

Práctica 1. Busca Local

Nesta práctica aplicaremos un algoritmo de Busca Local ao problema do “viaxante de comercio”. O código desenvolvido servirá como base para o resto das prácticas, tanto na estrutura xeral coma na implementación dalgúns dos métodos, que serán comúns a todas as prácticas de metaheurísticas.

Algoritmo Busca Local (especificación obrigatoria)

Neste apartado desenvolverase unha versión inicial da busca local do **primeiro mellor**, seguindo o esquema algorítmico xeral descrito na diapositiva 4.2.14. Todos deberedes utilizar o mesmo esquema algorítmico, coas especificacións obrigatorias que se indican de seguido:

1. Representación das solucións

Usaremos unha **representación de orde** formada por secuencia dos índices das cidades. Por exemplo, se tomamos (sen perda de xeneralidade) como cidade de partida e fin a de índice 0, o percorrido vaise representar, para o problema de $n=10$ cidades, coma unha permutación dos índices 1-9:

$$\{1\ 3\ 2\ 4\ 5\ 6\ 7\ 9\ 8\}$$

Non é preciso incluír a cidade de partida e fin na representación final do percorrido, xa que sempre é a mesma (índice 0). Nesta primeira práctica trataremos o problema de $n=10$ cidades, polo que a lonxitude de cada solución é sempre de 9 índices enteiros no rango $[1, n-1]$.

2. Solución inicial

A solución inicial será **totalmente aleatoria**.

3. Xeración da veciñanza dunha solución

Utilizaremos o operador de **intercambio de dous elementos**, descrito na transparencia 4.2.22. Para este operador o número máximo de veciños diferentes que é posible xerar a partir dunha solución dada (todos os posibles intercambios) é:

$$\sum_{i=1}^{n-2} i = \frac{(n-1) \cdot (n-2)}{2}$$

Para o caso particular $n=10$, temos un tamaño máximo da veciñanza de 36, que é o número de distintas permutacións de dous índices.

4. Función de custe

A función de custe C especificada na diapositiva 4.2.17:

$$C(S) = D(0, S[0]) + \sum_{i=1}^{n-2} D(S[i-1], S[i]) + D(S[n-2], 0)$$

Práctica 1. Busca Local

Esta función xa ten en conta que o percorrido comeza e remata na cidade de **índice 0** e asume que os valores de distancia están gardados nunha matriz de distancias D e que os índices do percorrido están gardados nunha permutación de índices S .

Disponse, como datos de partida, da táboa de distancias entre as cidades a visitar, accesible no arquivo **distancias_10.txt** que se achega coa práctica. Por evitar redundancias, o arquivo ten gardada unicamente a **matriz triangular inferior** de distancias.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|------|------|------|------|------|-----|------|------|------|
| 1 | 681 | | | | | | | | |
| 2 | 1864 | 1900 | | | | | | | |
| 3 | 882 | 354 | 1067 | | | | | | |
| 4 | 1918 | 900 | 761 | 1631 | | | | | |
| 5 | 1127 | 1915 | 1031 | 537 | 633 | | | | |
| 6 | 517 | 416 | 721 | 1371 | 1865 | 691 | | | |
| 7 | 1831 | 1418 | 1322 | 586 | 1161 | 520 | 1011 | | |
| 8 | 320 | 1722 | 544 | 1037 | 495 | 801 | 1450 | 1513 | |
| 9 | 1886 | 997 | 785 | 1070 | 1450 | 555 | 228 | 1009 | 1205 |

Para calcular os valores da matriz de distancias D a partir da información do arquivo cómpre ter en conta que:

- A columna i -ésima garda as distancias entre a cidade i -ésima e as cidades $i+1, i+2, \dots$
- A diagonal da matriz D é 0, xa que mediría (distancia entre unha cidade e ela mesma)
- A matriz é simétrica $D(i, j) = D(j, i)$

Así, por exemplo, $D(0, 1) = D(1, 0) = 681$; $D(0, 9) = D(9, 0) = 1886$; $D(8, 9) = D(9, 8) = 1205$

5. Mecanismo de selección de solucións

Tomarase como criterio de aceptación o **primeiro mellor** dos veciños.

6. Criterio de parada

Finalizaremos a busca cando o se percorra toda a veciñanza sen obter solucións mellores ca actual.

Entrega da práctica

A data límite de entrega será a que se especifique na ferramenta de entregas da Aula Virtual.

Será obrigatorio entregar o código fonte da práctica, nun arquivo comprimido autocontido (que conteña todos os arquivos precisos para a execución), e en publicar os resultados de **10 execucións diferentes do algoritmo** nunha folla de cálculo compartida (da que proporcionaremos o enlace mediante un anuncio na Aula Virtual).

Avaliación da práctica

Este traballo contará para a parte práctica da materia cunha ponderación do 15%. A avaliación do funcionamento da práctica realizarase mediante un test de autoavaliación sobre os resultados de execución da vosa implementación en diferentes casos (matrices de distancias, números aleatorios, ...)

Práctica 1. Busca Local

en diferentes etapas do proceso de busca. Nese test, para que a execución da práctica sexa totalmente determinista (non aleatoria) e, xa que logo, os resultados sexan idénticos en todas as execucións, vos facilitaremos un arquivo con números aleatorios, que substituirá á xeración aleatoria que se precisa nos apartados 2 e 3 da práctica. Indicamos de seguido a especificación para este apartado concreto:

Para a realización do test de autoavaliación o programa deberá incluír unha opción adicional relativa aos números aleatorios, que lle permita ao usuario unha destas opcións:

- 1) xeración automática dos números aleatorios segundo o especificado anteriormente
- 2) lectura dos números aleatorios a partir dun arquivo de nome **aleatorios.txt**, como o que vos facilitamos de exemplo.

Esta opción non é obrigatoria para a entrega da práctica, pero sí debes implementala para a realización do test. Para verificar o funcionamento correcto desta funcionalidade proporcionaremos un arquivo **aleatorios.txt** de proba.