $\pi = (\pi_1,\dots,\pi_{i-1},\pi_i,\pi_{i+1},\dots,\pi_j,\dots,\pi_n)$

$N_1(\pi) = \{(\pi_1,\dots,\pi_{i+1},\pi_i,\dots,\pi_n): i = 1,\dots, n-1\}$

Vizinhos de $(1,2,3,4) =$

$i = 1 \Rightarrow \dots, \pi_1, \pi_2, \dots \Rightarrow (2,1,3,4)$

$i = 2 \Rightarrow \dots, \pi_2, \pi_3, \dots \Rightarrow (1,3,2,4)$

$i = 3 \Rightarrow \dots, \pi_3, \pi_4, \dots \Rightarrow (1,2,4,3)$

$= \{(2,1,3,4), (1,3,2,4), (1,2,4,3)\}$

troca duas posições adjacentes

Vizinhança

$$N_2(\pi) = \{ (\pi_1,\ldots,\pi_j,\ldots,\pi_i,\ldots,\pi_n) :$$
$$i = 1,\ldots,n-1;\ j=i+1,\ldots,n \}$$

Vizinhos de (1,2,3,4) =

$i = 1, j = 2 \Rightarrow \ldots,\pi_2, \ldots, \pi_1, \ldots \Rightarrow (2,1,3,4)$

$i = 1, j = 3 \Rightarrow \ldots,\pi_3, \ldots, \pi_1, \ldots \Rightarrow (3,2,1,4)$

$i = 1, j = 4 \Rightarrow \ldots,\pi_4, \ldots, \pi_1, \ldots \Rightarrow (4,2,3,1)$

$i = 2, j = 3 \Rightarrow \ldots,\pi_3, \ldots, \pi_2, \ldots \Rightarrow (1,3,2,4)$

$i = 2, j = 4 \Rightarrow \ldots,\pi_4, \ldots, \pi_2, \ldots \Rightarrow (1,4,3,2)$

$i = 3, j = 4 \Rightarrow \ldots,\pi_4, \ldots, \pi_3, \ldots \Rightarrow (1,2,4,3)$

= { (2,1,3,4), (1,3,2,4), (1,2,4,3), (3,2,1,4), (1,4,3,2), (4,2,3,1) }

troca dois elementos (swap)

Vizinhança

$$N_3(\pi) = \{ (\pi_1,\ldots, \pi_{i-1},\pi_{i+1},\ldots,\pi_j,\pi_i,\ldots,\pi_n) :$$
$$i = 1,\ldots,n-1;\ j=i+1,\ldots,n \}$$

Vizinhos de (1,2,3,4) =

$i = 1, j = 2 \Rightarrow \ldots,\pi_0, \pi_2, \ldots, \pi_2, \pi_1, \ldots, \pi_n \Rightarrow (2,1,3,4)$

$i = 1, j = 3 \Rightarrow \ldots,\pi_0, \pi_2, \ldots, \pi_3, \pi_1, \ldots, \pi_n \Rightarrow (2,3,1,4)$

$i = 1, j = 4 \Rightarrow \ldots, \pi_0, \pi_2, \ldots, \pi_4, \pi_1, \ldots, \pi_n \Rightarrow (2,3,4,1)$

$i = 2, j = 3 \Rightarrow \ldots,\pi_3, \pi_3, \ldots, \pi_3, \pi_2, \ldots, \pi_n \Rightarrow (1,3,2,4)$

$i = 2, j = 4 \Rightarrow \ldots,\pi_1, \pi_3, \ldots, \pi_4, \pi_2, \ldots, \pi_n \Rightarrow (1,3,4,2)$

$i = 3, j = 4 \Rightarrow \ldots,\pi_2, \pi_4, \ldots, \pi_4, \pi_3, \ldots, \pi_n \Rightarrow (1,2,4,3)$

= { (2,1,3,4), (2,3,1,4), (2,3,4,1), (1,3,2,4), (1,3,4,2), (1,2,4,3) }

altera a posição de um elemento (shift)

Vizinhança: exemplo 1

- Espaço de busca
  - definido pelo conjunto de soluções S e por uma vizinhança N

- permutações com a vizinhança N_1

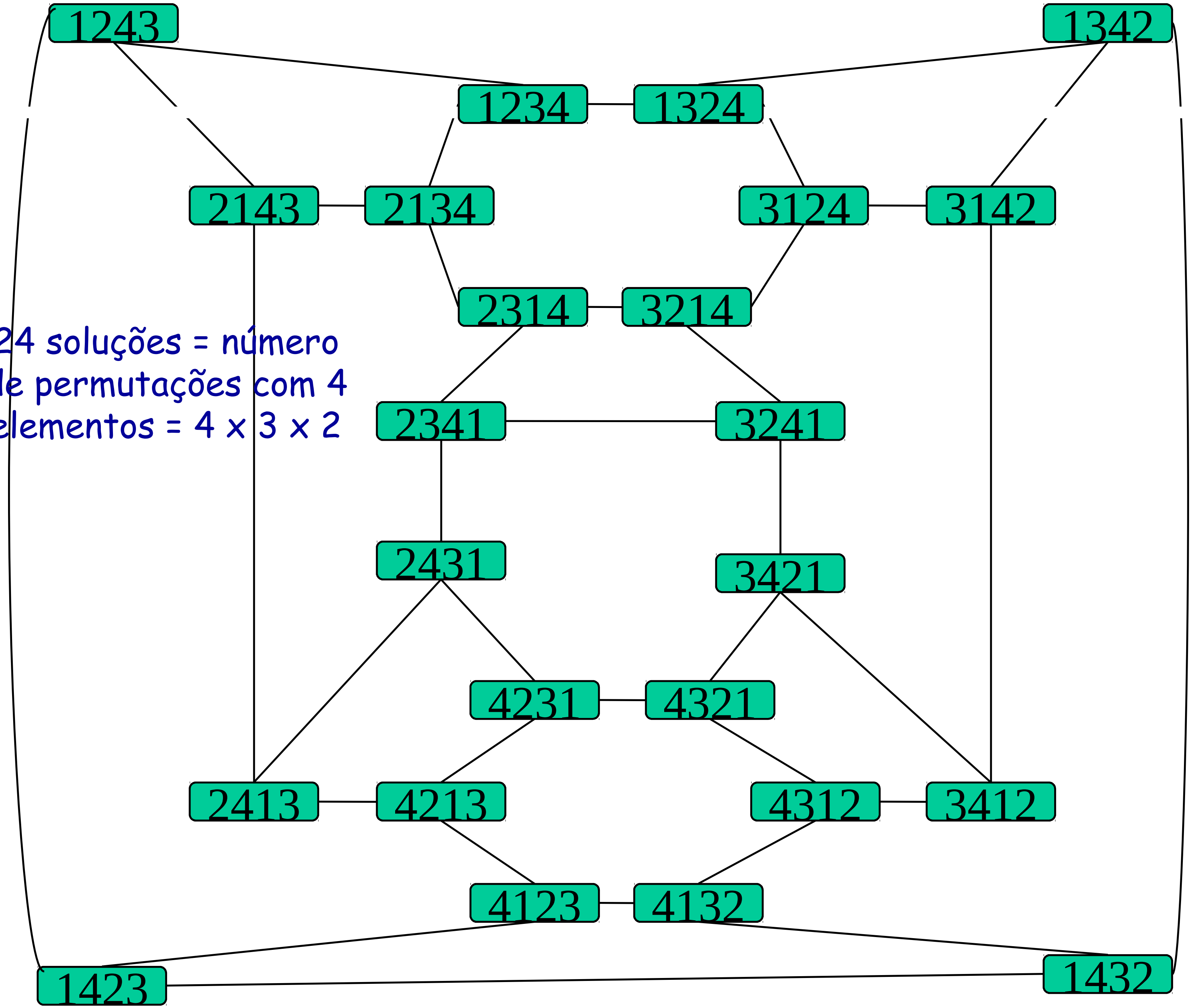
$$N_1(\pi) = \{ (\pi_1,\dots,\pi_{i+1},\pi_i,\dots,\pi_n): i = 1,\dots, n-1 \}$$

Vizinhos de

(1,2,3,4) = { (2,1,3,4), (1,3,2,4), (1,2,4,3) }

(1,3,2,4) = { (3,1,2,4), (1,2,3,4), (1,3,4,2) }

(1,2,4,3) = ...



24 soluções = número de permutações com 4 elementos = 4 x 3 x 2

Vizinhança: exemplo 2

- Vetores de pertinência 0-1 com a vizinhança N_4

  $v = (v_1, \ldots, v_i, \ldots, v_n)$

  $N_4(v) = \{ (v_1, \ldots, 1\text{-}v_i, \ldots, v_n) : i = 1,\ldots,n \}$

  Vizinhos de $(1,0,1,1)$ =

  $i = 1 \Rightarrow (0,0,1,1)$

  $i = 2 \Rightarrow (1,1,1,1)$

  $i = 3 \Rightarrow (1,0,0,1)$

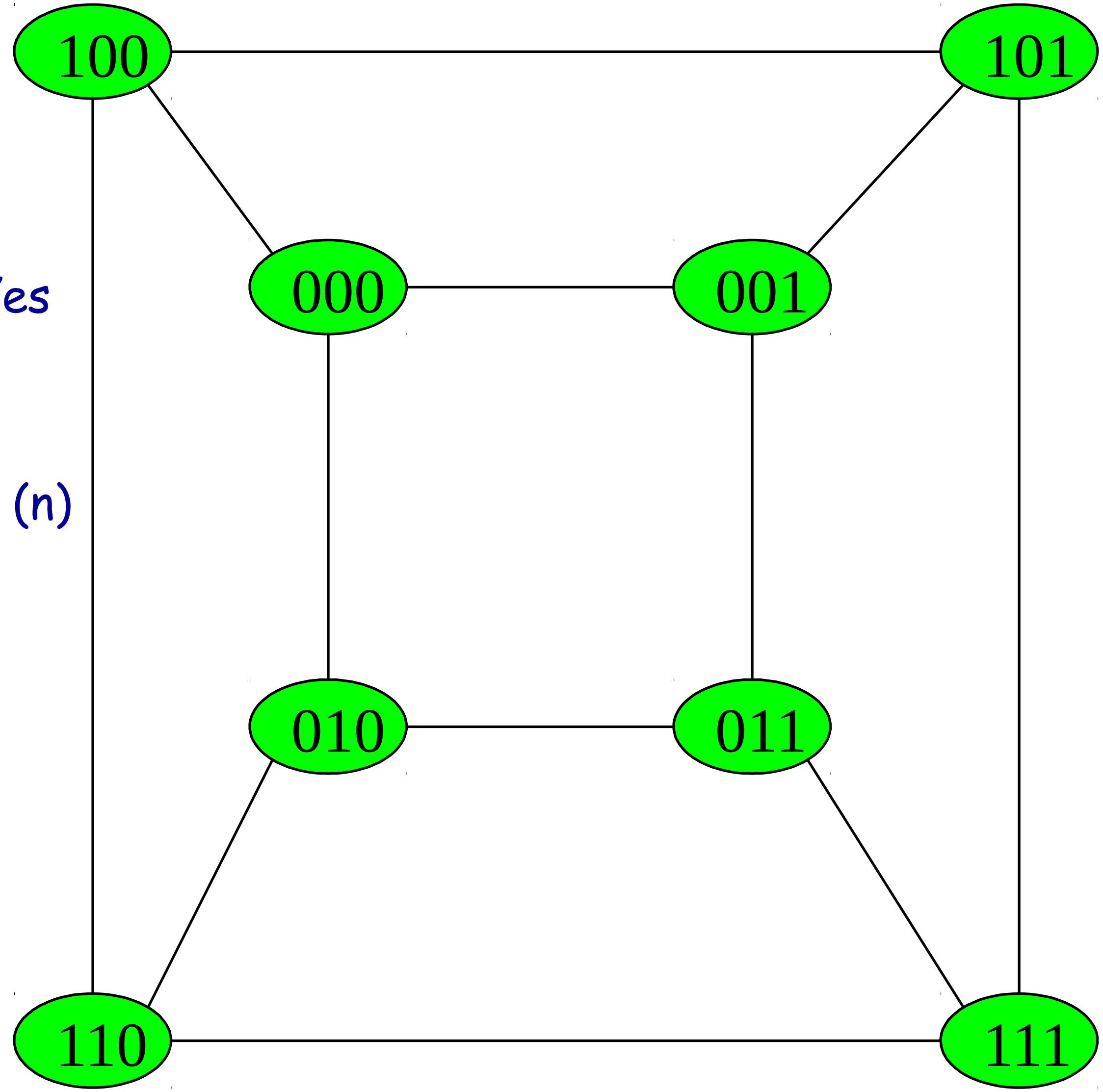
  $i = 4 \Rightarrow (1,0,1,0)$

$= \{ (0,0,1,1), (1,1,1,1), (1,0,0,1), (1,0,1,0) \}$

Vizinhança

2^3= 8 soluções

diâmetro = 3 (n)

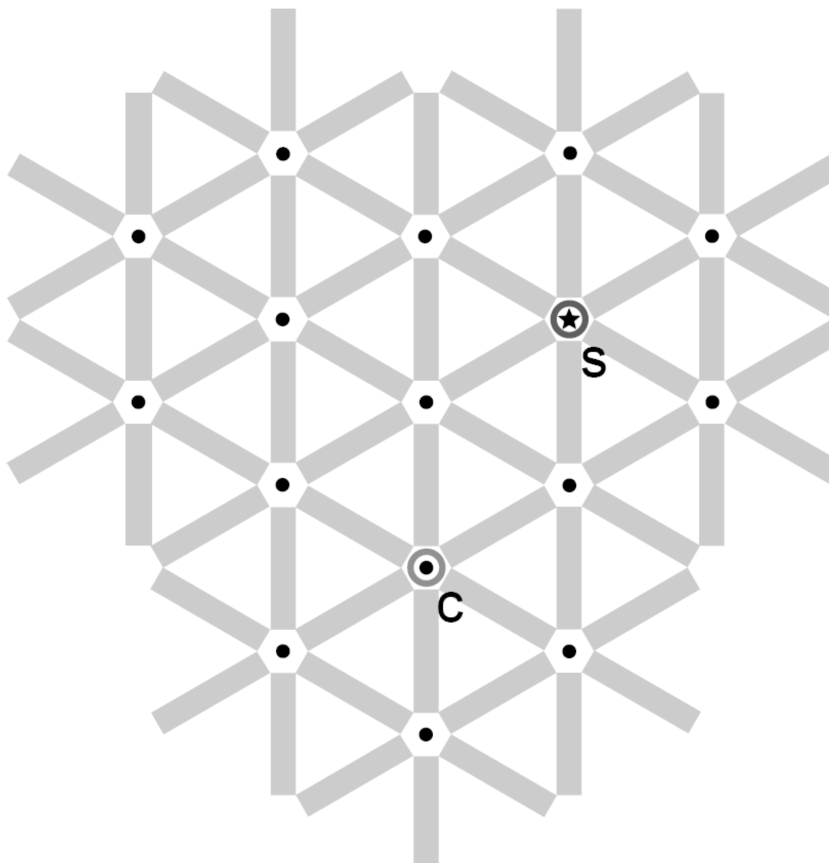


Vizinhança

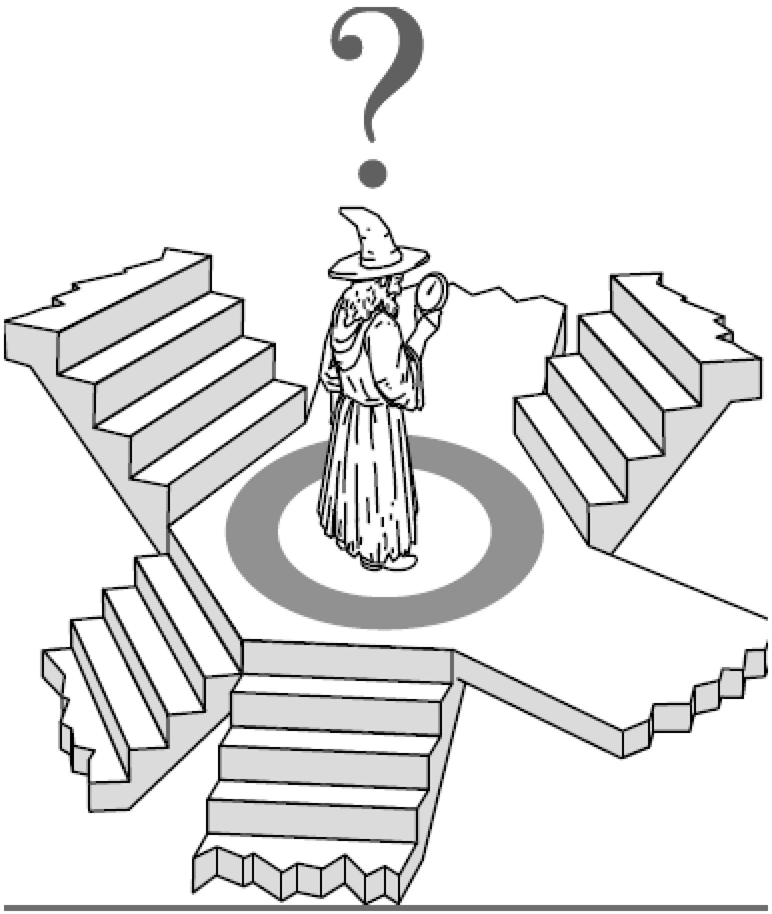
- distância entre duas soluções
  - número de soluções visitadas ao longo de uma caminho mais curto entre elas
- diâmetro
  - distância entre duas das soluções mais afastadas
  - informação sobre o pior caso no número de movimentos para se chegar a solução (ótima) partindo-se de pontos (vértices) arbitrários no espaço de busca

Busca Local

- Visão Global
  - espaço de busca
  - vértices: soluções candidatas
  - s: solução ótima
  - c: posição corrente na busca
  - vizinhos estão conectados por linhas



Busca local



- Visão Local
  - em cada passo da busca o processo de busca move para uma posição vizinha escolhida com base somente em informações locais

Busca local

- Tipicamente os componentes
  - espaço de busca, conjunto de soluções e relação de vizinhança
  - fornecem a base de muitas métodos heurísticos de busca local
- Entretanto, baseados em uma dada definição de
  - um espaço de busca, um conjunto de soluções e uma relação de vizinhança, um vasto conjunto de estratégias de busca especificadas pela definição de solução inicial e passo da busca (movimento) pode ser derivada

Busca local

- Definições
  - passo da busca (movimento)
    - um par de soluções vizinhas s e s'
    - s pode ser alcançado de s', e vice-versa
  - trajetória de busca (caminho no espaço de busca)
    - sequência finita de posições no espaço de busca (soluções)
    - onde duas soluções consecutivas quaisquer são vizinhas

Busca local

- Definições
  - estratégia de busca
    - especificada pela definição de solução inicial e movimento
    - de alguma forma, independente da instância do problema
    - em geral, qualquer estratégia de busca corresponde a um caminho no grafo de vizinhança

Busca local

- Estratégia de Melhoria Iterativa
  - estratégia que guia a busca na direção de melhorar a qualidade da solução corrente
  - dois critérios básicos de seleção são
    - primeira-melhoria (first-improvement)
      - seleciona qualquer (eventualmente a primeira) solução que aprimora a solução corrente, na vizinhança
    - descida mais rápida (best-improvement)
      - seleciona a melhor solução entre todas as soluções da vizinhança

Busca local

- Função de avaliação
  - mecanismo usado para guiar a busca
  - função que mapeia soluções candidatas de uma instância de uma dado problema a um valor (número real) de tal forma que o ótimo global corresponda a solução ótima
  - é usada para rankear soluções candidatas na vizinhança da solução corrente
  - tipicamente é dependente do problema
  - escolha da função de avaliação, de alguma forma, depende do espaço de soluções, conjunto de soluções e vizinhança

Busca local

- Função de avaliação x função de otimização
  - função de avaliação
    - parte do algoritmo de busca local
  - função de otimização
    - parte do problema de otimização
  - frequentemente, em algoritmos para problemas de otimização combinatória
    - a função de avaliação corresponde ao que se deseja otimizar
  - entretanto, as vezes, funções diferentes podem guiar melhor a busca
- na literatura, raramente é feita a distinção entre função de avaliação e função objetivo

Busca local - melhoria iterativa

```
procedure melhoria-iterativa-first-improvement(s0)
  s ← s0;
  melhoria ← .verdadeiro.
  while  melhoria do
     melhoria ← .falso.
     for-all s'∈N(s) e melhoria =.falso. do
        if  f(s') < f(s) then
           s ← s';
           melhoria ← .verdadeiro.
        end-if
      end-for-all
   end-while
   return s
end melhoria-iterativa-first-improvement
```

a cada iteração, seleciona qualquer (eventualmente a primeira) solução da vizinhança que aprimora a solução corrente

termina quando não existe vizinho que melhore a solução

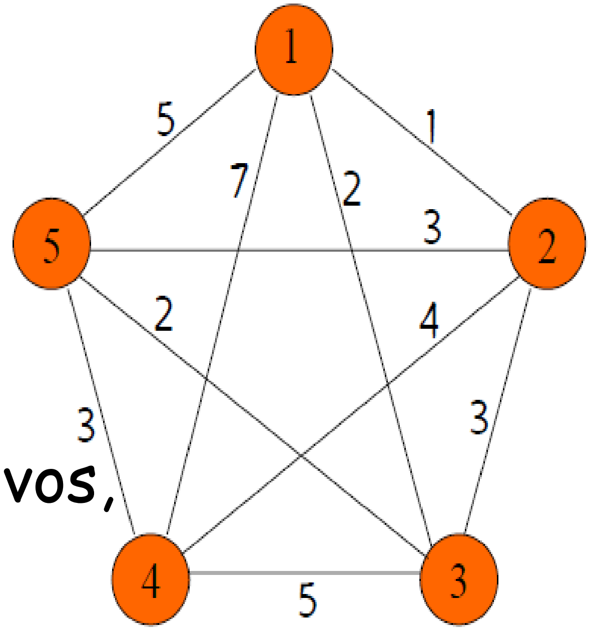
Hill Climbing (melhoria-iterativa-first-improvement)

- Uma solução do PCV (permutação)

S: 1-2-3-4-5-1

Custo: 17

- A vizinhança de s é formada pela troca de qualquer par de nós sucessivos, conservando a origem.



s: 1-2-3-4-5-1 (17)

s=s': 1-2-3-5-4-1 (16) → c(s')<c(s) então s=s'

s': 1-3-2-5-4-1 (18) → c(s')>c(s) fim

- Ótimo local s: 1-2-3-5-4-1 (16)

Busca local - melhoria iterativa

```
procedure descida-mais-rápida(s0)
  s ← s0; melhoria ← .verdadeiro.
  while melhoria do
     melhoria ← .falso.; fmin ← +∞
     for-all s' ∈ N(s) do
        if  f(s') < fmin then
            smin ← s'; fmin ← f(s')
                end-if
     end-for-all
     if  fmin <  f(s)  then
            s ← smin ; melhoria ← .verdadeiro.
     end-if
   end-while
   return s
end descida-mais-rápida
```

seleciona a melhor solução (que aprimora a solução corrente) da vizinhança

Steepest Descent (melhoria-iterativa-best-improvement)

s: 1-2-3-4-5-1 (17)

1-3-2-4-5-1 (17)

1-2-4-3-5-1 (17)

s = s*: 1-2-3-5-4-1 (16)

c(s*) < c(s) então s=s*

s*:1-3-2-5-4-1 (18)

1-2-5-3-4-1 (18)

c(s*) > c(s) fim

Ótimo local s:1-2-3-5-4-1 (16)

Busca local

- Ótimo local
  - solução tão boa ou melhor do que qualquer das soluções vizinhas
- Problema de minimização

s^+ é um ótimo local

$$\Uparrow\Downarrow$$

$$c(s^+) \leq c(s),\ \forall\ s \in N(s^+)$$

- Ótimo global ou solução ótima S^*

$$c(s^*) \leq c(s), \qquad \forall\ s \in F$$

Busca local

- Atualizações incrementais (valor delta)
  - ideia chave
    - após cada passo da busca, calcular o efeito das diferenças entre a solução corrente s e a solução vizinha s' no valor da função de avaliação
    - valores da função de avaliação, muitas vezes, consistem de contribuições independentes dos componentes da solução
    - consequentemente o valor de s' pode ser calculado de forma eficiente usando-se o valor de s, através das diferenças entre s e s' em termos de componentes da solução
  - questão fundamental para a implementação eficiente de algoritmos de busca local

Busca local

- Dificuldades
  - término prematuro no primeiro ótimo local encontrado
    - é comum acontecer
    - a solução corrente não é suficientemente boa
  - sensível à solução de partida
  - sensível à vizinhança escolhida
  - sensível à estratégia de busca
  - pode exigir um número exponencial de iterações!

Busca local

- Escolhas importantes (questões fundamentais)
  - solução inicial viável
    - as vezes é conveniente executar a busca partindo de diferentes soluções iniciais
      - neste caso, precisa-se decidir quantas pontos de partida e como distribuí-los

Busca local

- Escolhas importantes (questões fundamentais)
  - definição da vizinhança para o problema específico
    - geralmente esta escolha é guiada pela intuição pois não existe teoria disponível para funcionar como "guia"
    - existe um trade off entre vizinhanças pequenas e vizinhanças grandes
      - uma vizinhança grande "promete" fornecer melhores ótimos locais
        - » porém gastará mais tempo
      - espera-se então que poucas delas possam ser exploradas em um determinado tempo
      - geralmente, o diâmetro do grafo de uma vizinhança grande é menor do que o de uma vizinhança menor
        - » potencialmente o caminho da busca pode explorar mais regiões do espaço de busca

Busca local

- Escolhas importantes (questões fundamentais)
  - definição da vizinhança para o problema específico
    - idealmente o tamanho seria aquele para o qual qualquer solução candidata que seja um ótimo local, seja também um ótimo global
      - vizinhanças exatas
      - geralmente de tamanho exponencial em relação ao tamanho do problema
    - o que é melhor, poucas vizinhanças grandes ou mais vizinhanças menores?
      - esta questão e outras similares, geralmente são respondidas de forma experimental

Busca local

- Escolhas importantes (questões fundamentais)
  - complexidade de cada iteração
    - proporcional ao tamanho da vizinhança
    - a complexidade de tempo de uma iteração de uma busca local deve ser polinomial em relação ao tamanho do problema
      - mesmo assim, dependendo do tamanho do problema, tempo quadrático ou cúbico pode ser proibitivo
    - eficiência depende da forma como é calculada a variação da função objetivo para cada solução vizinha
    - algoritmos eficientes são capazes de recalcular as variações de modo a atualizá-las quando a solução corrente se modifica
      - evitando cálculos repetitivos e desnecessários da função objetivo

Busca local

- Extensões para contornar algumas dificuldades da busca local
  - redução da vizinhança
    - investigar um subconjunto da vizinhança da solução corrente
    - por exemplo, para o TSP, usar lista de candidatos
      - para cada vértice incluir na lista um número limitado dos vizinhos mais próximos, ordenados de forma crescente pelo peso da aresta
      - o passo da busca fica limitado a considerar somente arestas que conectam o vértice i com um dos vértices na lista de candidatos de i
      - intuição: soluções de qualidade incluem arestas de tamanho pequeno

Busca local

- Extensões para contornar algumas dificuldades da busca local (continuação)
  - multi-partida
    - repetir a busca local a partir de diferentes soluções sempre quando encontrar um ótimo local
  - permite movimento que não melhora a solução
    - quando encontrar um ótimo local, permite a seleção de soluções candidatas com valor igual ou pior

Busca local

- Extensões para contornar algumas dificuldades da busca local (continuação)
  - multi-vizinhança
    - considera mais de uma vizinhança
    - ao atingir um ótimo local com relação a uma vizinhança, inicia uma outra busca local empregando outra vizinhança
    - o algoritmo termina quando a solução corrente é um ótimo local em relação a todas as vizinhanças empregadas

Intensificação versus Diversificação

- A utilização de escolhas aleatórias como parte do processo de busca local
  - pode levar a uma melhoria significativa na performance e robustez do algoritmo
- Por outro lado, é necessário balancear componentes
  - aleatórios e
  - direcionados ao objetivo da estratégia de busca

intensificação versus diversificação

Intensificação versus Diversificação

- Intensificação
  - estratégia de busca que objetiva melhorar
    - a qualidade da solução de forma gulosa ou
    - as chances de encontrar uma solução no futuro próximo explorando, por exemplo, o caminho dado pela função de avaliação

Intensificação versus Diversificação

- Diversificação
  - estratégia de busca que
    - tenta prevenir a estagnação da busca certificando-se que o processo de busca consiga uma cobertura razoável quando explorando o espaço de busca e não fique estagnado em regiões relativamente confinadas que não possuam soluções de qualidade suficientemente boas

Intensificação versus Diversificação

- Existe uma grande variedade de técnicas que combinam intensificação e diversificação
  - ao mesmo tempo que o resultado de algoritmos que utilizam essas técnicas funcionam bem na prática
  - tipicamente, o seu comportamento não é bem entendido
  - aplicação com sucesso é, na maioria das vezes, baseada em intuição e experiência
    - ao invés de princípios ou insights teóricos ou empíricos
  - conhecimento sobre o problema é fundamental para conseguir performance de qualidade e robustez