

INTELIGÊNCIA ARTIFICIAL

PARTE 3

Buscas Heurísticas

- Best First
- A^*
- Hill Climbing



1

Busca Heurística

1 - Busca Heurística

Definições de Heurística:

- *Arte de inventar ou descobrir.* [dic. Aurélio]
- *Método que pretende levar a inventar, descobrir ou a resolver problemas.* [dic. Aurélio]
- *Log. Hipótese que, numa pesquisa, leva a uma descoberta científica; método analítico para a descoberta de verdades científicas.* [dicio.com.br]

1 - Busca Heurística

Definições de Heurística:

- *Heurística é um método ou processo criado com o objetivo de encontrar soluções para um problema. É um procedimento simplificador (embora não simplista) que, em face de questões difíceis envolve a substituição destas por outras de resolução mais fácil a fim de encontrar respostas viáveis, ainda que imperfeitas. Tal procedimento pode ser tanto uma técnica deliberada de resolução de problemas, como uma operação de comportamento automática, intuitiva e inconsciente.*
[wikipedia]

1 - Busca Heurística

Características:

- Utiliza conhecimento específico do problema na escolha do próximo nó a ser expandido.
- A função de avaliação estima o custo do caminho do nó atual até o objetivo, indicando o quanto o nó é promissor com relação a atingir a meta estabelecida.
- Funções heurísticas são específicas para cada problema.

1 - Busca Heurística

Como escolher uma boa função heurística?

- Deve ser admissível, ou seja, nunca superestimar o custo real da solução. Estima-se o custo do caminho mais barato do estado atual até o estado final mais próximo.
Exemplo: A rota mais curta entre duas cidades.
- Utilizar a distância direta é admissível porque o caminho mais curto entre dois pontos é sempre uma linha reta.

2

Algoritmos *Best-First* (*Busca Gulosa*)

2 - Algoritmos *Best-First* (Busca Gulosa)

Características do *Best-First Search*:

Faz a busca pelo melhor primeiro, considerando apenas o custo do nó observado até o nó objetivo, calculando sua distância em linha reta imaginária.

- **Seja:**

$h'(n)$ = função que estima o custo para se chegar do nó observado n até o nó final.

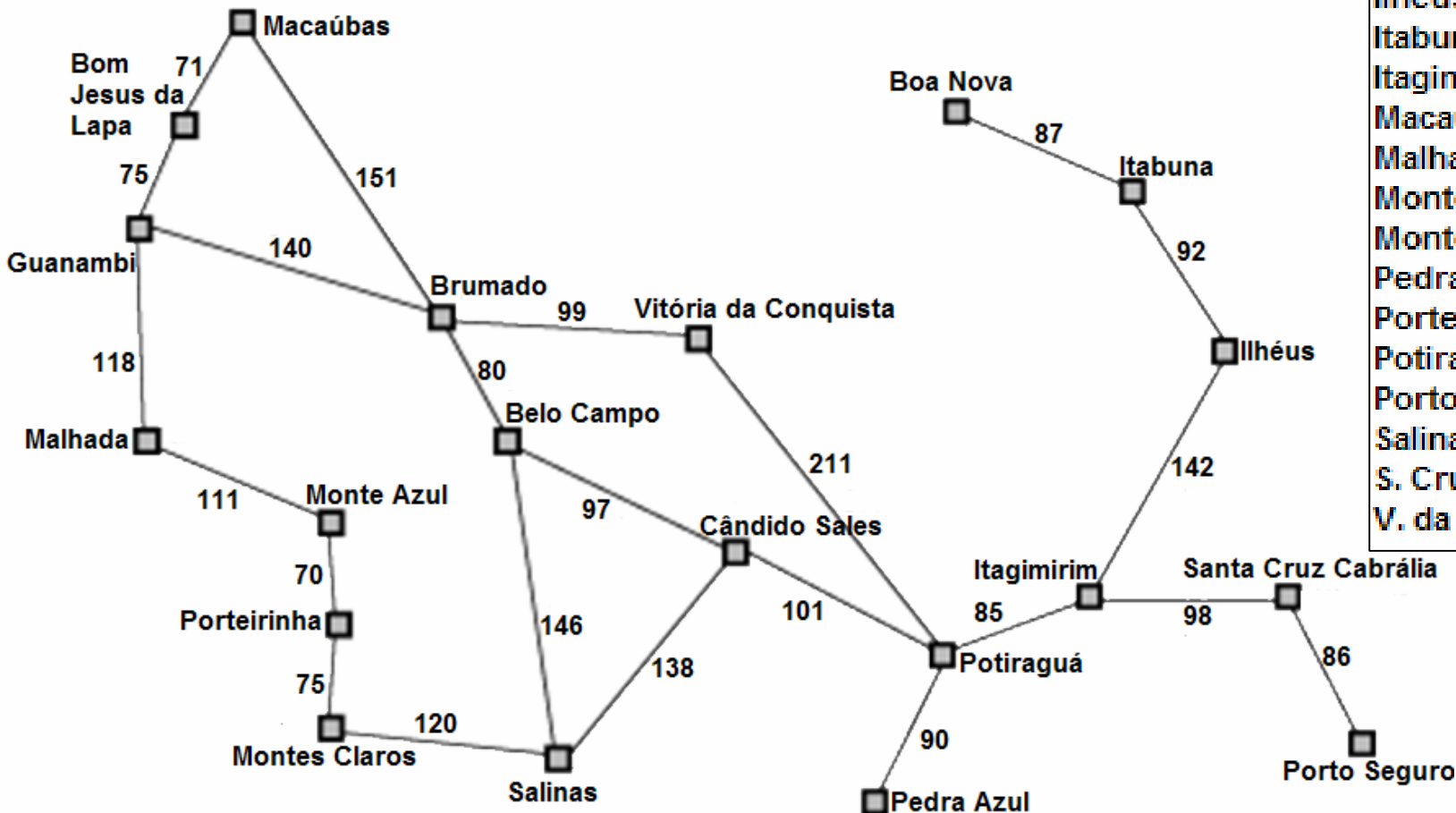
Função de avaliação: $f(n) = h'(n)$

$f(n)$ é o custo estimado da solução de custo mais baixo, passando por n .

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



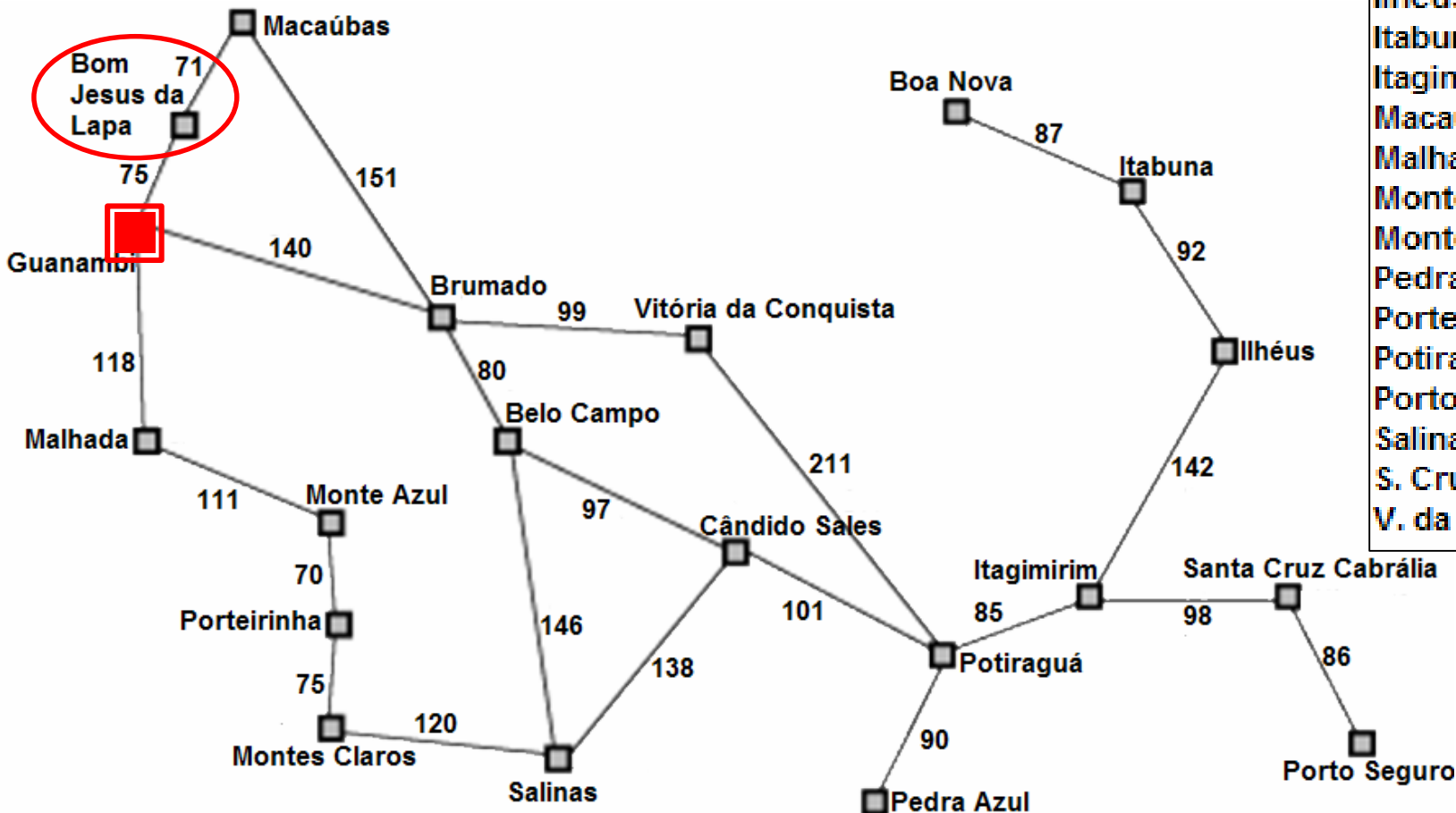
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

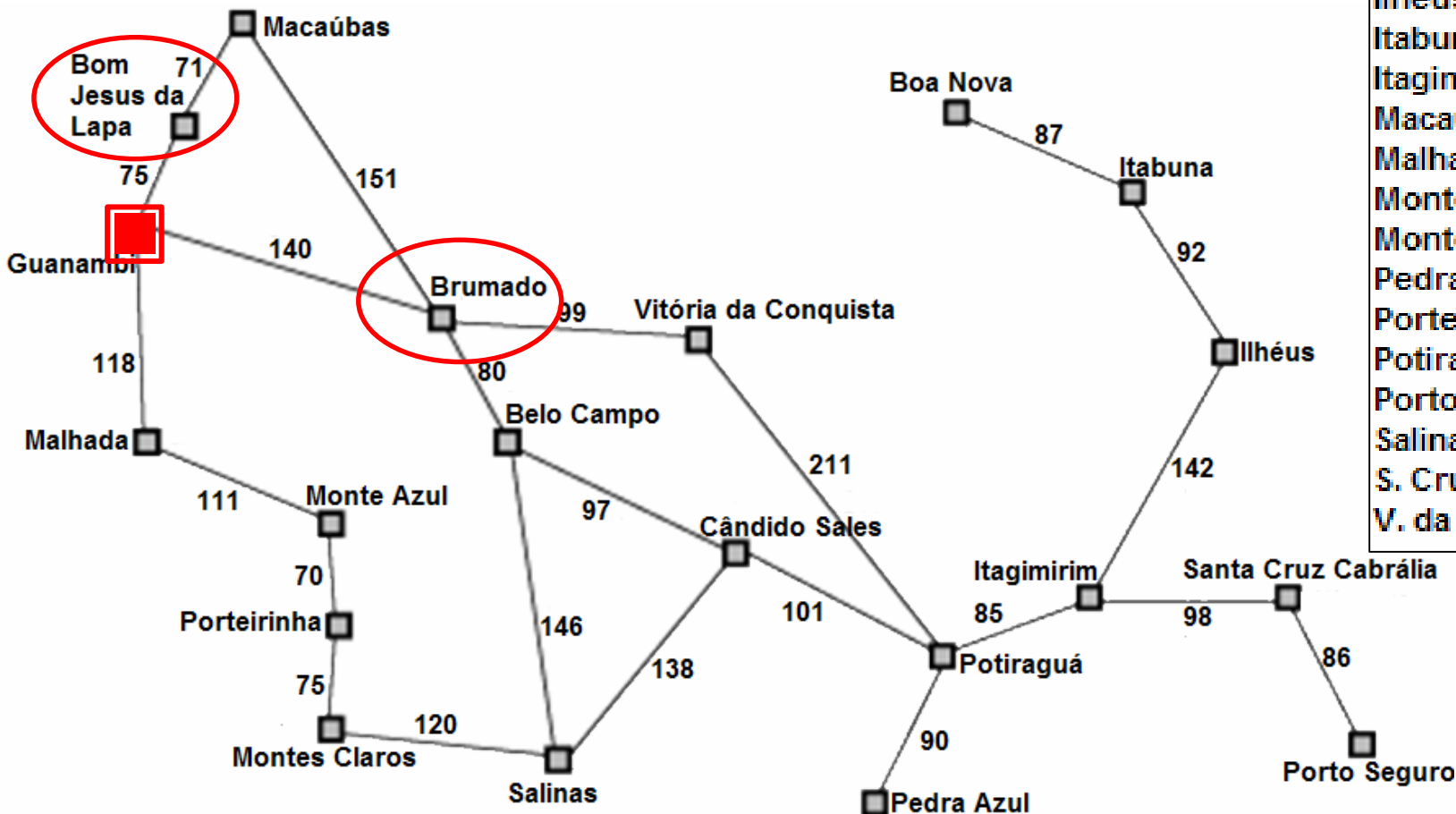


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

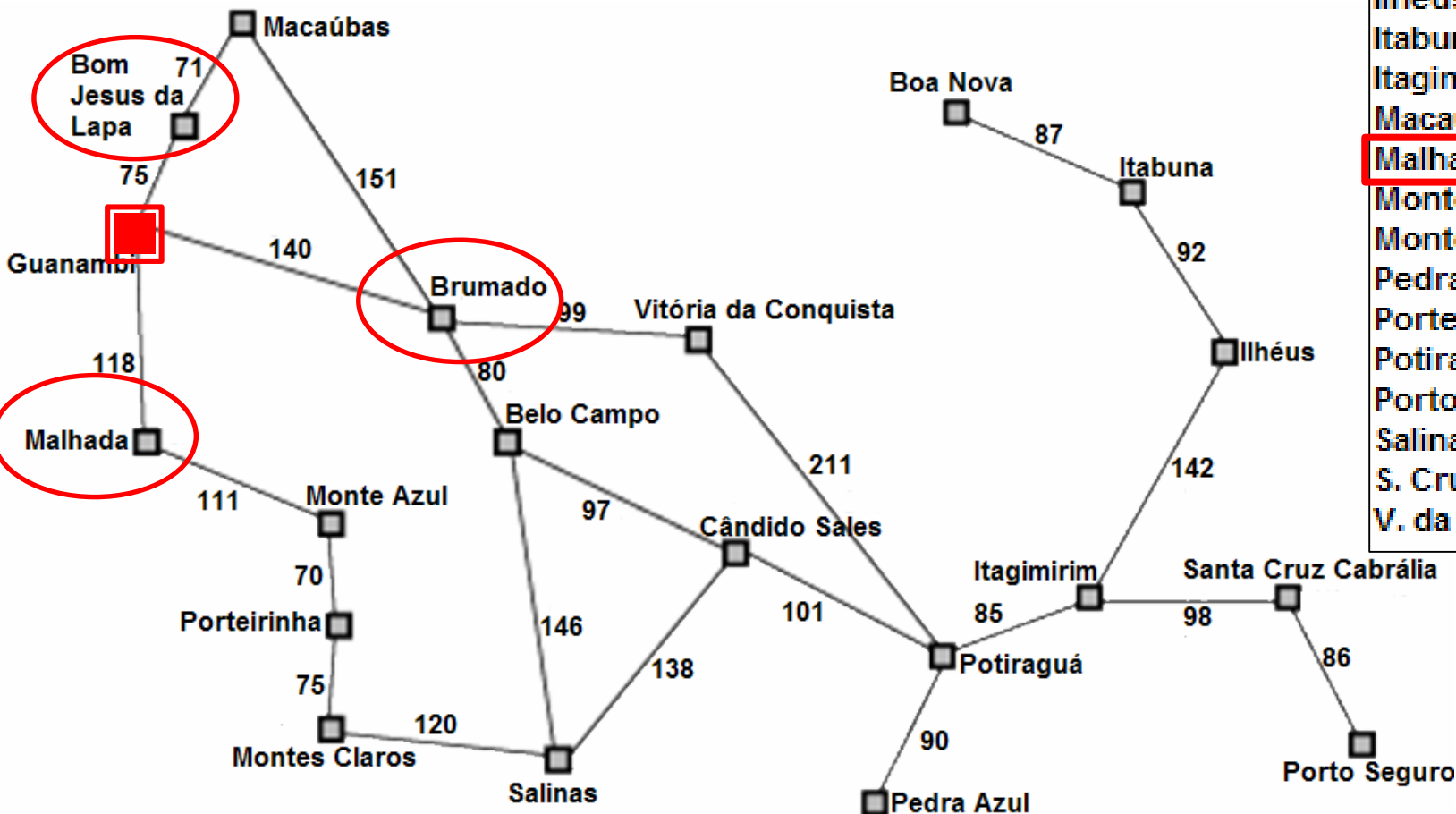


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

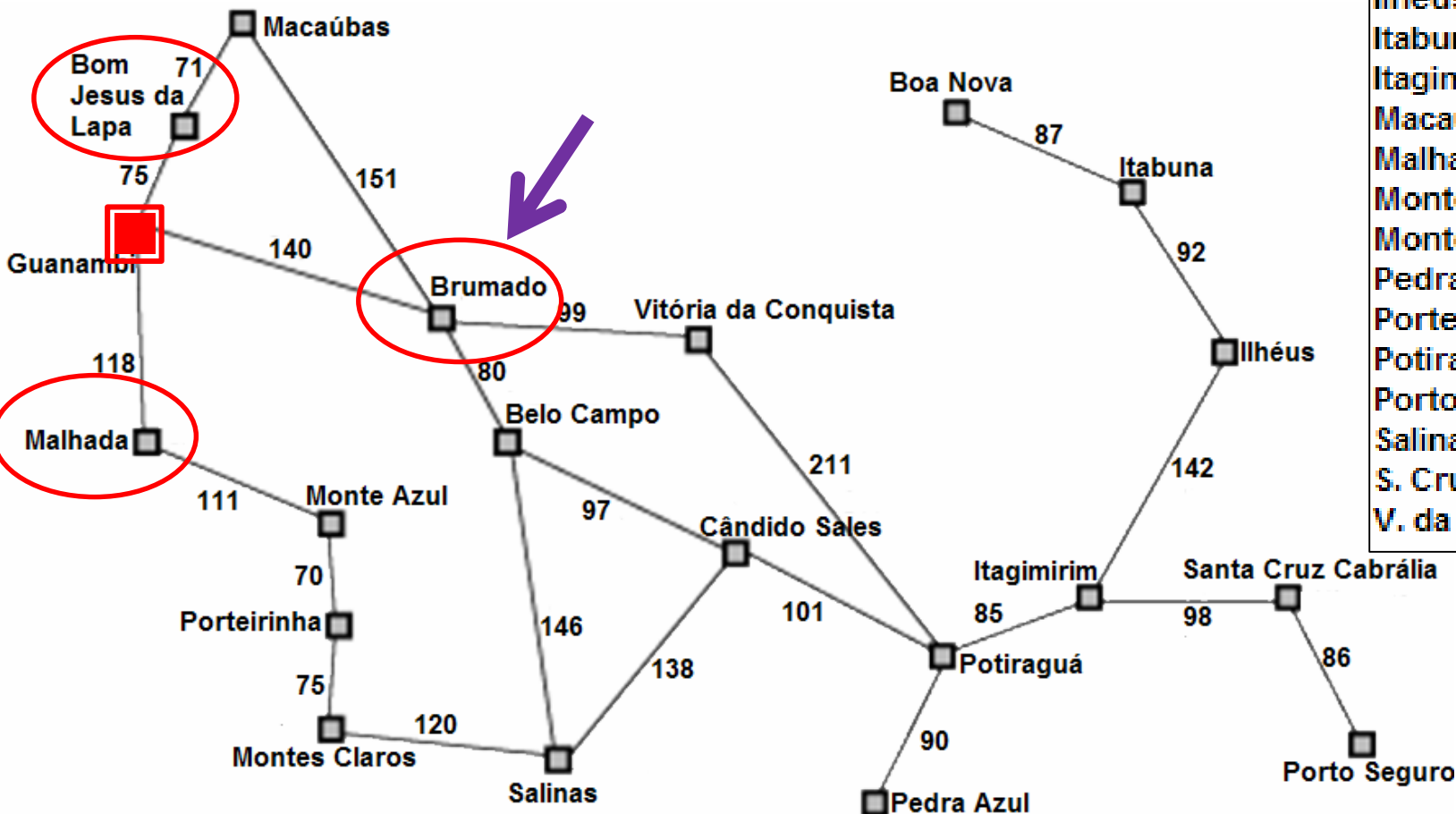


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



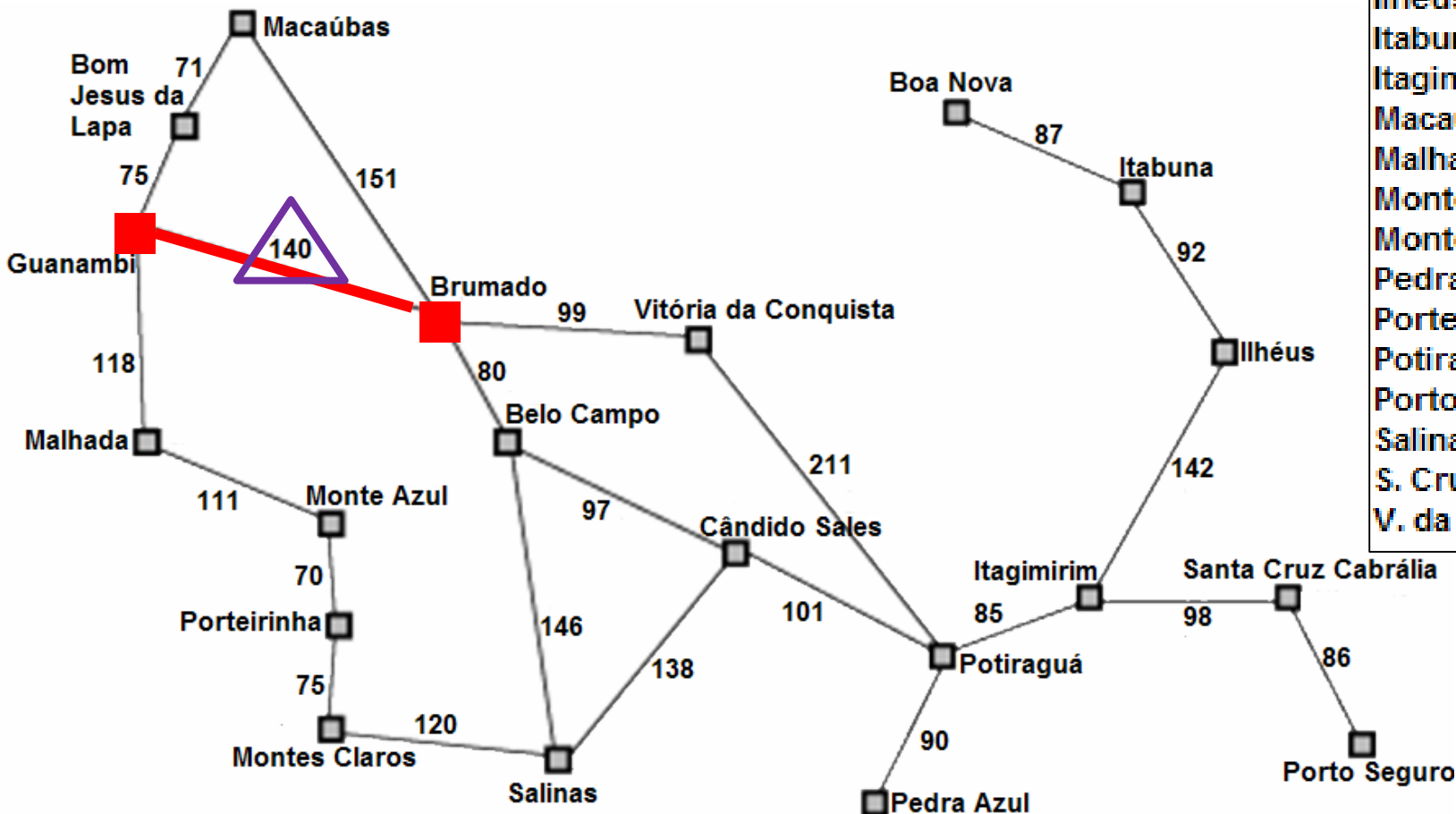
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



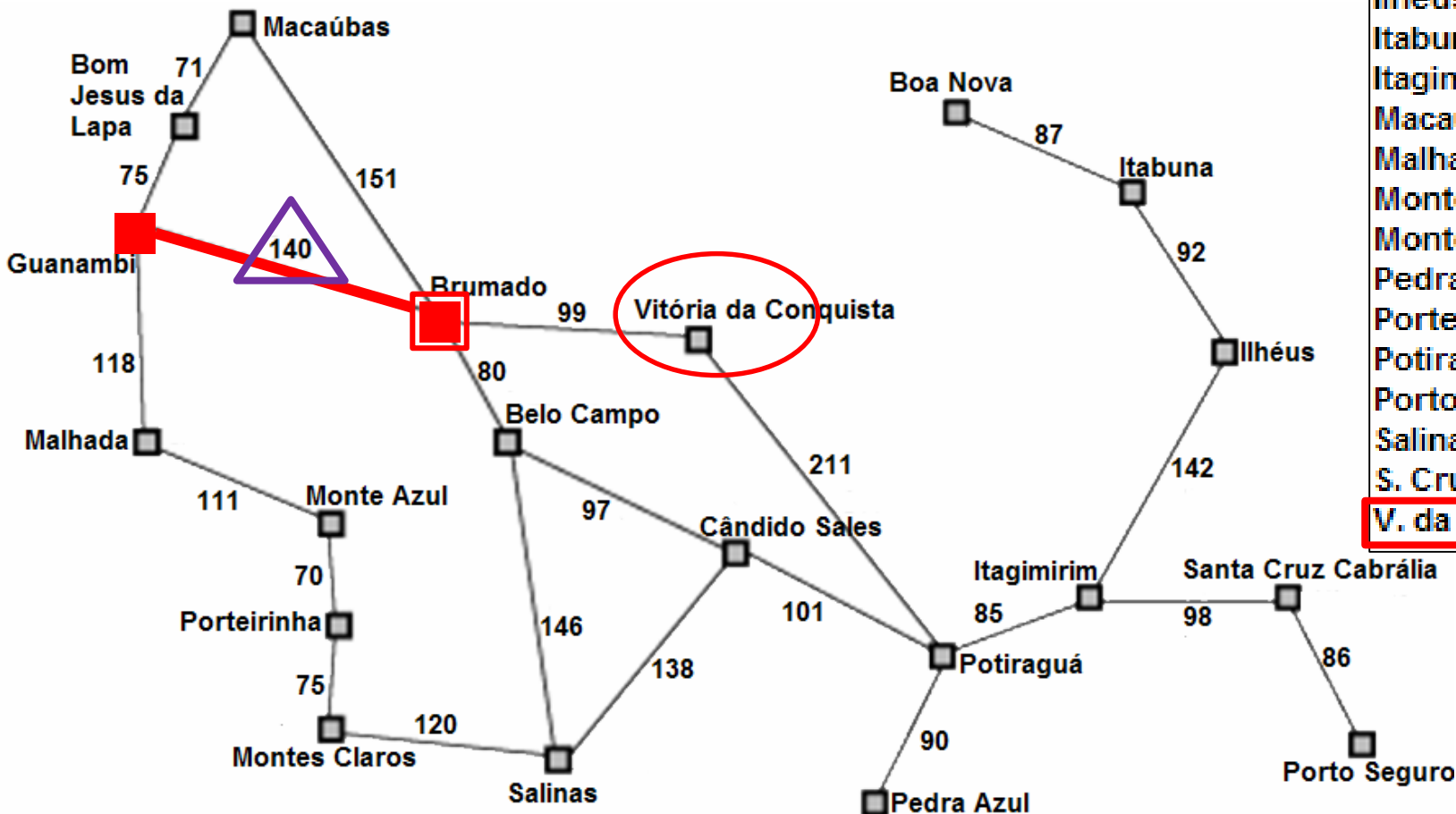
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



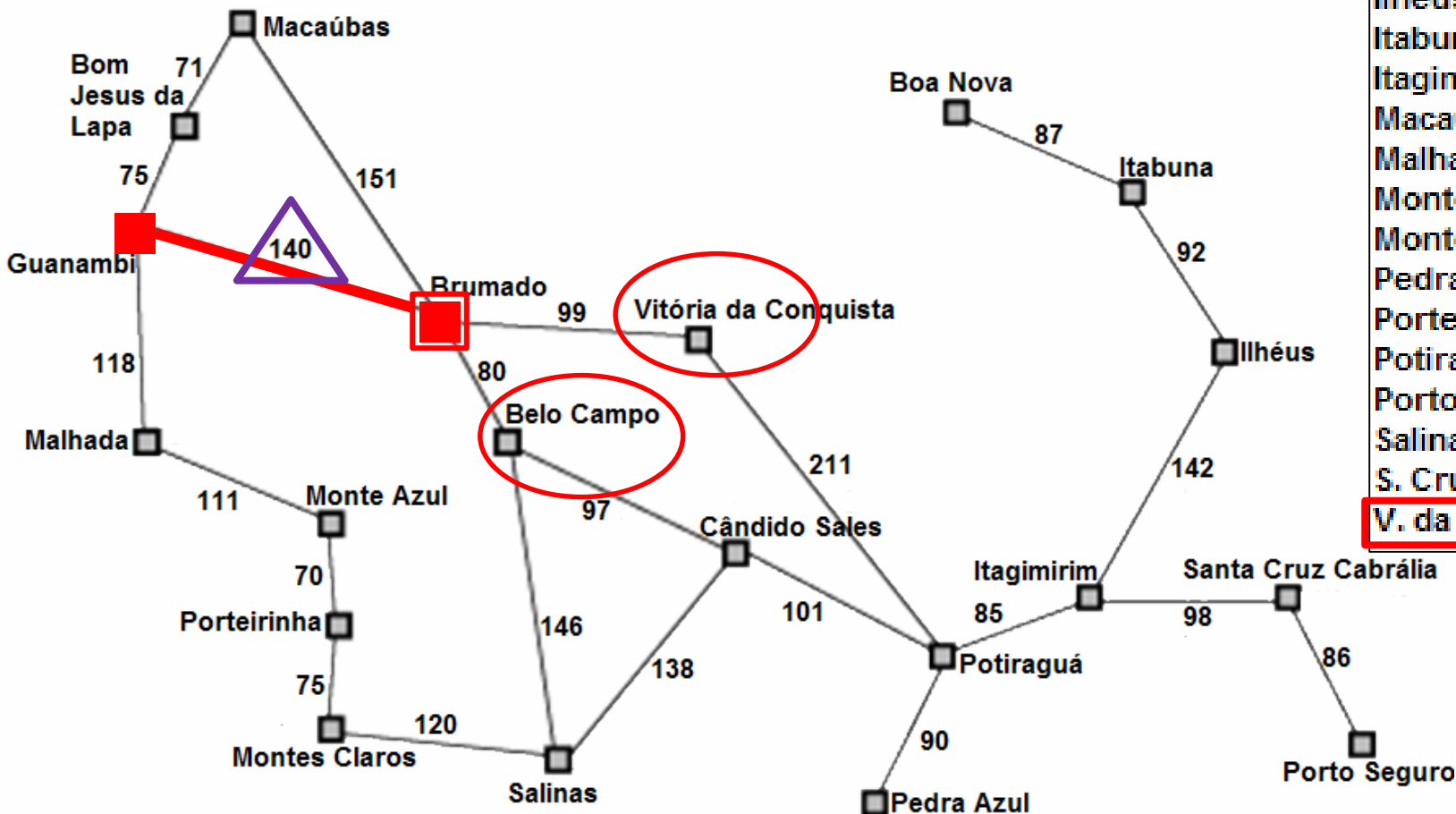
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

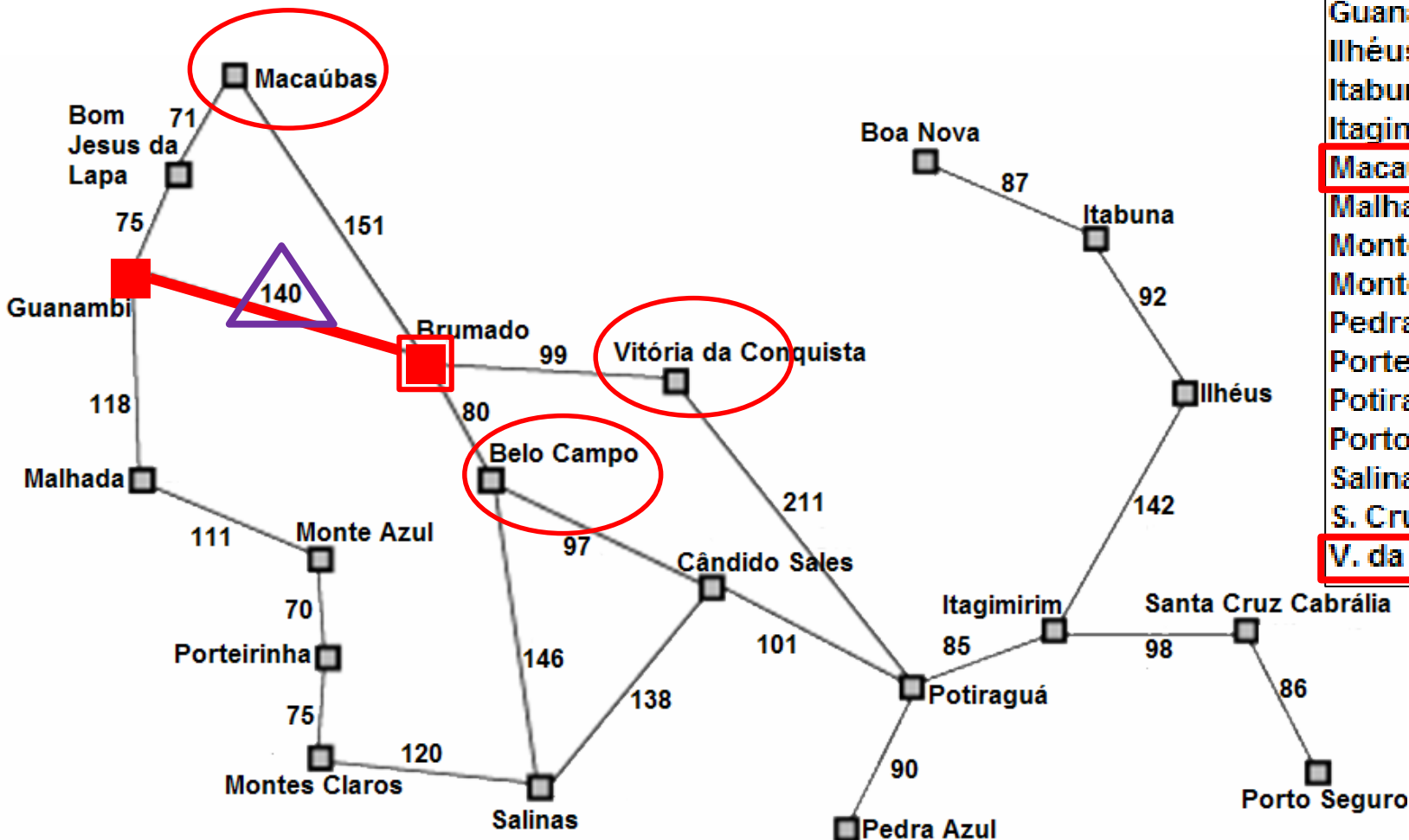


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

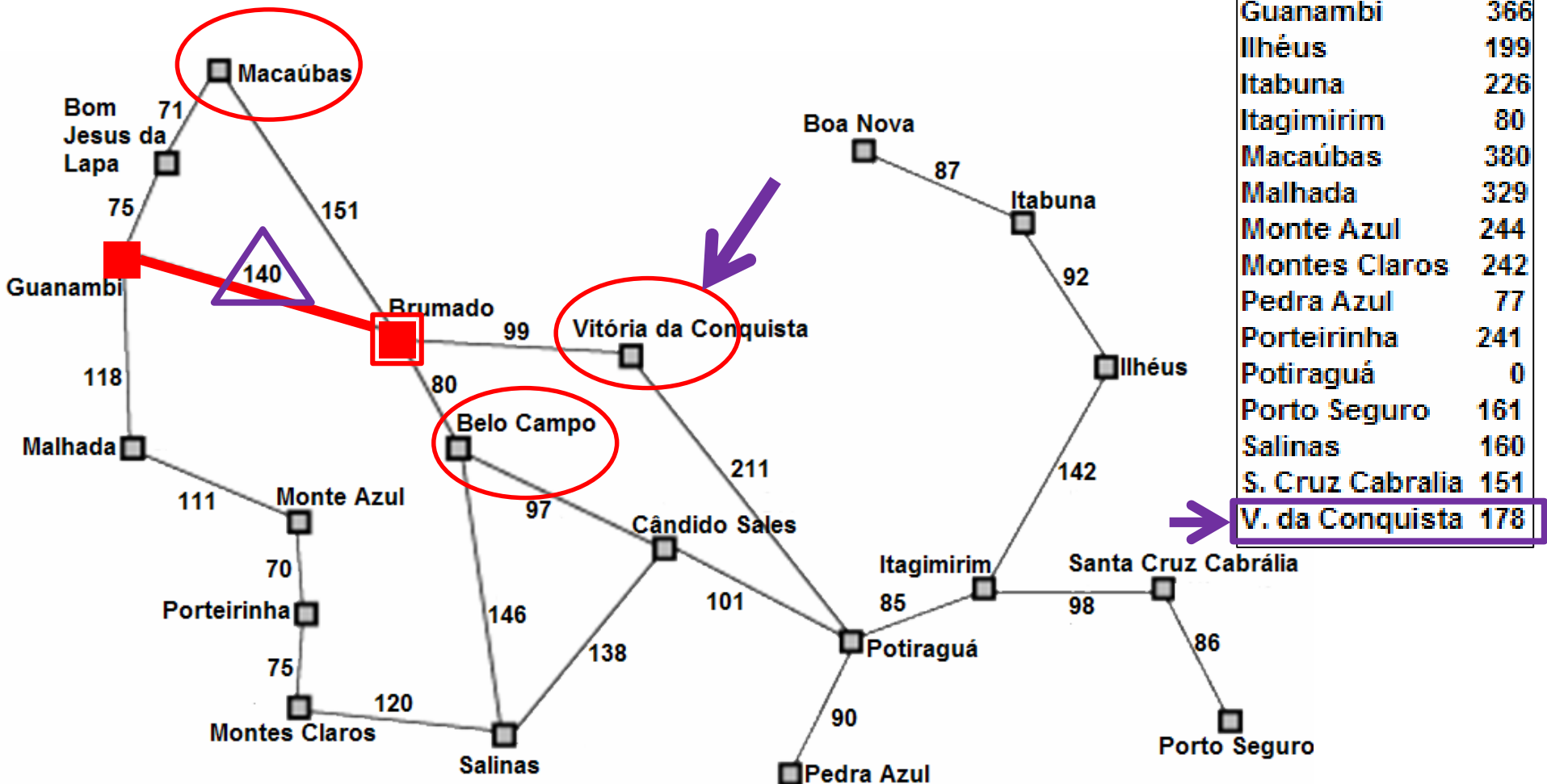


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

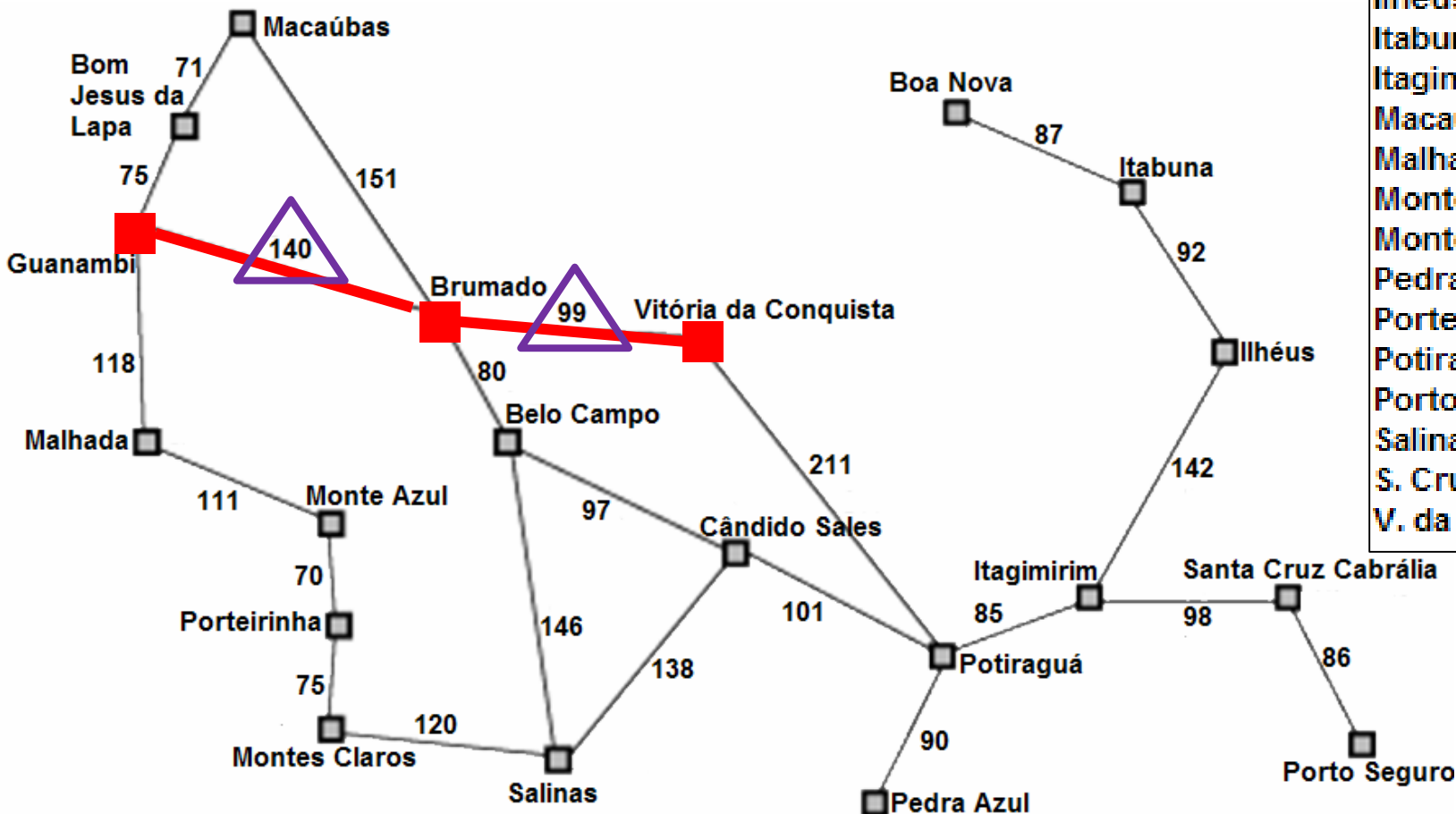
- Objetivo: de Guanambi à Potiraguá:



2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



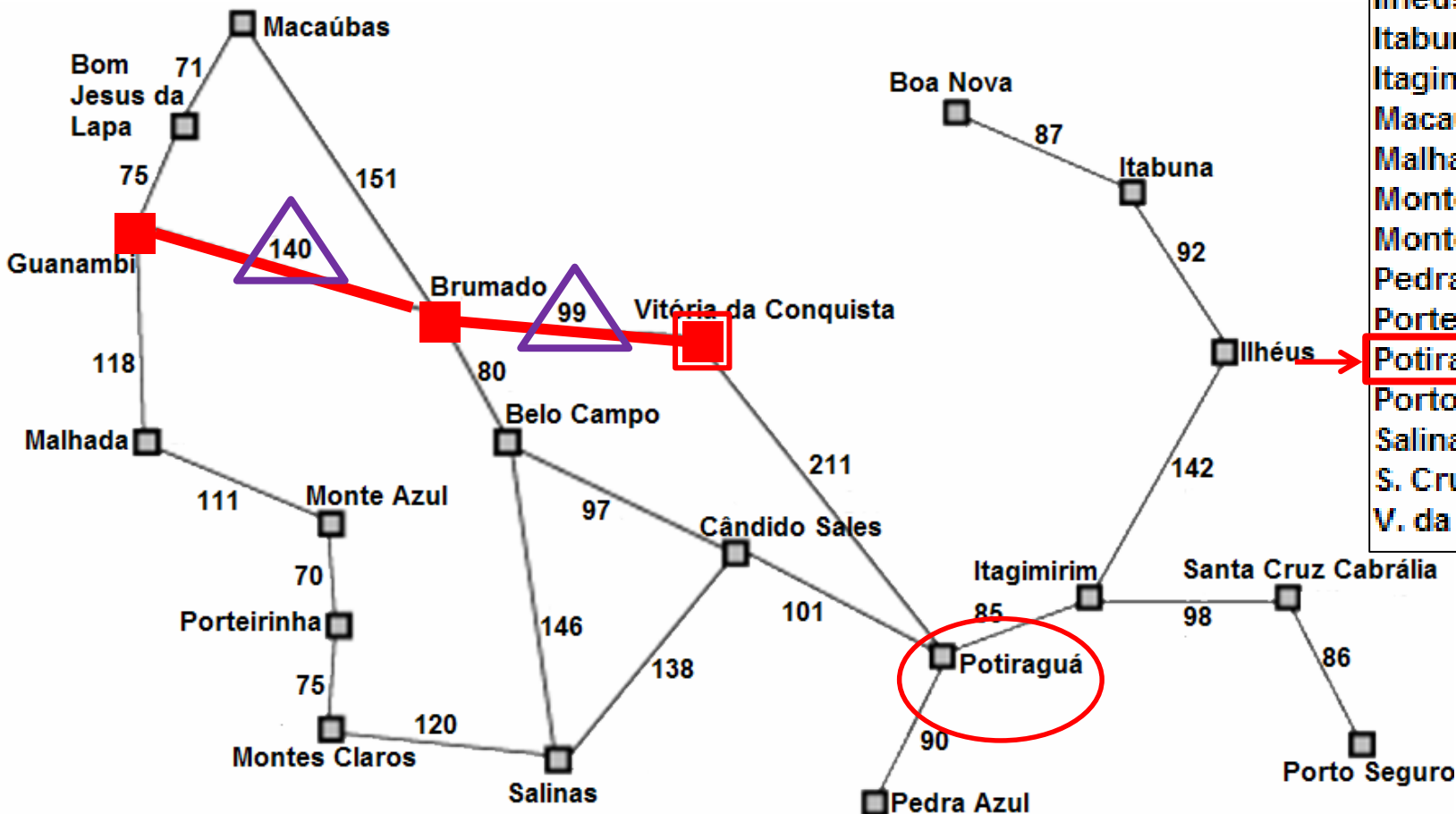
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



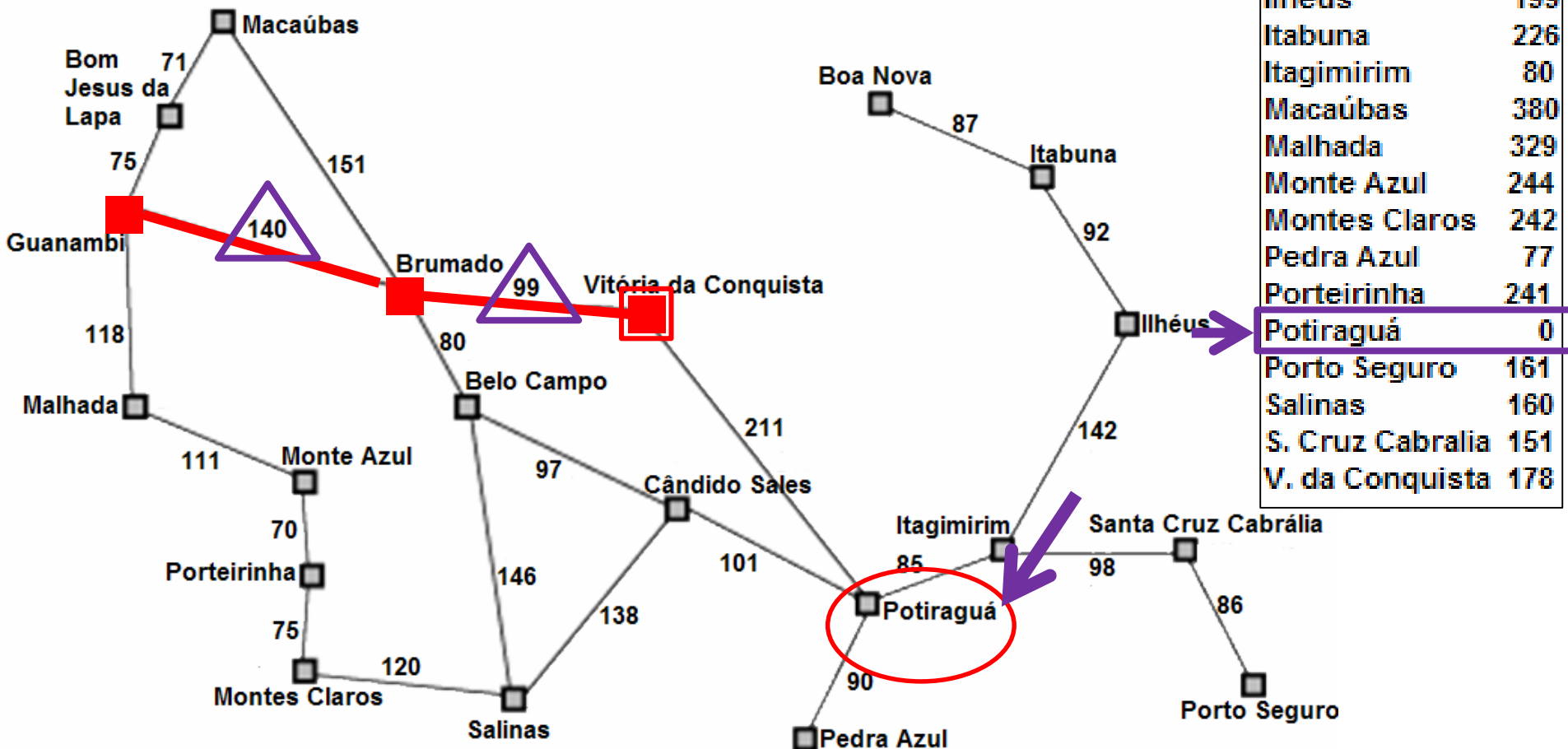
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

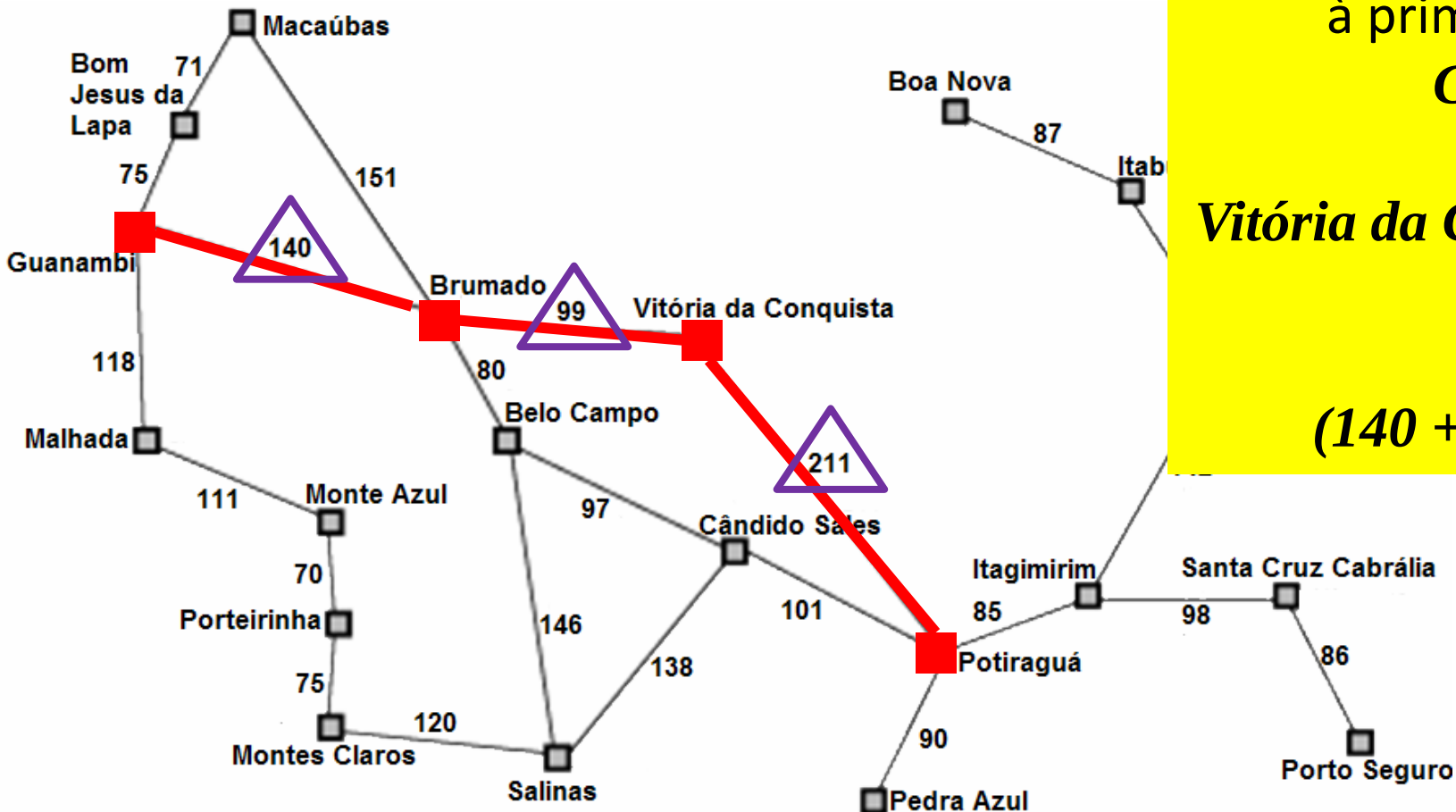
- Objetivo: de Guanambi à Potiraguá:



2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exemplo 1 : A Busca do Menor Caminho

- Objetivo: de **Guanambi** à **Potiraguá**:



A solução encontrada foi a mais econômica, à primeira vista:

***Guanambi,
Brumado,
Vitória da Conquista,
Potiraguá
= 450 Km
(140 + 99 + 211)***

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Construa a Árvore de Busca deste Exercício :

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Algoritmo: A Busca do Menor Caminho (Análise)

- Função de avaliação: escolher o nó cuja distância é a menor em linha reta do nó final.

- Sejam:

$Q(n)$ = conjunto de nós a serem pesquisados.

$h'(n)$ = a distância em linha reta de n até o nó final.

$g(n)$ = custo do nó inicial até o nó n selecionado.

$S(n)$ = o estado da busca.

H = a somatória das distâncias reais.

$f(n) = h'(n)$.

2 - Algoritmos *Best-First* (Busca Gulosa)

Algoritmo: A Busca do Menor Caminho

1. Inicialize Q com o nó de busca $S(0)$.
2. Inicialize $h'(n)$ com a distância de $S(0)$.
3. Inicialize $g(n)$ com 0 .
4. Calcule $f(n) = h'(n)$.
5. Escolha o melhor elemento de $h'(n)$ e adicione em H sua distância $g(n)$.
6. Guarde em $S(n)$ o estado n .
7. Remova todos os elementos de $Q(n)$ e de $h'(n)$.

2 - Algoritmos *Best-First* (Busca Gulosa)

Algoritmo: A Busca do Menor Caminho

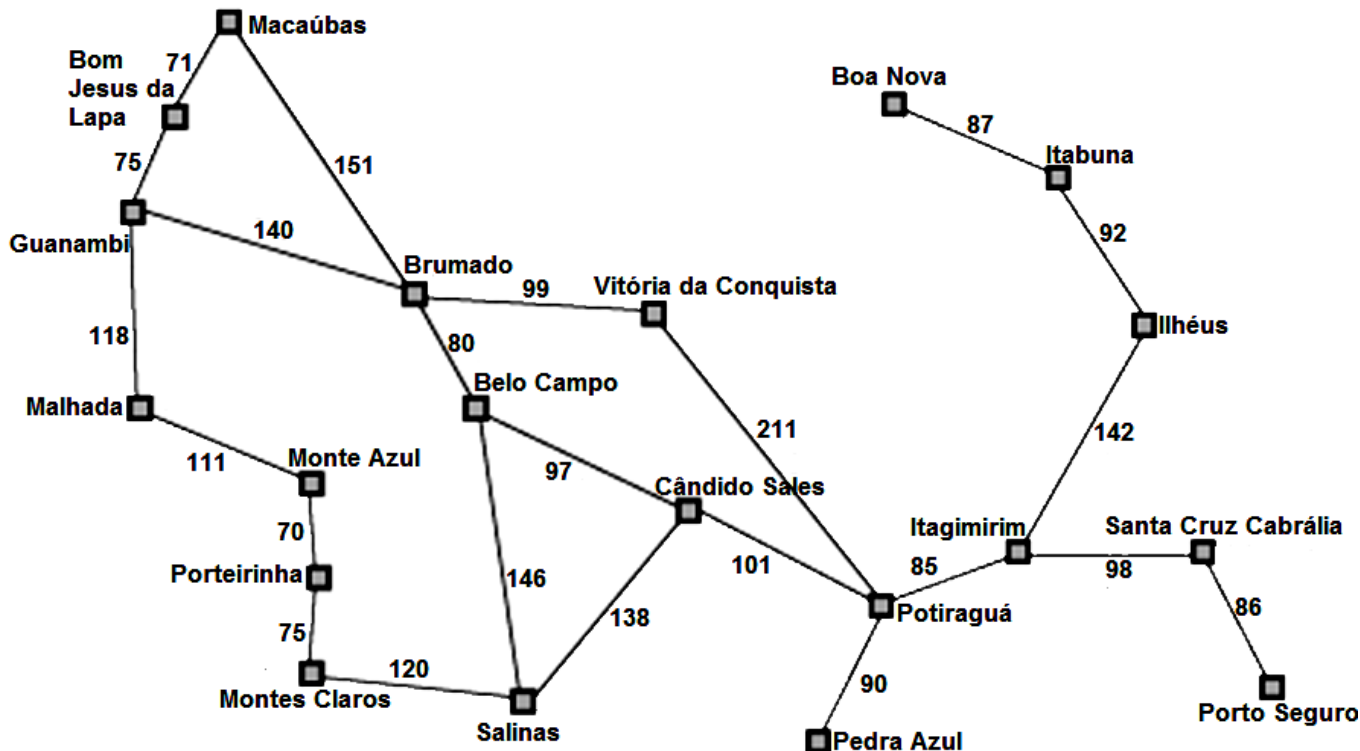
8. Verifique se $f(n) = 0$:
 - então, retorne H e $S(n)$ e encerre;
 - senão, prossiga.
9. Encontre os descendentes do estado (n) e os adicione em $Q(n)$.
10. Crie todas as extensões de n para cada descendente encontrado, as adicione em $h'(n)$ e suas distâncias do nó pai em $g(n)$.
11. Retorne ao passo 4.

2 - Algoritmos *Best-First* (Busca Gulosa)

Exercício 2.1:

O mapa mostra distâncias entre cidades da *Bahia*. A tabela mostra as distâncias em linha reta de cada cidade até *Vitória da Conquista*. Utilizando o algoritmo **Best First**, sendo $f(n) = h'(n)$, faça:

a) A árvore de busca de **Porteirinha** até **Vitória da Conquista**:



Distância em linha reta até Vitória da Conquista	
Belo Campo	82
Boa Nova	130
Bom Jesus da Lapa	218
Brumado	87
Cândido Sales	85
Guanambi	240
Ilhéus	217
Itabuna	185
Itagimirim	195
Macaúbas	228
Malhada	219
Monte Azul	163
Montes Claros	234
Pedra Azul	200
Porteirinha	221
Porto Seguro	327
Potiraguá	178
Salinas	174
Santa Cruz Cabralia	272
Vitória da Conquista	0

2 - Algoritmos *Best-First* (*Busca Gulosa*)

Exercício 2.1:

- b) Mostre o cálculo da distância total percorrida:

- c) O resultado obtido produziu a solução ótima? Justifique:

3

Algoritmos *Best-First* (A^*)

3 - Algoritmos *Best-First* (A^*)

Características da A^* Search:

Combina a busca pelo menor custo e a busca pelo melhor primeiro. Considera tanto o custo do nó selecionado até o nó objetivo, quanto o custo para atingir o nó selecionado.

- Foi chamado de algoritmo A . Com heurística apropriada atinge-se um comportamento ótimo, e passa-se a chamá-lo por A^* .
- Sua aplicação vai desde aplicativos para encontrar rotas de deslocamento entre localidades até resolução de jogos, onde é muito utilizado.

3 - Algoritmos *Best-First* (A^*)

Algoritmo de A^* Search:

- Sejam:

$g(n)$ = função do custo do nó inicial até o nó n selecionado.

$h'(n)$ = função que estima o custo para se chegar do nó n até o nó final. Esta função é a responsável pela parte heurística do processamento.

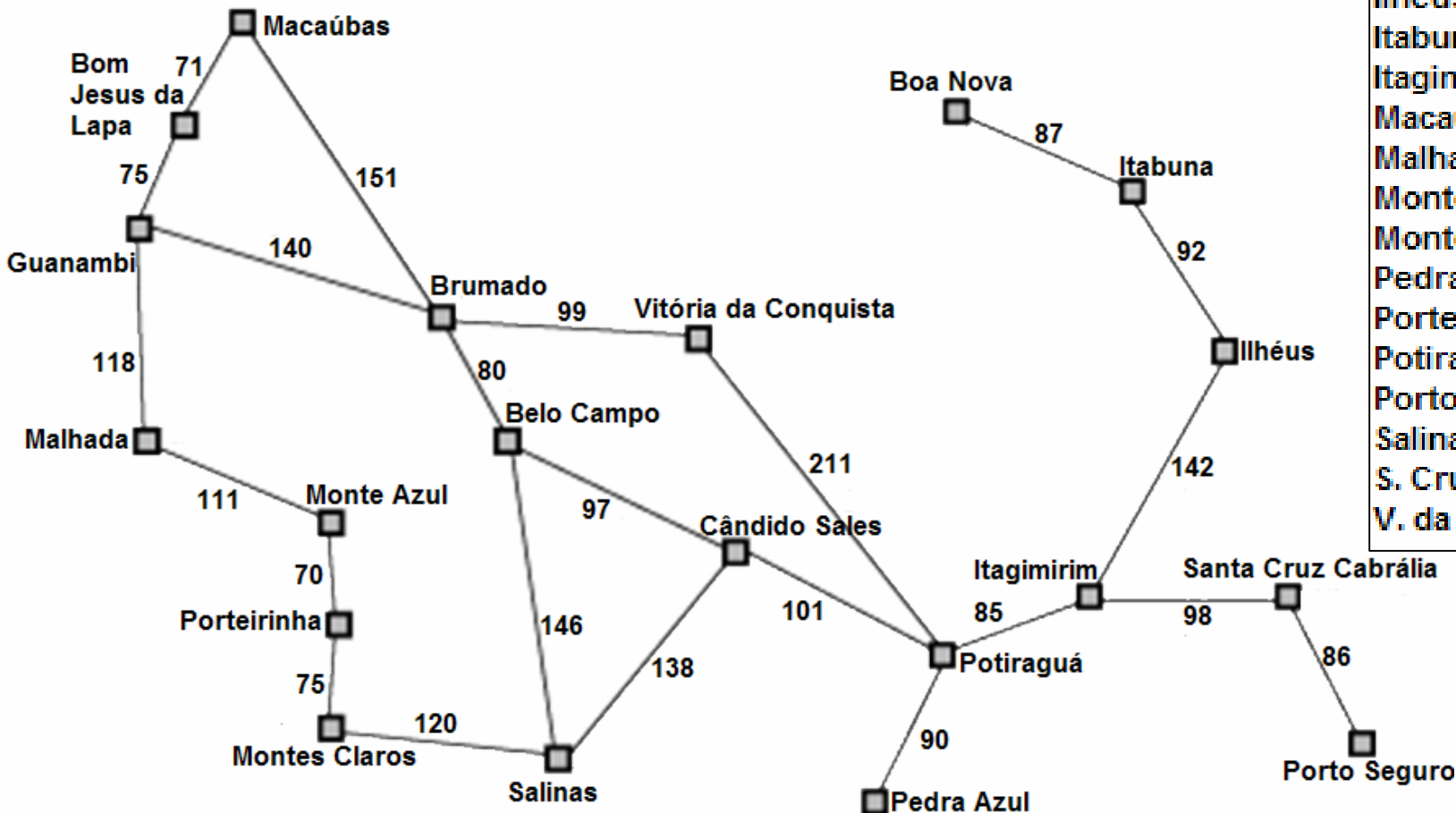
Função de avaliação: $f(n) = g(n) + h'(n)$

$f(n)$ é o custo estimado da solução de custo mais baixo, passando por n .

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



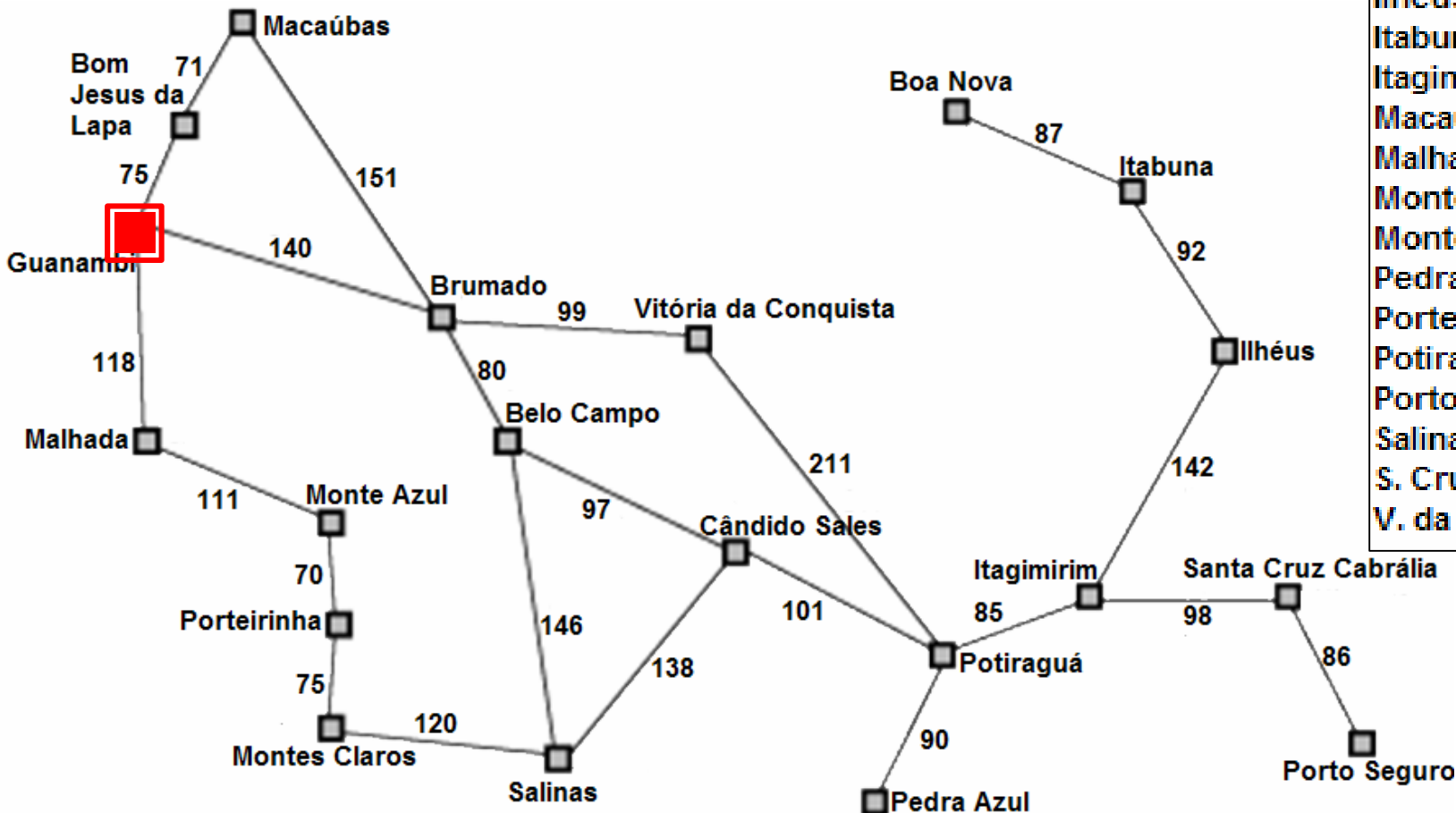
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



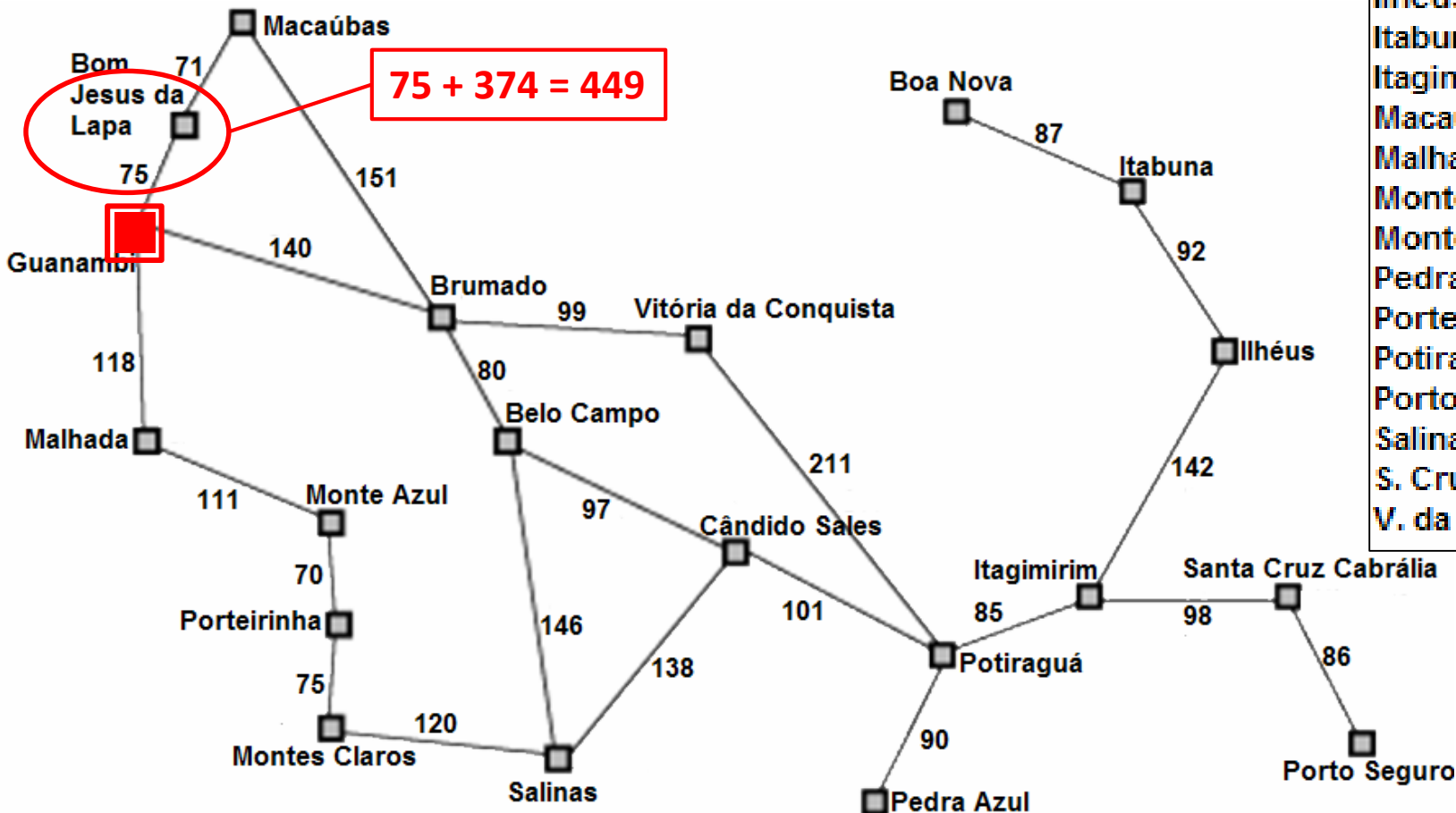
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



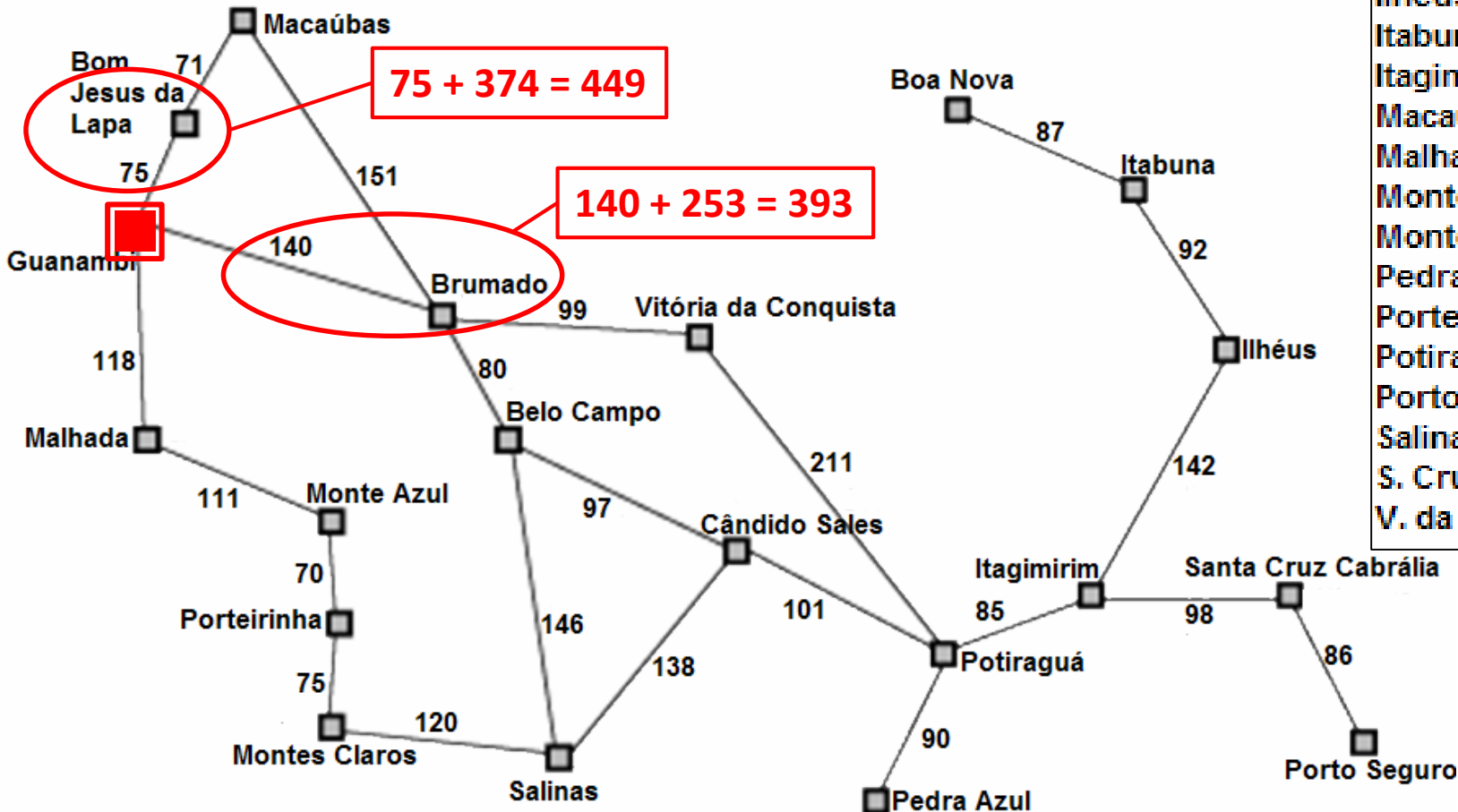
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



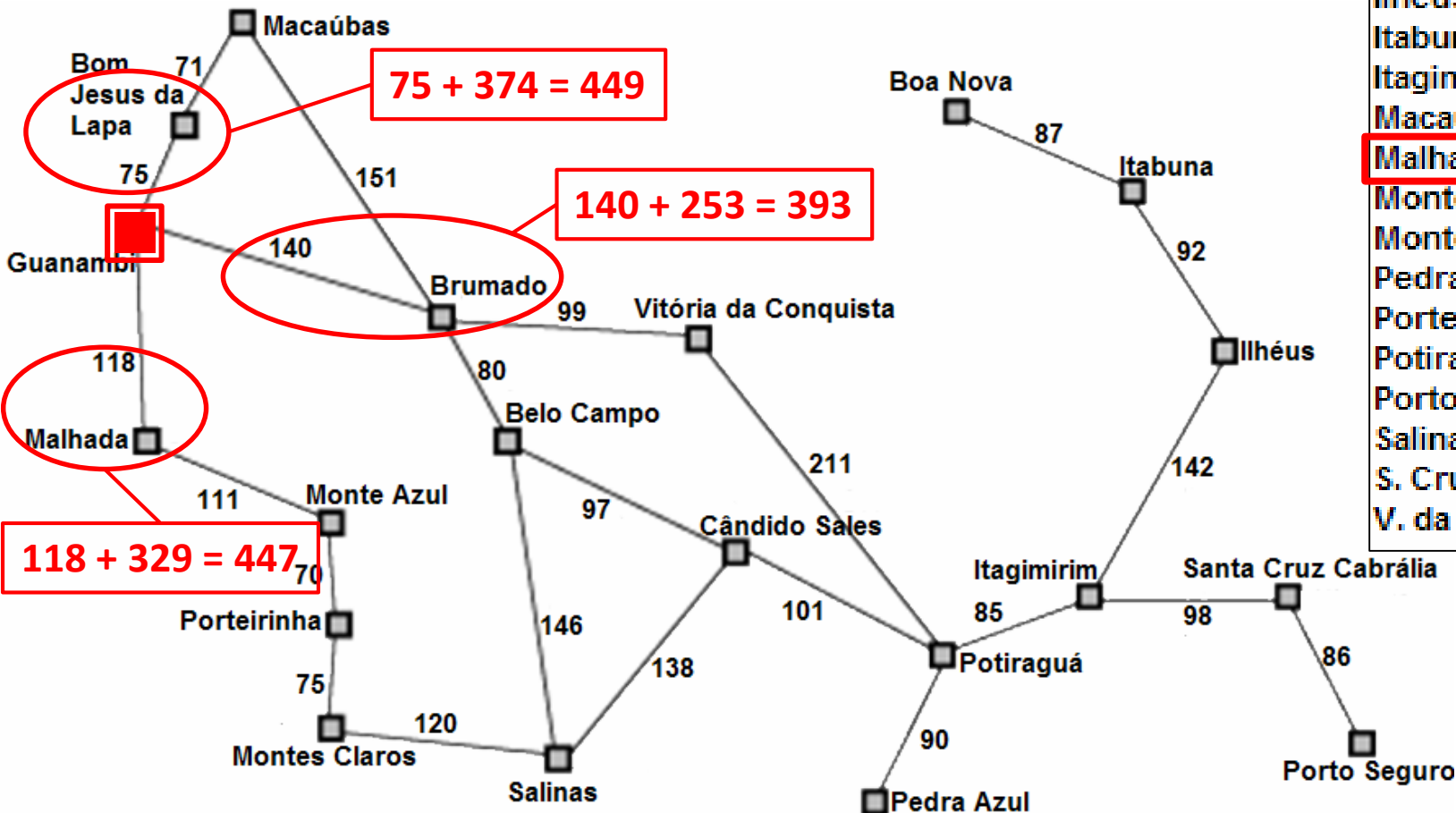
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



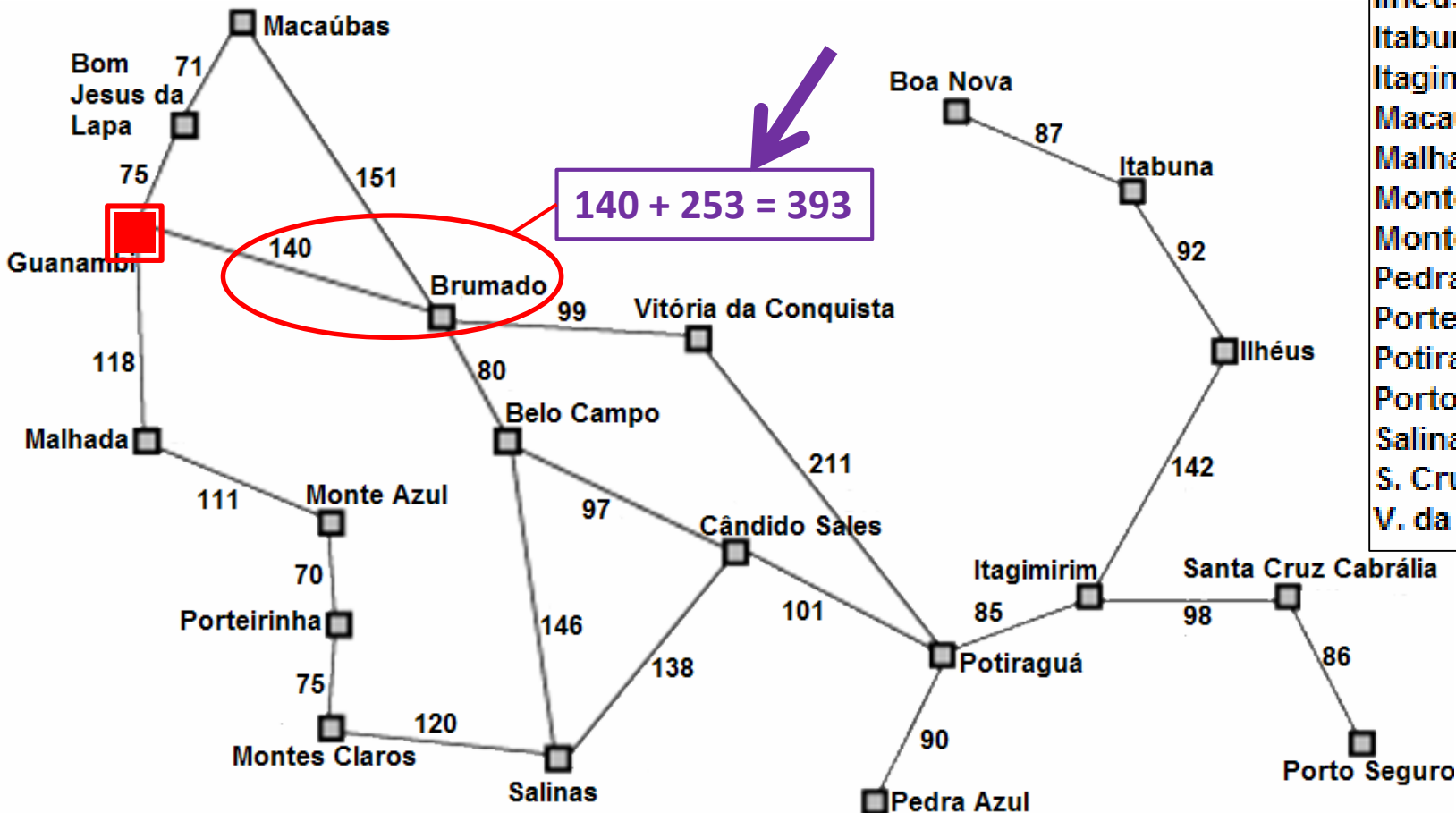
Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de **Guanambi** à **Potiraguá**:

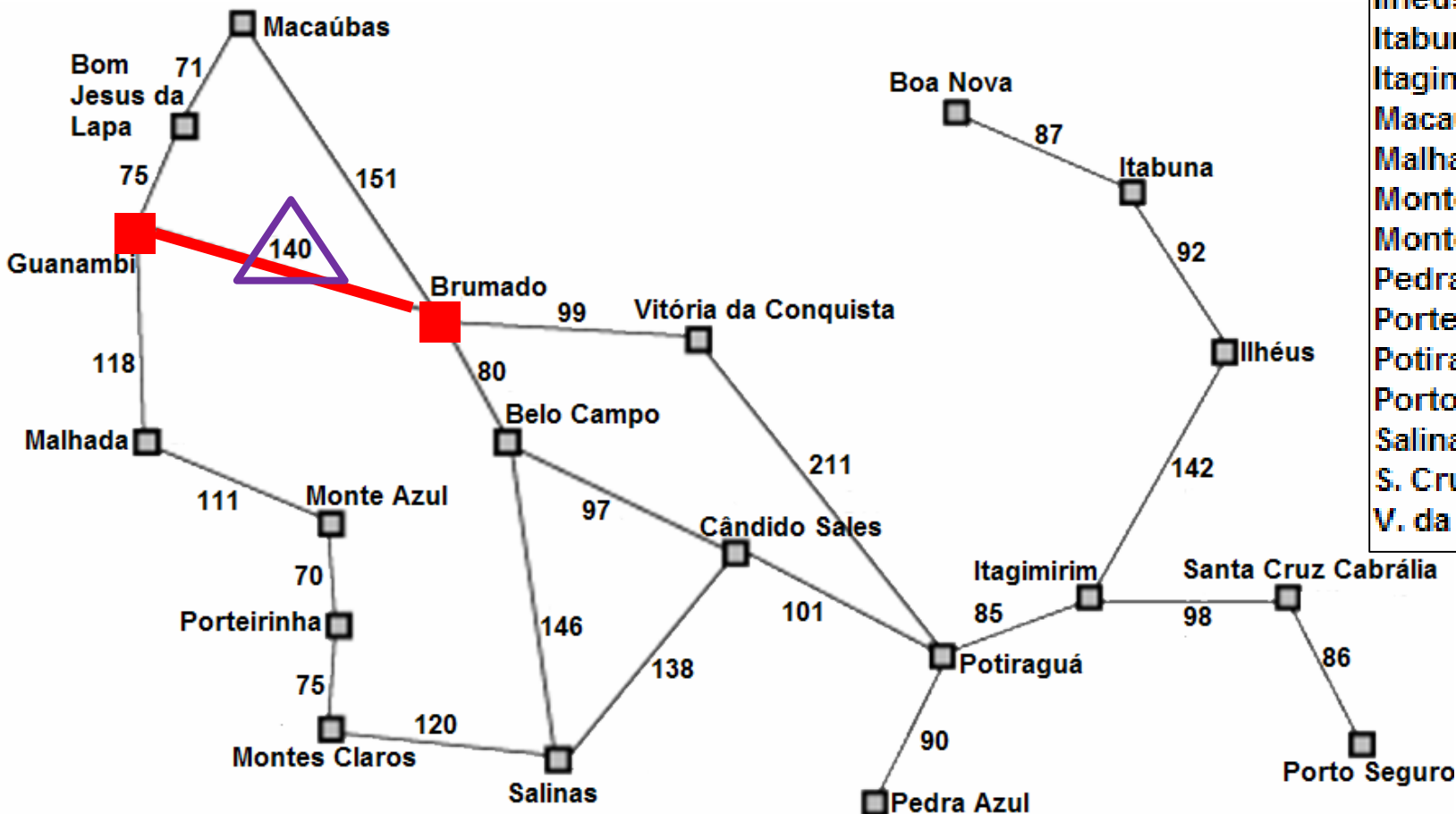
Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



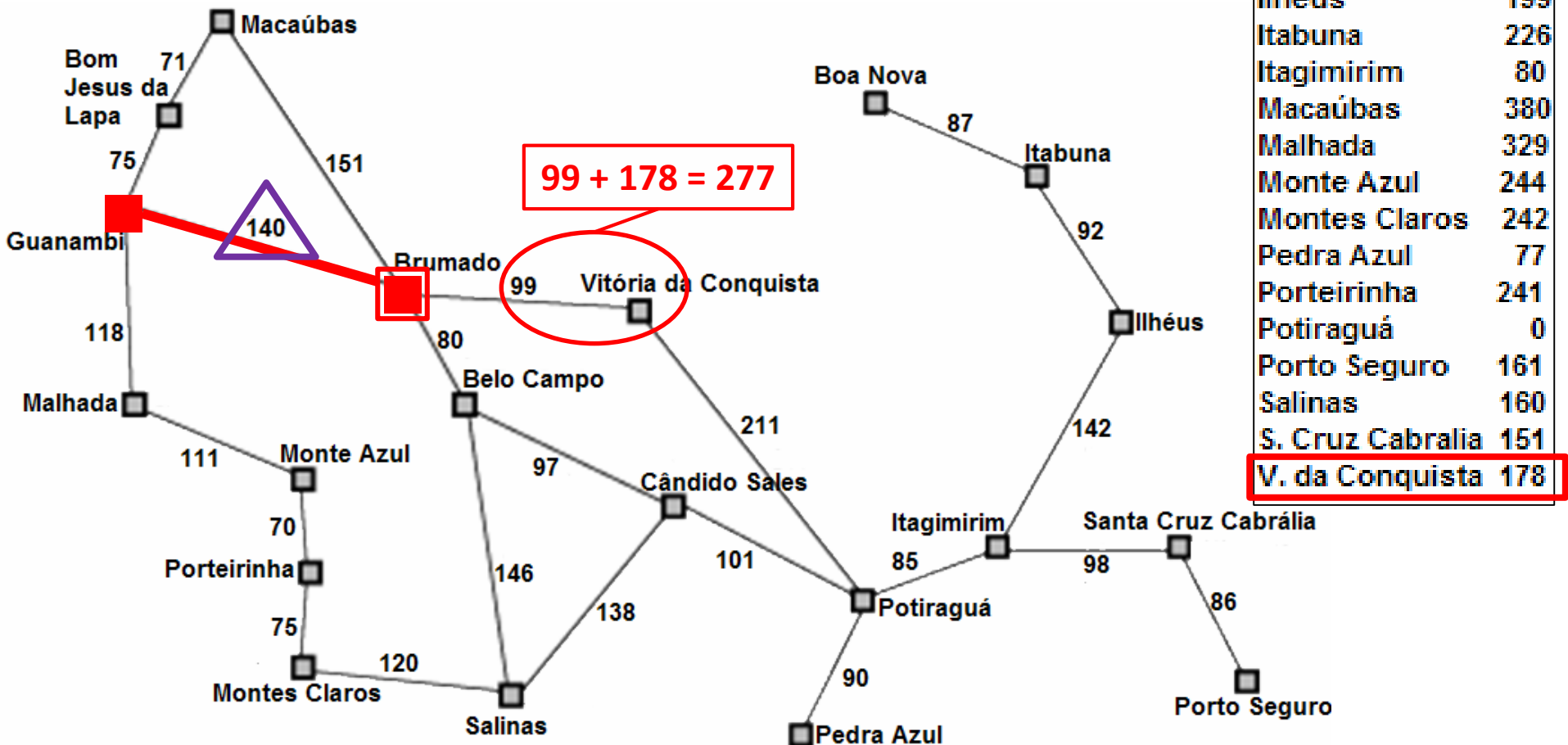
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

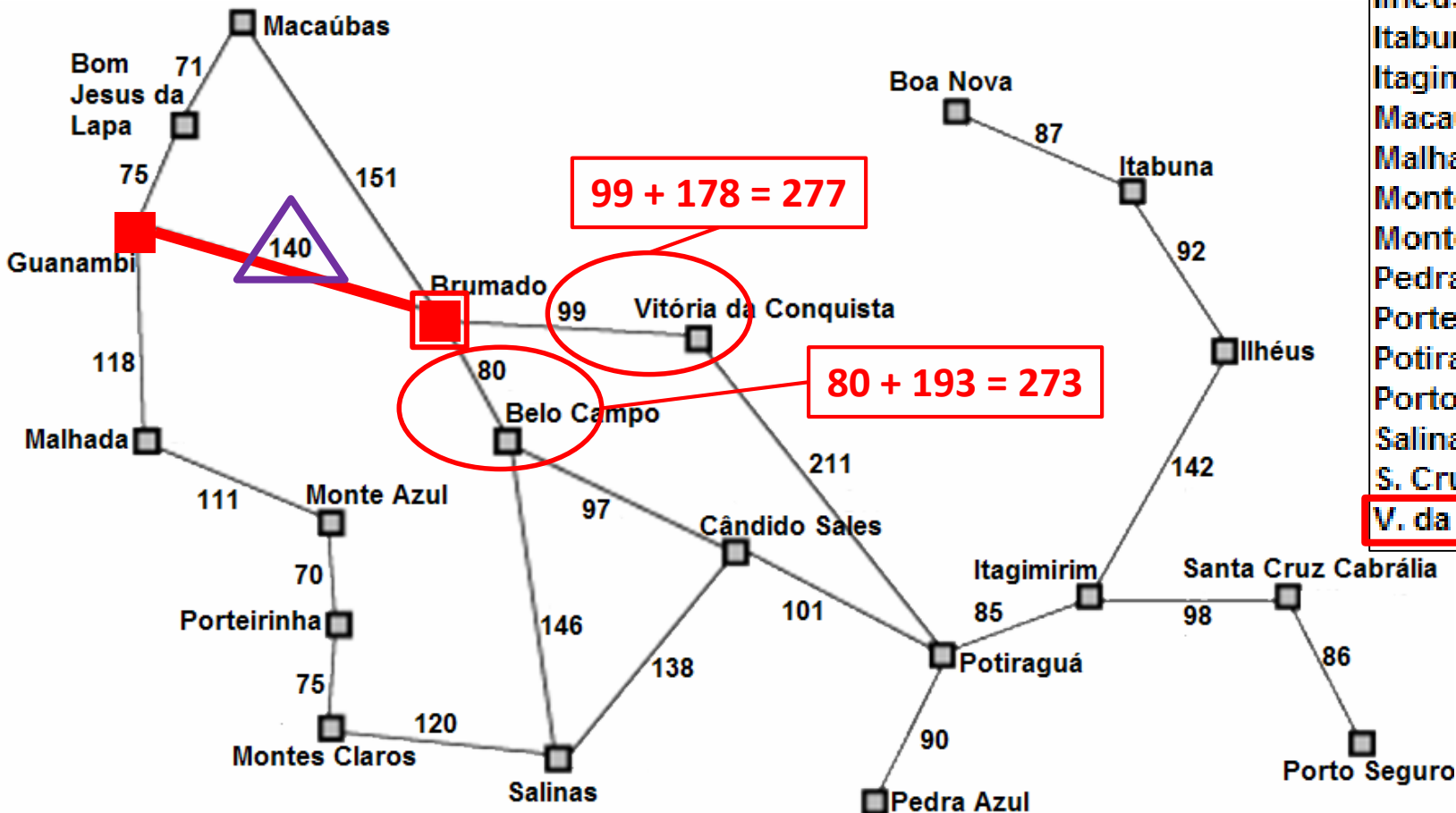
- Objetivo: de Guanambi à Potiraguá:



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



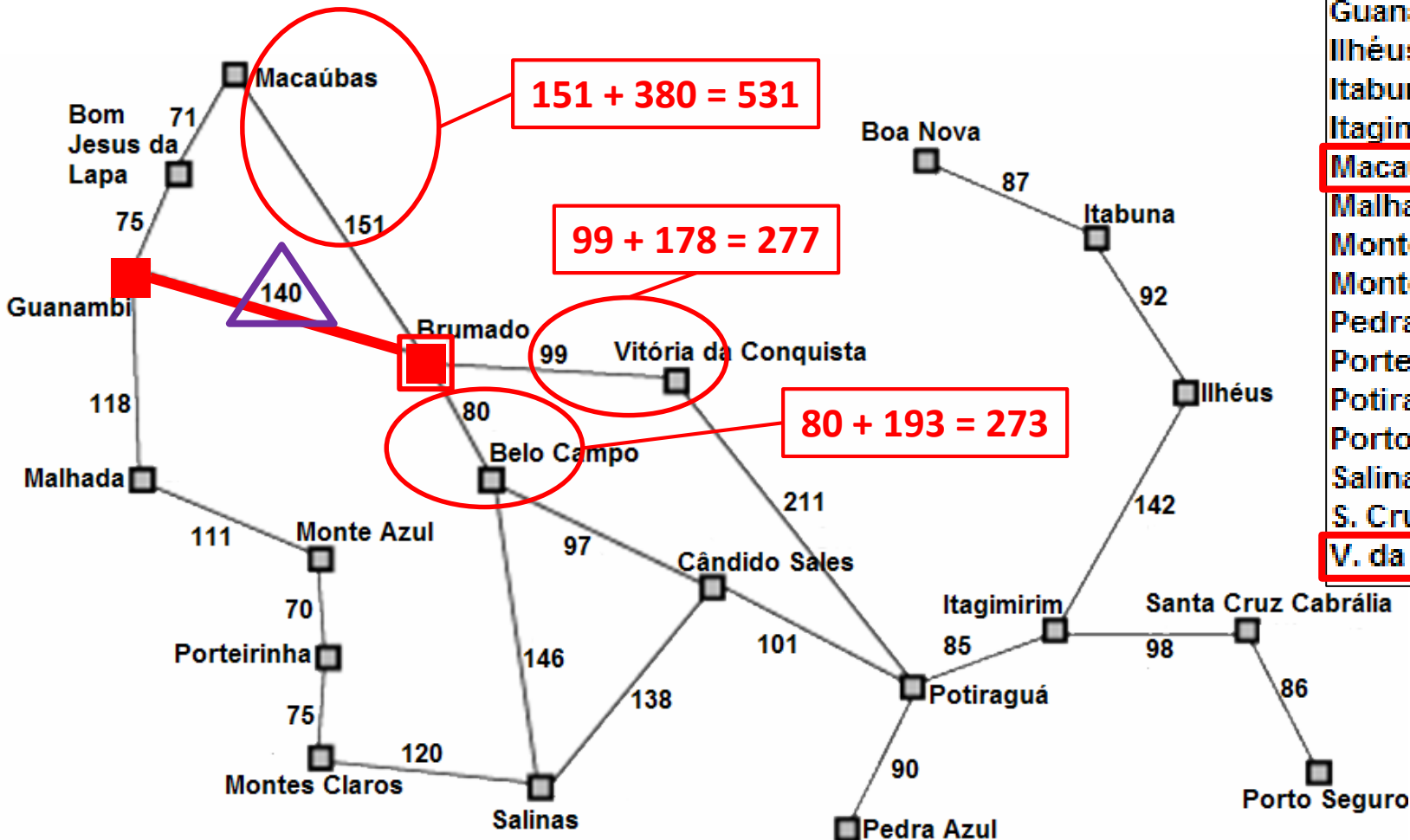
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

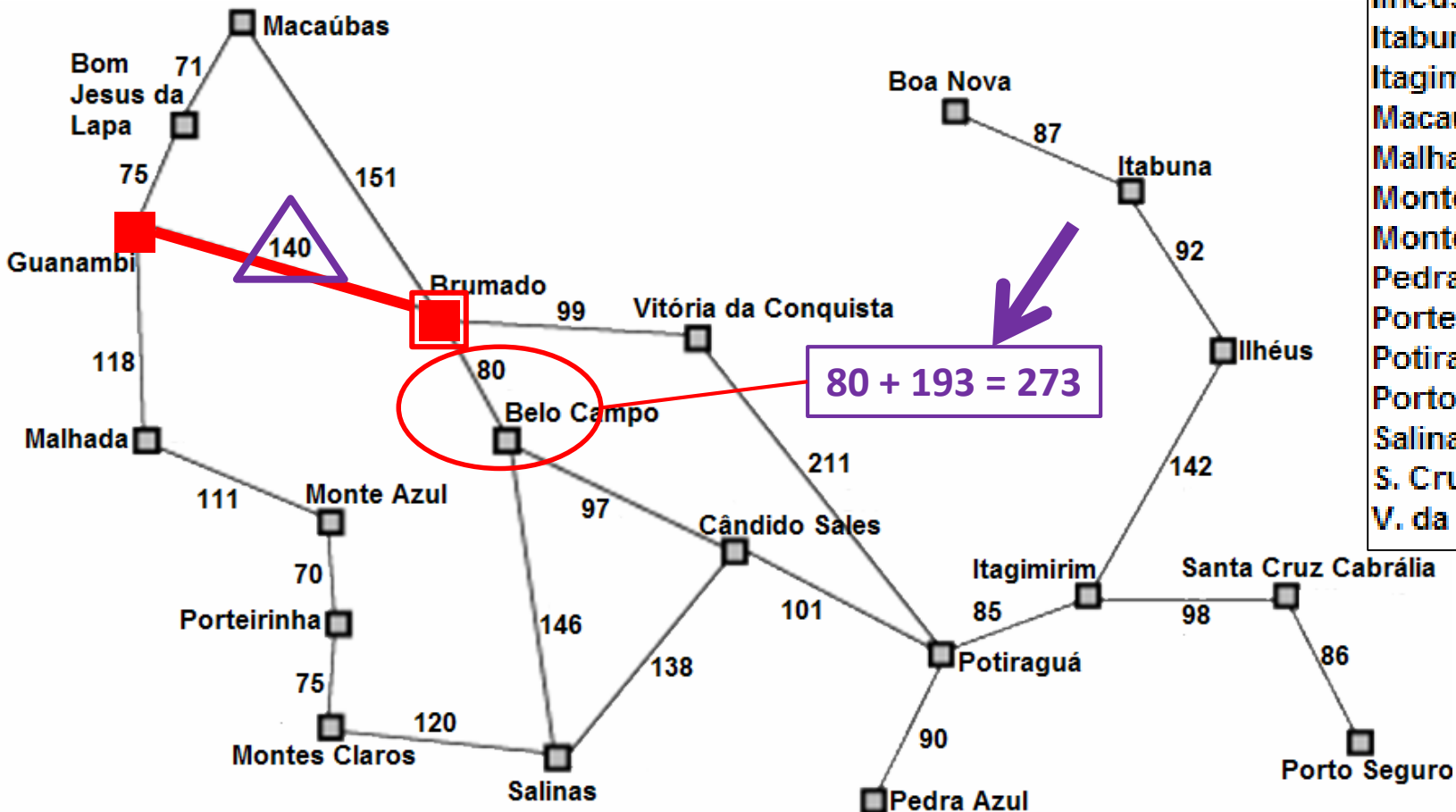


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



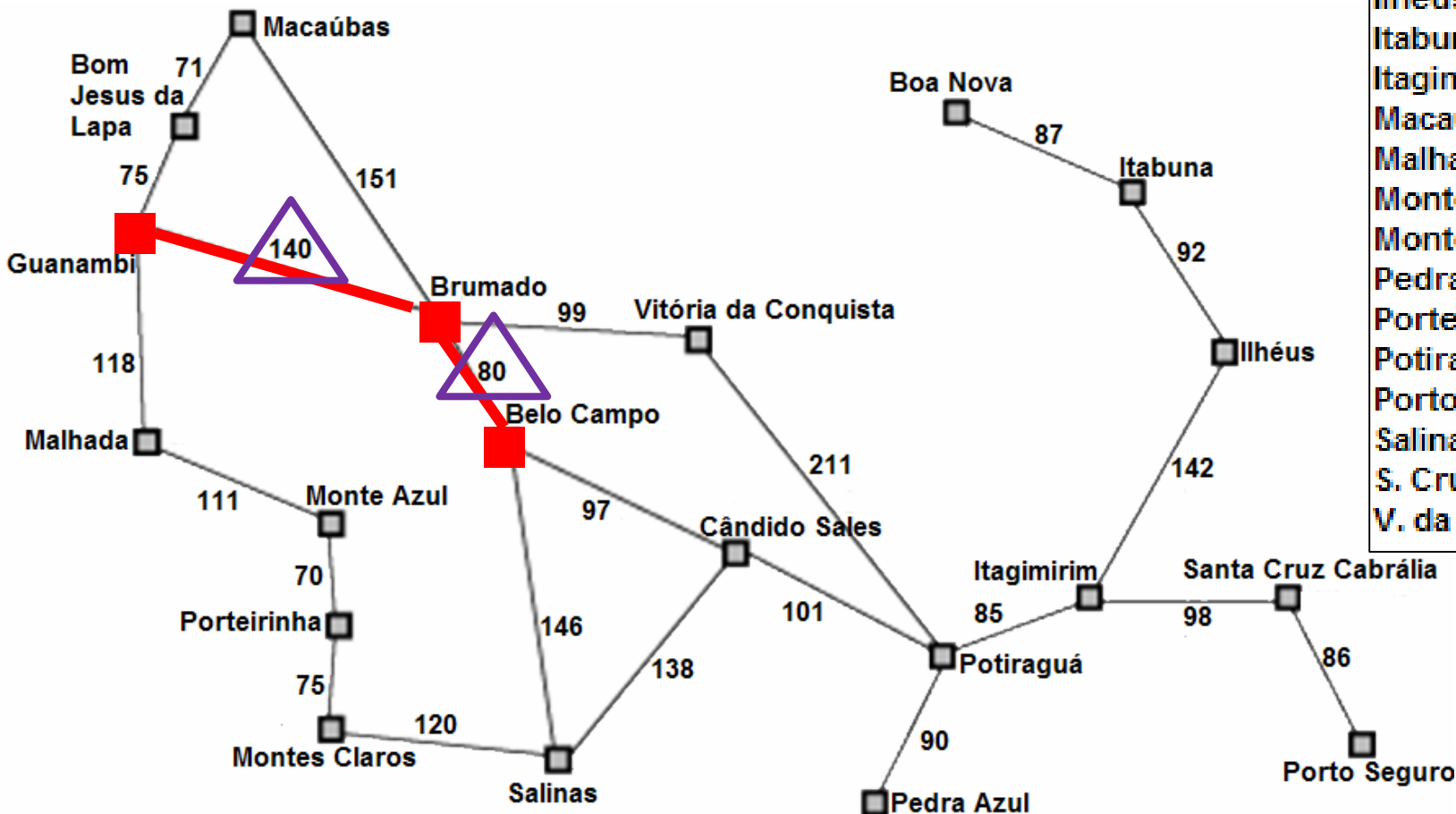
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

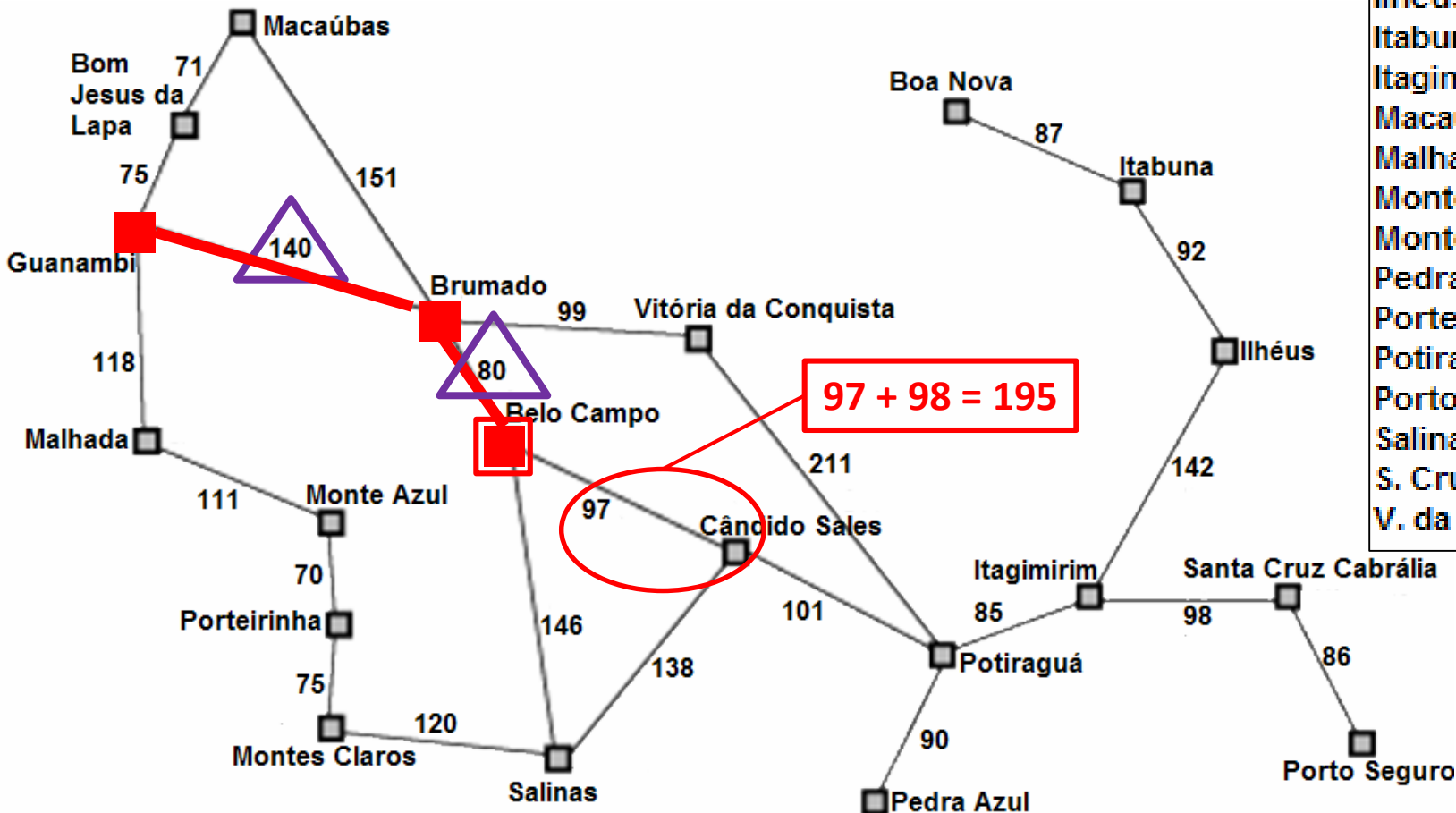


Distância em linha reta para Potiraguá	
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



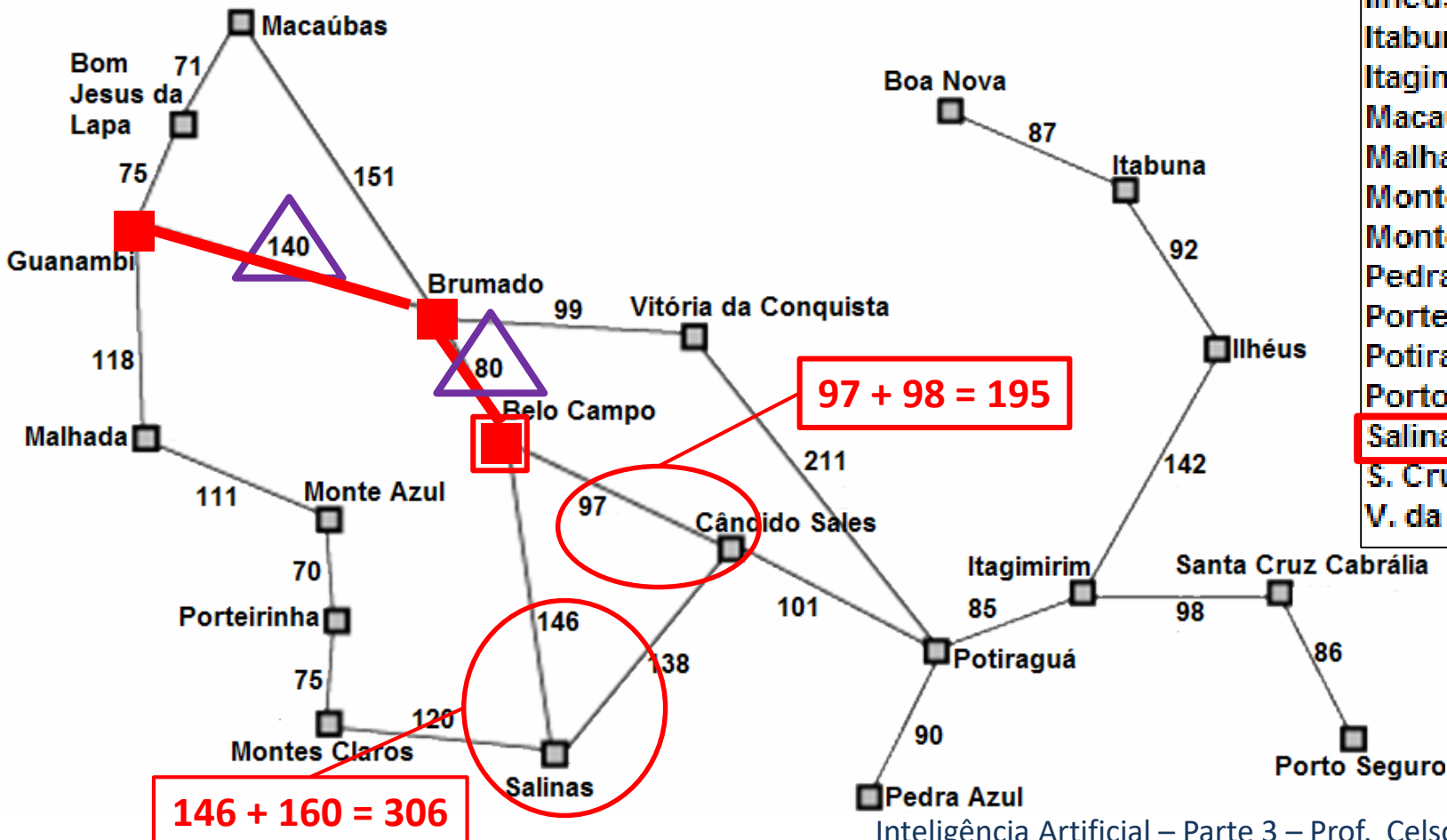
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



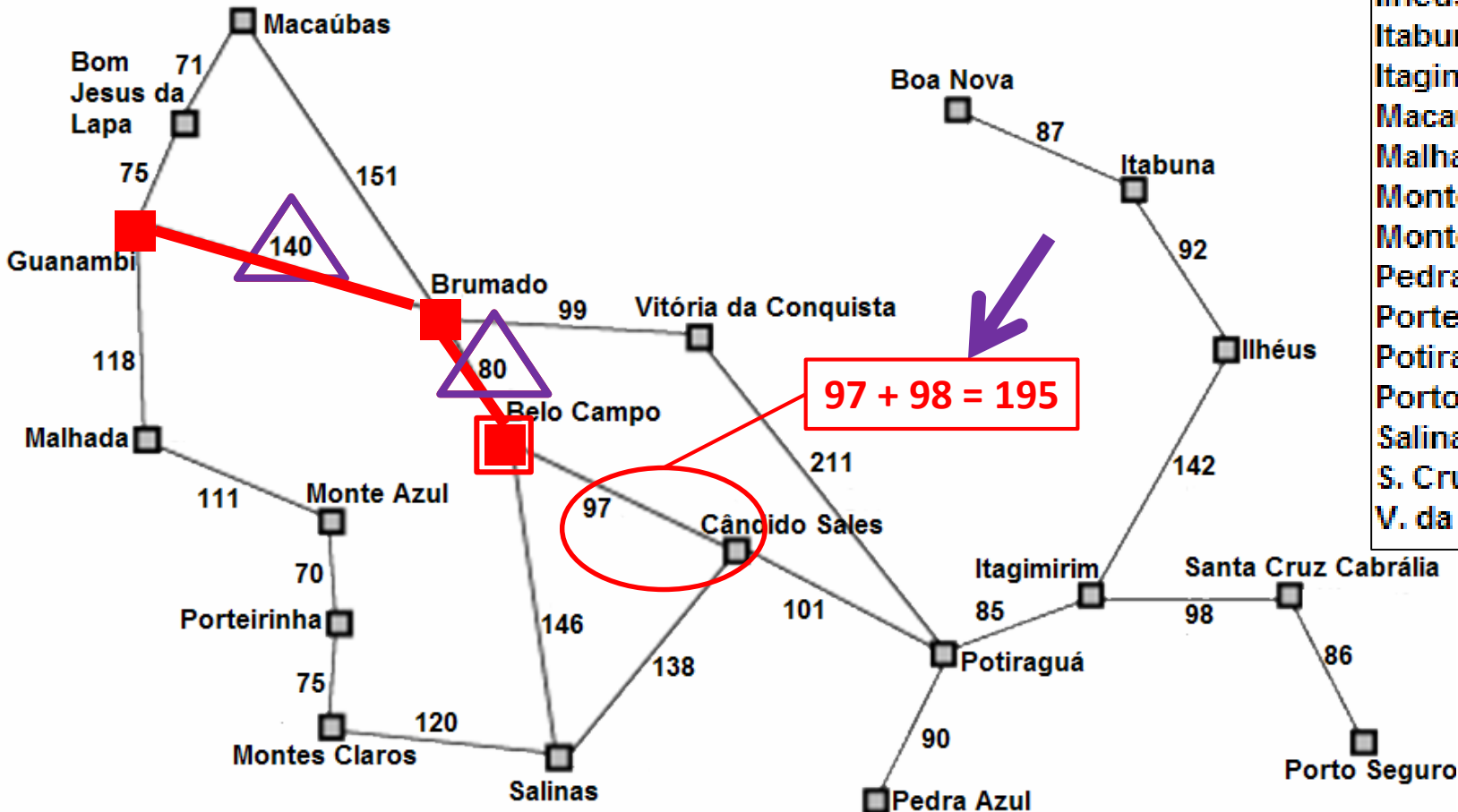
3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:

Distância em
linha reta para
Potiraguá

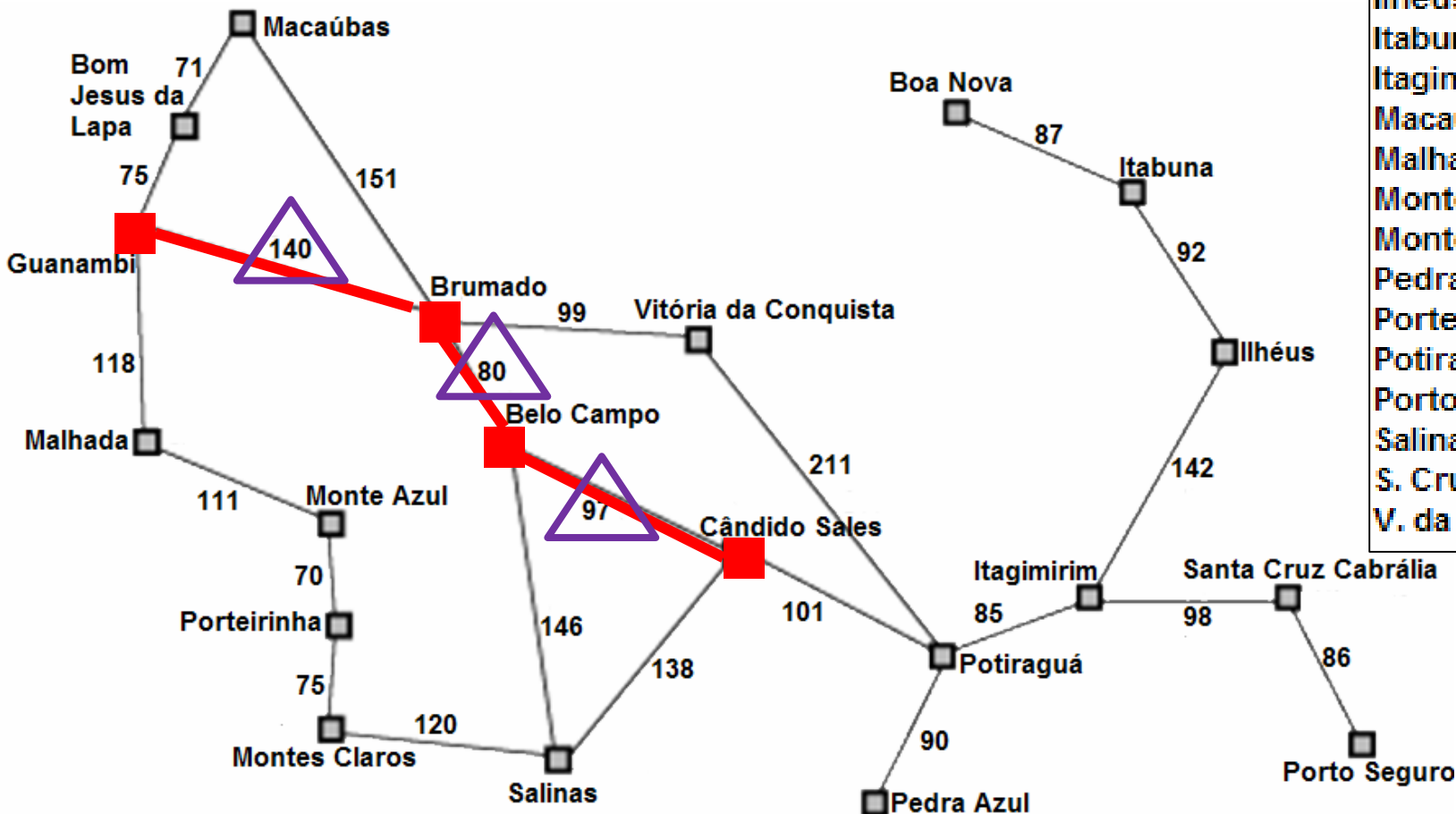
Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
→ Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de Guanambi à Potiraguá:



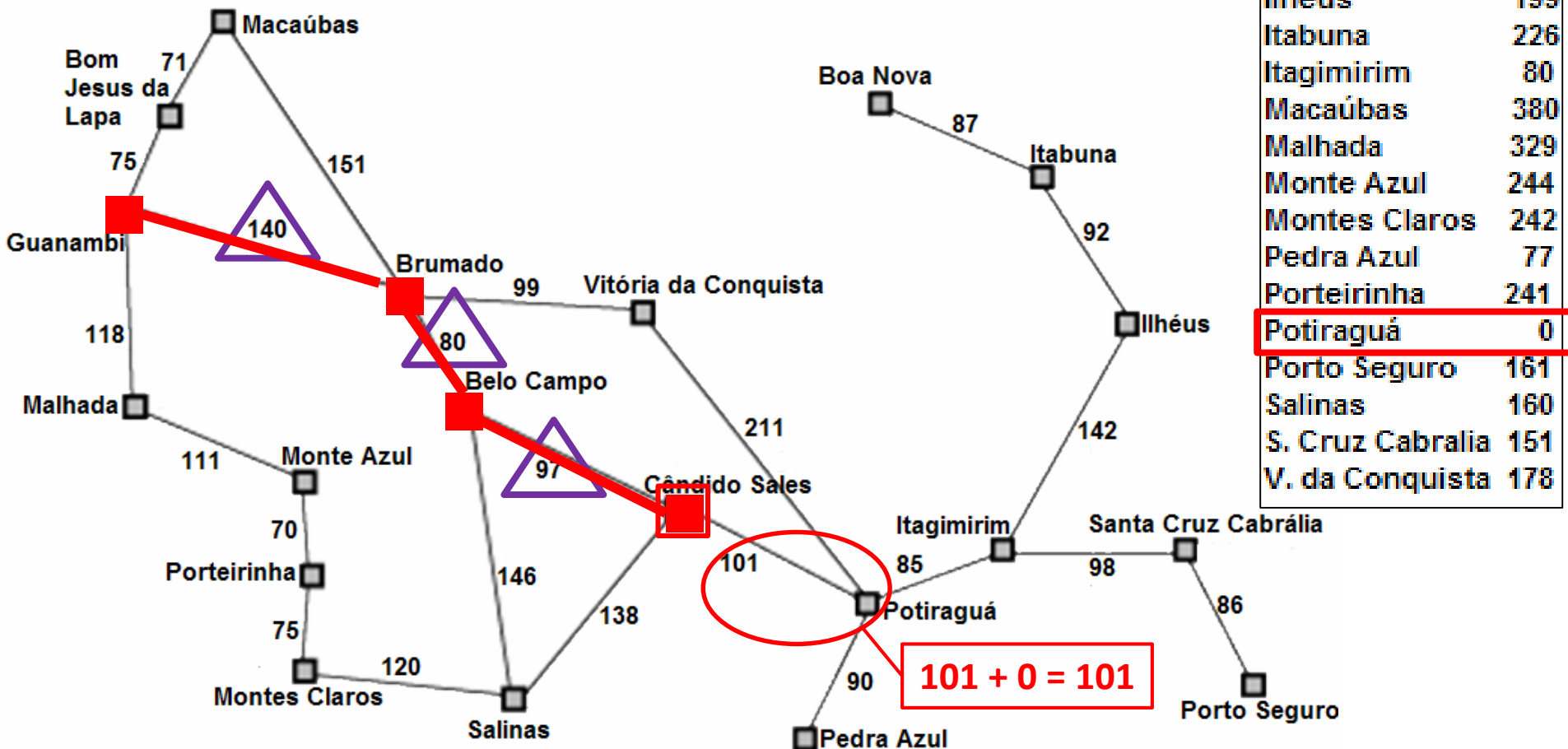
Distância em
linha reta para
Potiraguá

Belo Campo	193
Boa Nova	234
B. Jesus da Lapa	374
Brumado	253
Cândido Sales	98
Guanambi	366
Ilhéus	199
Itabuna	226
Itagimirim	80
Macaúbas	380
Malhada	329
Monte Azul	244
Montes Claros	242
Pedra Azul	77
Porteirinha	241
Potiraguá	0
Porto Seguro	161
Salinas	160
S. Cruz Cabralia	151
V. da Conquista	178

3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

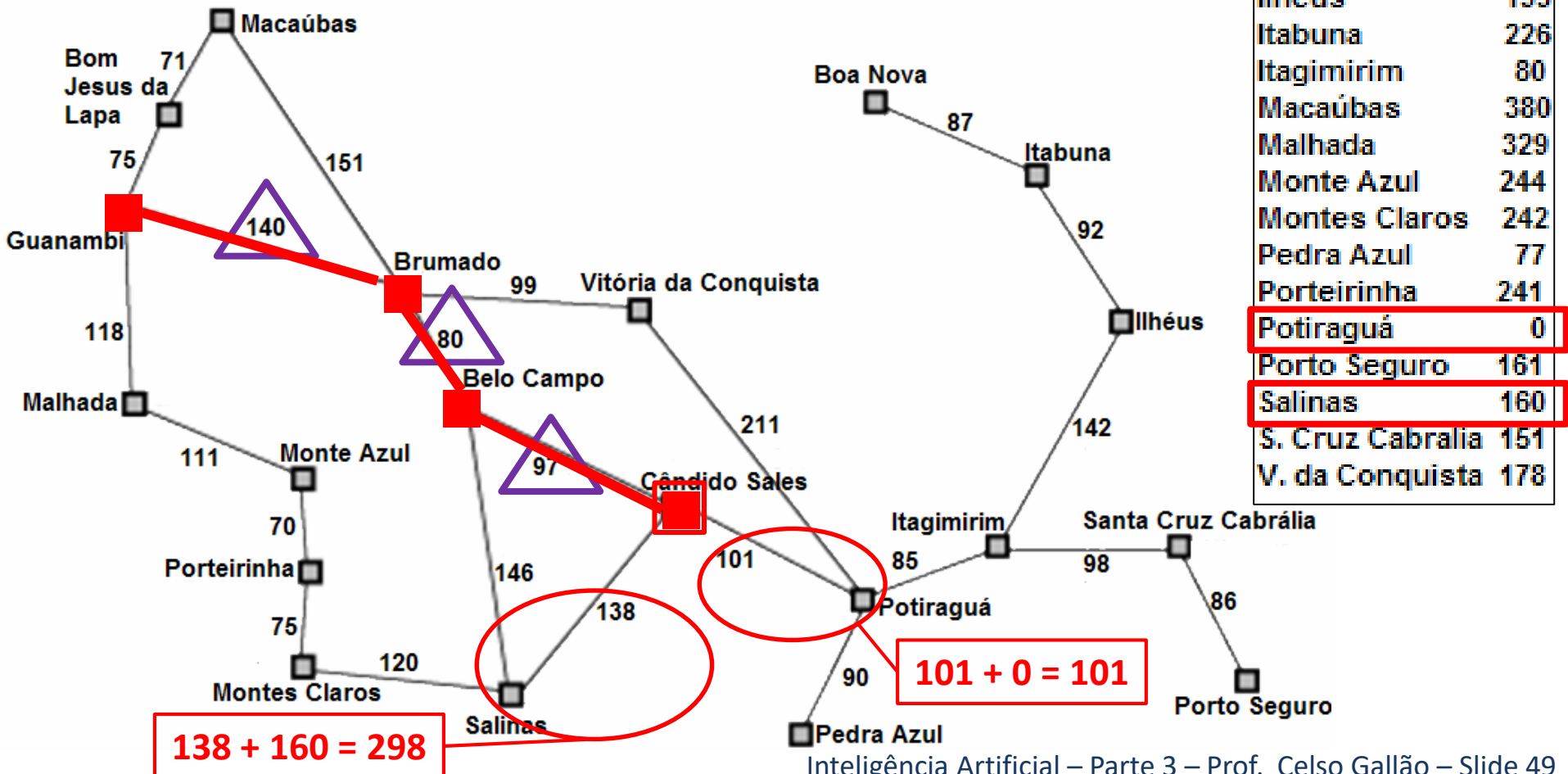
- Objetivo: de Guanambi à Potiraguá:



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

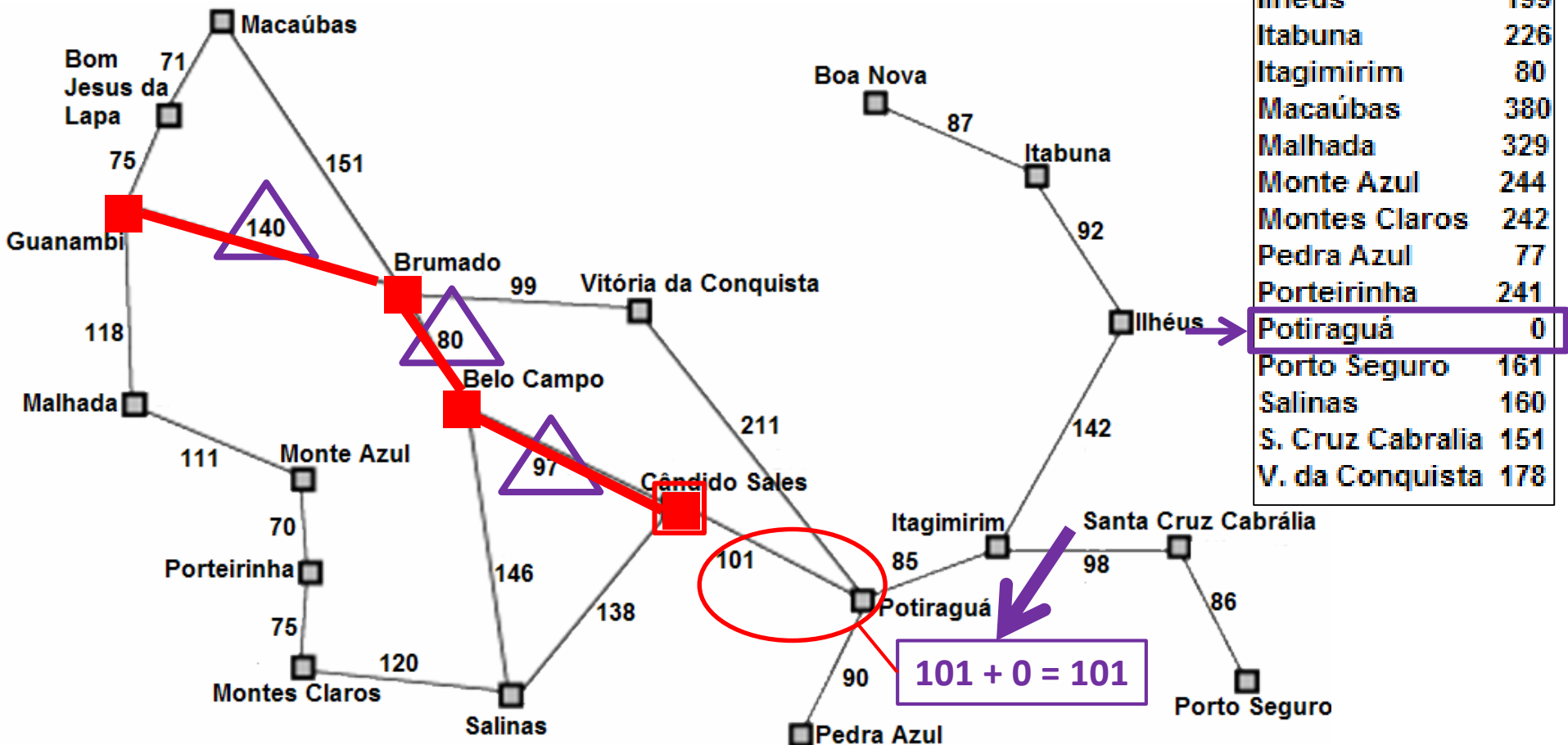
- Objetivo: de Guanambi à Potiraguá:



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

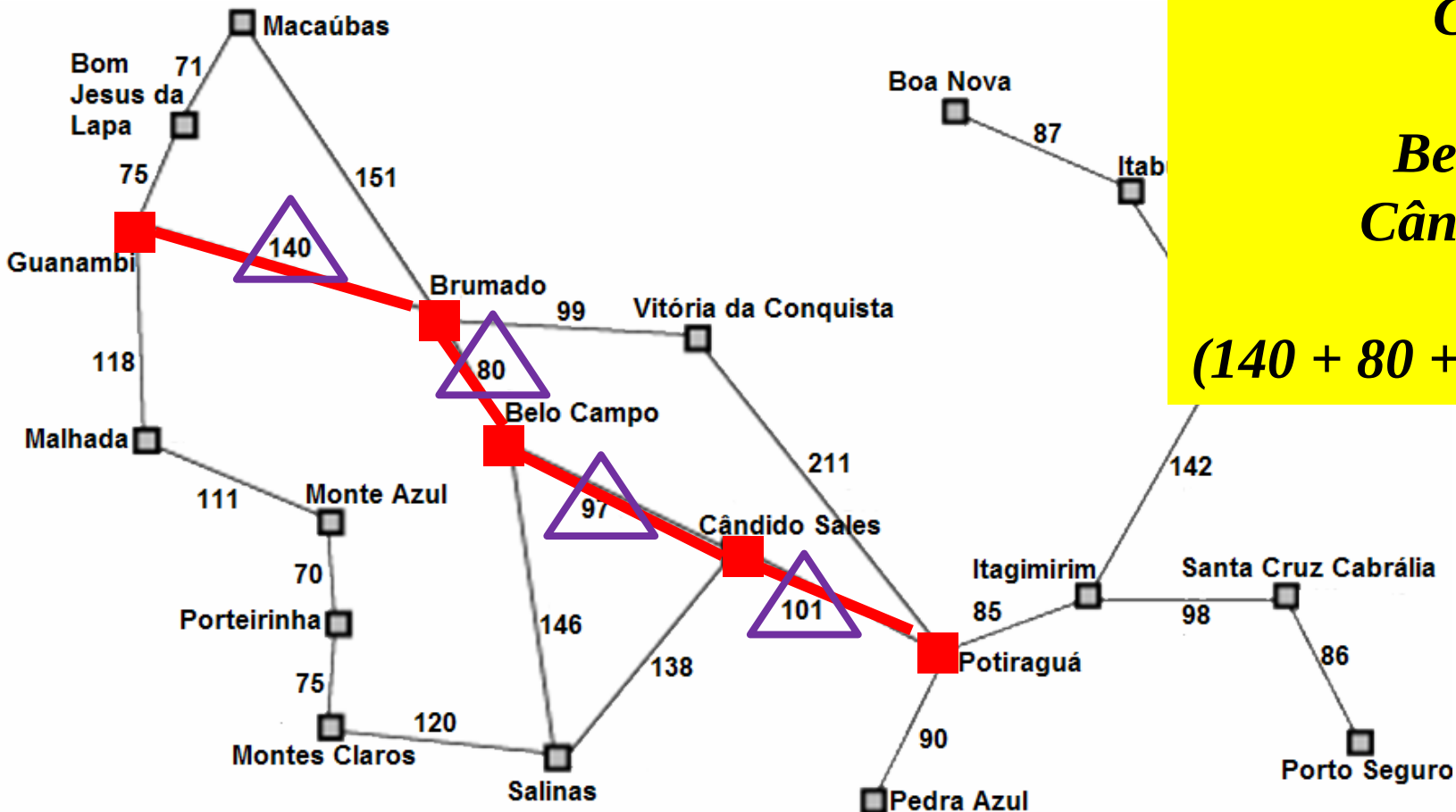
- Objetivo: de Guanambi à Potiraguá:



3 - Algoritmos *Best-First* (A^*)

Exemplo 2: A Busca do Menor Caminho

- Objetivo: de **Guanambi** à **Potiraguá**:



A solução encontrada
foi a melhor:
***Guanambi,
Brumado,
Belo Campo,
Cândido Sales
= 418 Km
(140 + 80 + 97 + 101)***

3 - Algoritmos *Best-First* (A^*)

Construa a Árvore de Busca deste Exercício :

3 - Algoritmos *Best-First* (A^*)

Algoritmo: A Busca do Menor Caminho (Análise)

- Função de avaliação: escolher o nó cuja distância é a menor em linha reta do nó final somado com a distância para chegar até este nó.

- Sejam:

$Q(n)$ = conjunto de nós a serem pesquisados.

$h'(n)$ = a distância em linha reta de n até o nó final.

$g(n)$ = custo do nó inicial até o nó n selecionado.

$S(n)$ = o estado da busca.

H = a somatória das distâncias reais.

$f(n) = h'(n) + g(n)$.

3 - Algoritmos *Best-First* (A^*)

Admissibilidade de A^* :

- Se existe uma solução, então A^* sempre encontrará a solução ótima se:
 - o fator de ramificação for finito;
 - o custo do caminho nunca decrescer (não possuir custos negativos);
 - $h(n)$ nunca superestima o custo do caminho, ou seja,
 $h(n) \leq \text{custo real do caminho (heurística admissível)}$

3 - Algoritmos *Best-First* (A^*)

Algoritmo: A Busca do Menor Caminho

1. Inicialize Q com o nó de busca $S(0)$;
2. Inicialize $h'(n)$ com a distância de $S(0)$;
3. Inicialize $g(n)$ com 0 .
4. Calcule $f(n) = h'(n) + g(n)$.
5. Escolha o melhor elemento de $f(n)$ e adicione em H sua distância $g(n)$.
6. Guarde em $S(n)$ o estado n .
7. Remova todos os elementos de $Q(n)$ e de $h'(n)$.

3 - Algoritmos *Best-First* (A^*)

Algoritmo: A Busca do Menor Caminho

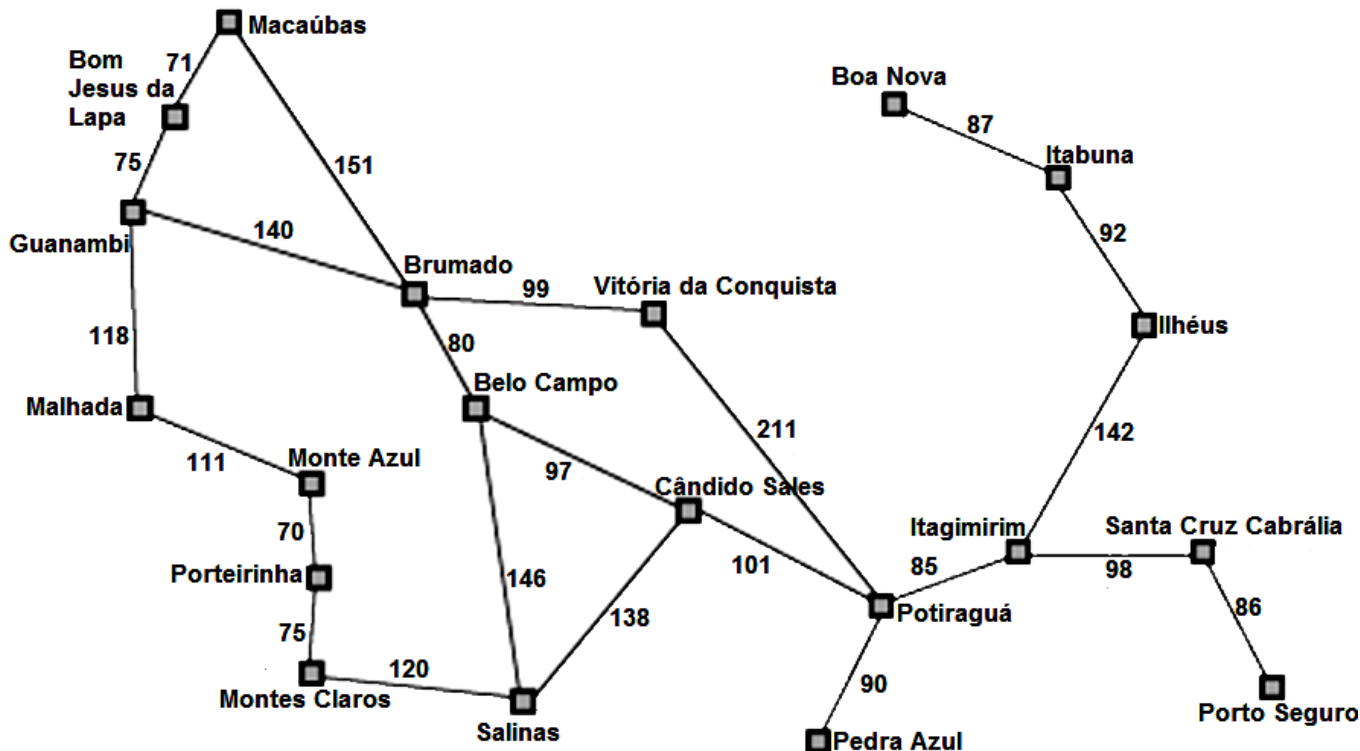
8. Verifique se $f(n) = 0$:
 - então, retorne H e $S(n)$ e encerre;
 - senão, prossiga.
9. Encontre os descendentes do estado (n) e os adicione em $Q(n)$.
10. Crie todas as extensões de n para cada descendente encontrado, adicione-as em $h'(n)$ e suas distâncias do nó pai em $g(n)$.
11. Retorne ao passo 4.

3 - Algoritmos *Best-First* (A^*)

Exercício 3.1:

O mapa abaixo mostra distâncias entre cidades da *Bahia*. A tabela abaixo mostra as distâncias em linha reta de cada cidade até *Vitória da Conquista*. Utilizando o algoritmo A^* , sendo $f(n) = g(n) + h'(n)$, faça:

a) A árvore de busca de *Porteirinha* até *Vitória da Conquista*:



Distância em linha reta até Vitória da Conquista	
Belo Campo	82
Boa Nova	130
Bom Jesus da Lapa	218
Brumado	87
Cândido Sales	85
Guanambi	240
Ilhéus	217
Itabuna	185
Itagimirim	195
Macaúbas	228
Malhada	219
Monte Azul	163
Montes Claros	234
Pedra Azul	200
Porteirinha	221
Porto Seguro	327
Potiraguá	178
Salinas	174
Santa Cruz Cabralia	272
Vitória da Conquista	0

3 - Algoritmos *Best-First* (A^*)

Exercício 3.1:

b) O cálculo da distância total percorrida.

c) Esta é a solução ótima? Justifique.

4

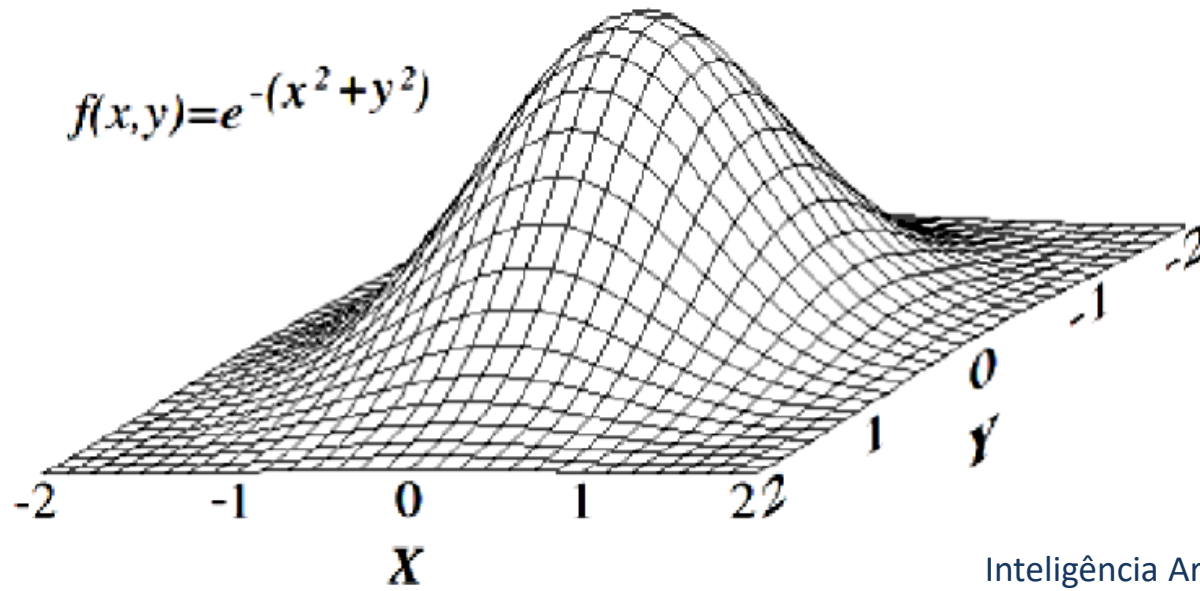
Algoritmos *Best-First* (*Hill Climbing*)

4 - Algoritmos *Best-First* (Hill Climbing)

Características de *Subida de Encosta*:

Hill Climbing é semelhante ao algoritmo que um alpinista usaria para encontrar um topo de uma montanha. Porém, com neblina e amnésia.

- Caminhar sempre na direção em que **o terreno é mais elevado**, sem nunca descer ou voltar para trás.

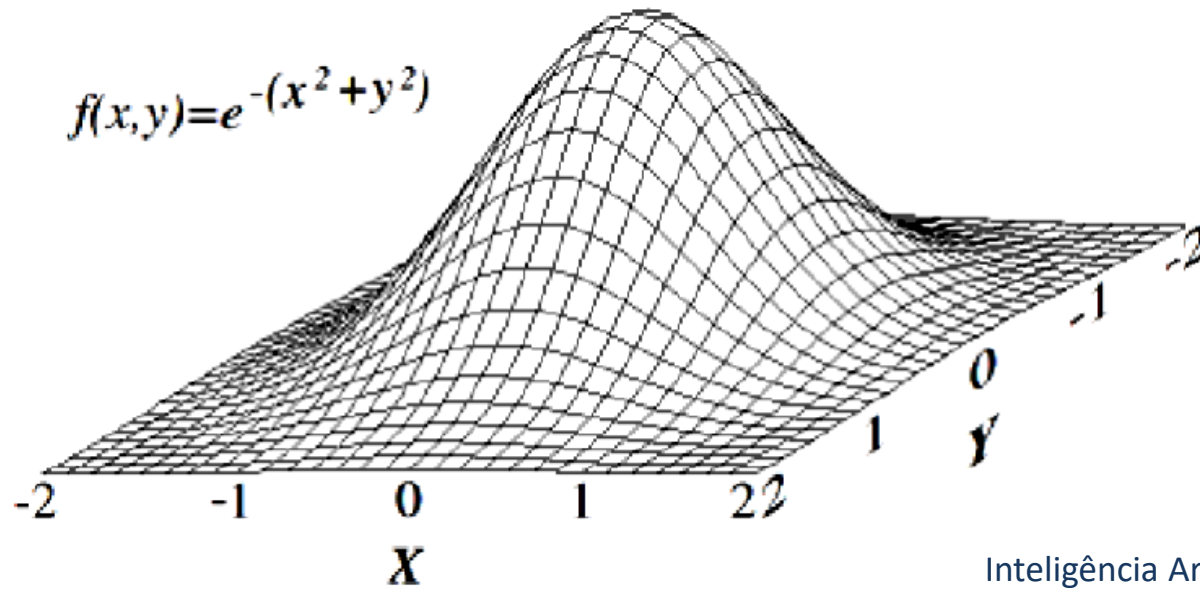


4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

Hill Climbing é um método de busca local que utiliza um procedimento de melhora iterativa (*iterative improvement*).

- A estratégia é aplicada a **um único ponto** x (solução candidata) no espaço de busca.
- Realiza uma **busca local** em torno de seu ponto inicial.

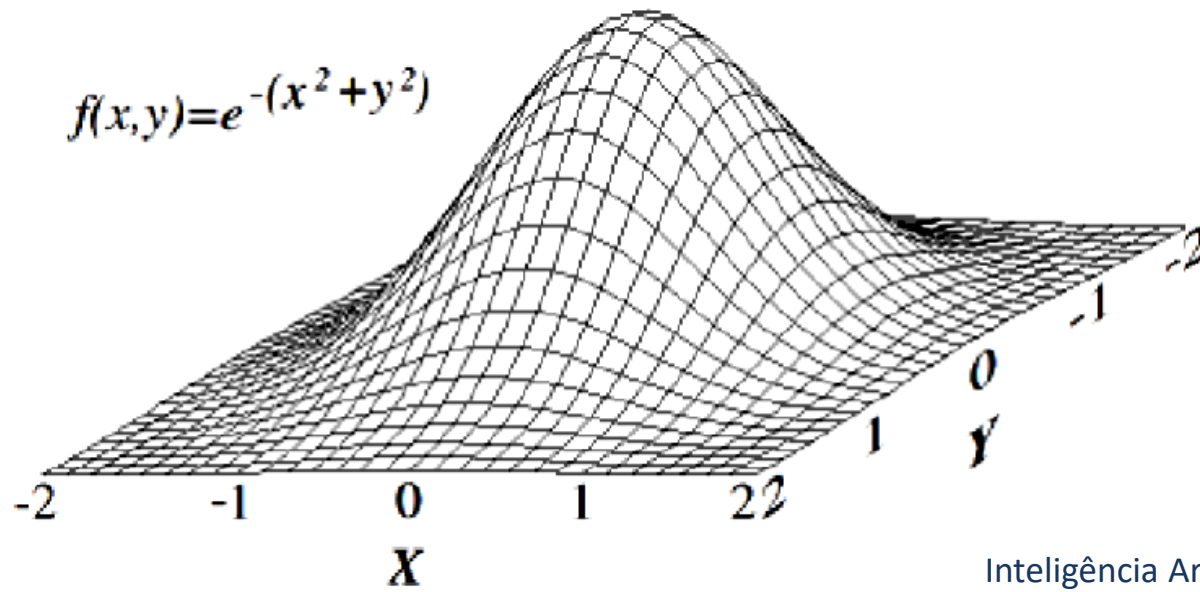


4 - Algoritmos *Best-First* (Hill Climbing)

Características de *Subida de Encosta*:

Para aliviar suas limitações, Hill Climbing pode ser implementado partindo de várias condições iniciais aleatórias e armazenando a melhor solução obtida.

- As soluções encontradas, apesar de boas, **não são** necessariamente ótimas.



4 - Algoritmos *Best-First* (Hill Climbing)

Características de *Subida de Encosta*:

Hill Climbing é um algoritmo genérico de procura direcionada, que tenta maximizar (ou minimizar) a função de avaliação $f(x)$, onde x são estados discretos representados pelos vértices de um gráfico.

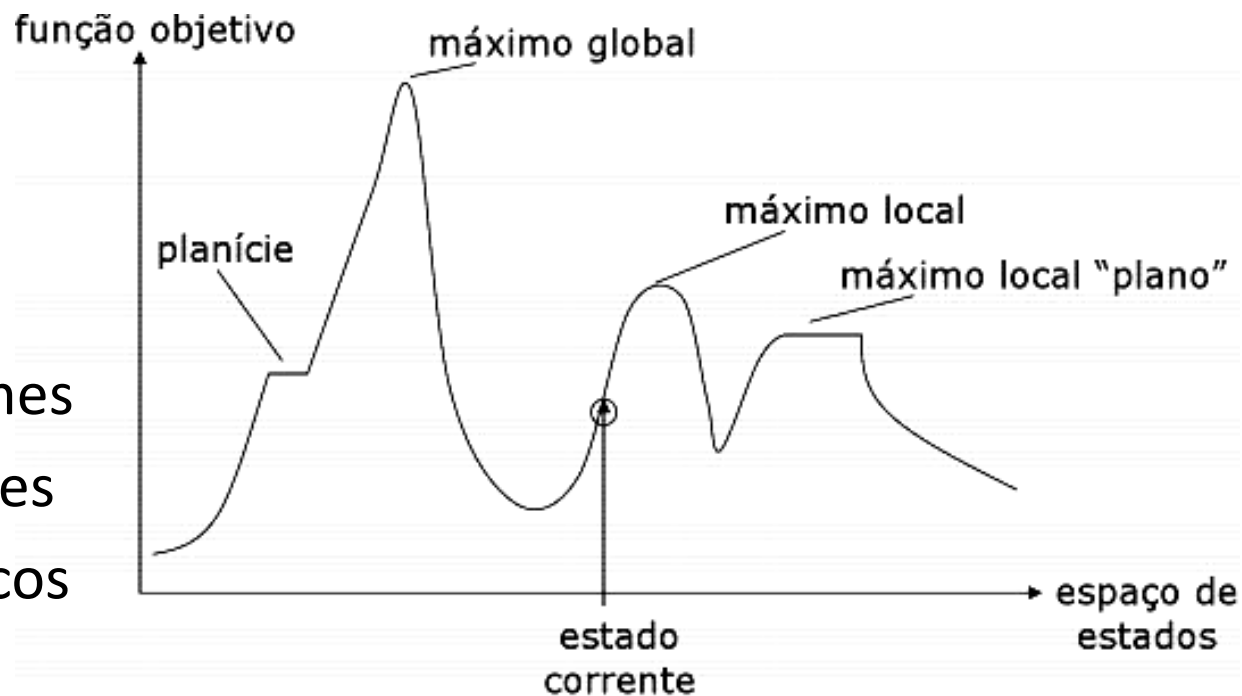
- É aplicável quando uma procura exaustiva do espaço de estado não é viável.
- O algoritmo é executado com limite de tempo.
- Quando o limite de tempo disponível não é totalmente utilizado, o algoritmo deve ser sucessivamente reiniciado utilizando novos pontos de partida.

4 - Algoritmos *Best-First* (Hill Climbing)

Características de *Subida de Encosta*:

O algoritmo move-se sempre na direção que apresentar a maior taxa de variação. Isso pode acarretar alguns problemas:

- 1 – Máximos Locais
- 2 – Encostas ou Cumes
- 3 – Platôs ou Planícies
- 4 – Anéis Concêntricos



4 - Algoritmos *Best-First* (*Hill Climbing*)

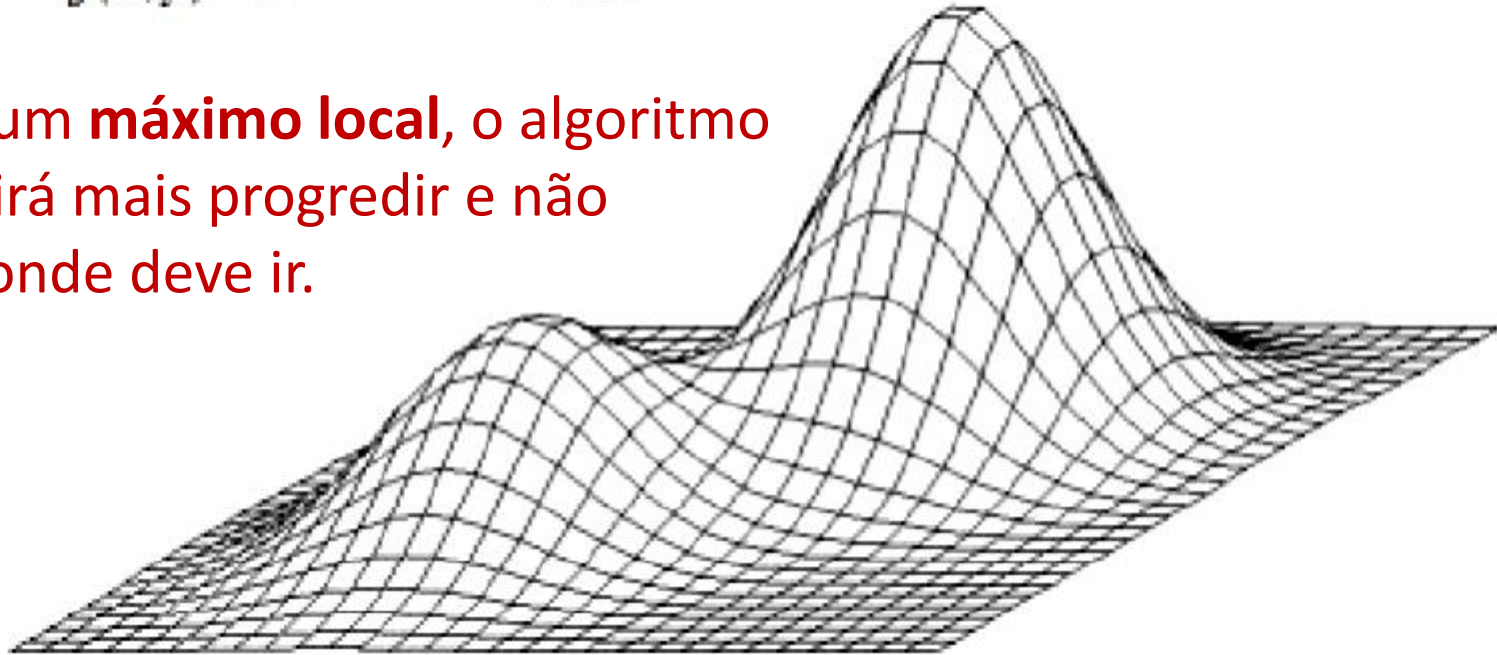
Características de *Subida de Encosta*:

1º Problema: Máximos locais

São picos mais altos do que seus vizinhos locais, porém, são mais baixos do que o pico máximo global.

$$f(x,y) = e^{-(x^2+y^2)} + 2e^{-((x-1.7)^2+(y-1.7)^2)}$$

Após atingir um **máximo local**, o algoritmo não conseguirá mais progredir e não saberá para onde deve ir.



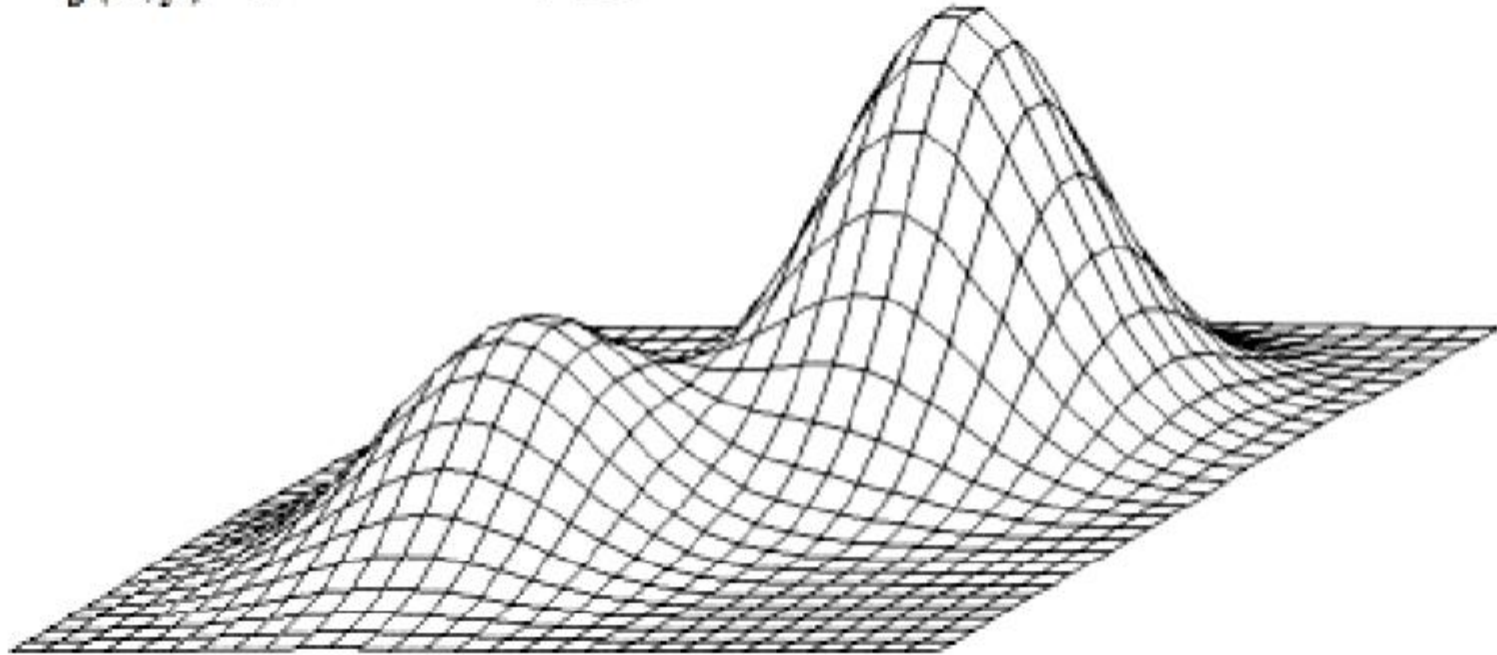
4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

1º Problema: Máximos locais

Exemplo: **no jogo de Xadrez**, eliminar a Rainha do adversário pode levar o jogador a perder o jogo.

$$f(x,y) = e^{-(x^2+y^2)} + 2e^{-((x-1.7)^2+(y-1.7)^2)}$$



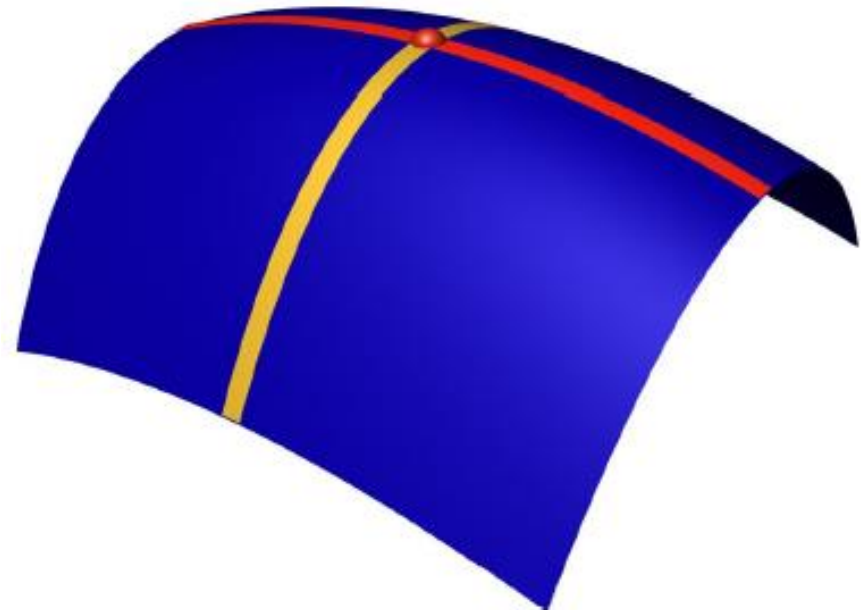
4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

2º Problema: Encostas ou Cumes

Apesar de estar em uma direção que pode levar ao pico, pode ser que nenhum dos operadores válidos conduza o algoritmo à convergência dentro de um tempo aceitável.

A convergência é muito lenta para o máximo global, e pode nunca ser atingida.



4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

3º Problema: Platôs ou Planícies

É uma região do espaço de estados onde a função de avaliação é plana, ou seja, dá o mesmo resultado.

Todos os movimentos locais são iguais (**taxa de variação zero**), o algoritmo pode não encontrar uma saída, e parar.



4 - Algoritmos *Best-First* (Hill Climbing)

Características de *Subida de Encosta*:

3º Problema: Platôs ou Planícies

Exemplo: **no jogo de 8-Puzzle**: Nenhum movimento possível vai influenciar no valor de $f(x,y)$, pois os números 6 e 8 estão à mesma distância de seus respectivos destinos (possuem o mesmo custo), de acordo com a ***Distância Manhattan***.

8		6
5	4	7
2	3	1

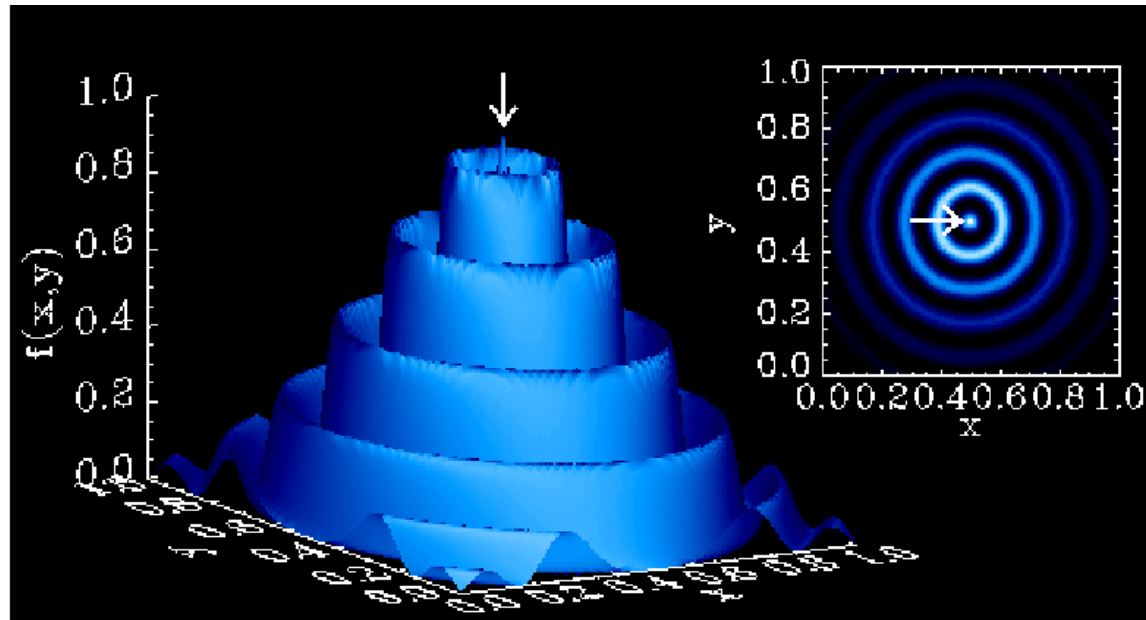
	1	2
3	4	5
6	7	8

4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

4º Problema: Anéis Concêntricos

É a existência de anéis em diversos níveis, produzindo muitos máximos locais, intransponíveis.



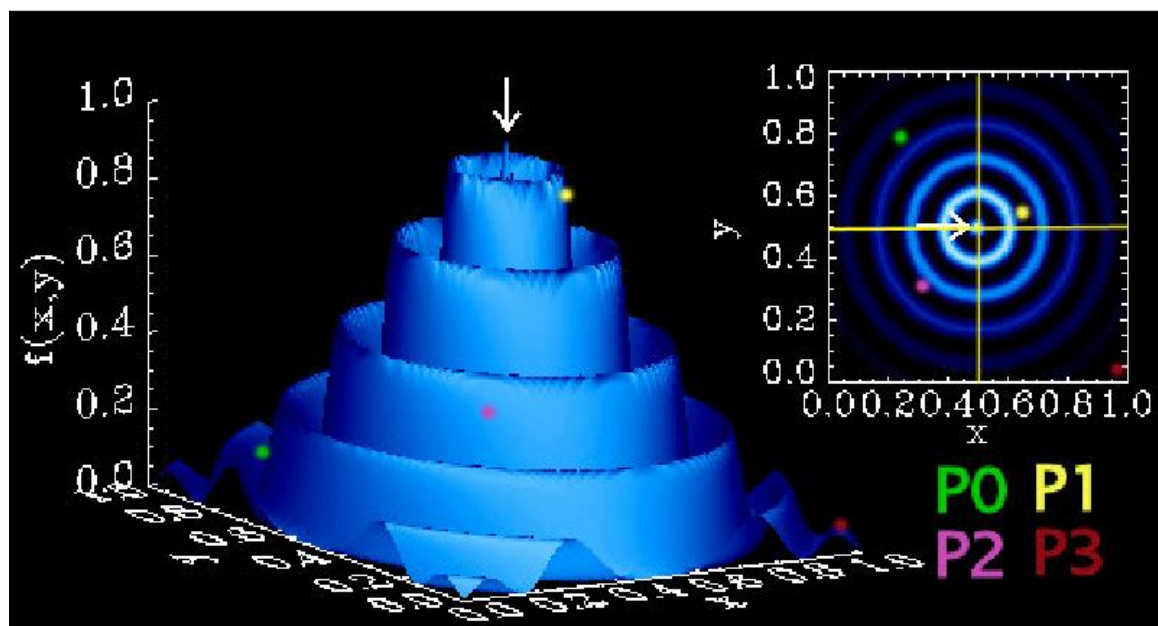
4 - Algoritmos *Best-First* (*Hill Climbing*)

Características de *Subida de Encosta*:

4º Problema: Anéis Concêntricos

É a existência de anéis em diversos níveis, produzindo muitos máximos globais, intransponíveis.

A solução é a **paralelização** utilizando N processos com N pontos iniciais, cada processo é responsável por um intervalo do espaço de estados de $f(x)$.



4 - Algoritmos *Best-First* (Hill Climbing)

Algoritmo de *Subida de Encosta*:

1. Inicialize aleatoriamente o ponto x na região factível do problema.
2. À cada iteração, um novo ponto x' é selecionado aplicando-se uma pequena perturbação no ponto atual, ou seja, selecionando-se um ponto x' que esteja na vizinhança de x .
3. Verifique se x' apresenta um melhor valor para a função de avaliação:
 - então, o novo ponto torna-se o ponto atual.
 - senão, verifique se o objetivo foi atingido.

4 - Algoritmos *Best-First* (*Hill Climbing*)

Algoritmo de *Subida de Encosta*:

4. Se o objetivo não foi atingido, retorne ao passo 2.
- Um objetivo pode estar baseado na quantidade de iterações, ou em um limiar pré-estabelecido, por exemplo.

4 - Algoritmos *Best-First* (*Hill Climbing*)

Algoritmo de *Subida de Encosta* (Análise):

A paralelização do *Hill Climbing* permitirá:

1. Maximizar a hipótese de encontrar uma solução ótima.
2. Optar pela melhor solução dentre as diversas encontradas.

A abordagem paralela do *Hill Climbing* não introduz necessariamente melhores tempos de computação, mas muito mais importante do que isso, permite encontrar soluções para o problema de melhor qualidade.

Bibliografias

Obrigatórias:

1. CORMEN, LEISERSON, RIVEST, STEIN, **Algoritmos: Teoria e Prática**. 2ª ed. Rio de Janeiro: Campus, 2002: Capítulo 16.
2. RUSSELL, Stuart J; NORVIG, Peter. **Inteligência Artificial**. 2ª ed. Rio de Janeiro: Campus, 2004, Capítulo 4.
3. LUGER, George. **Inteligência Artificial: Estruturas e Estratégias para a Resolução de Problemas Complexos**. 4ª ed. Porto Alegre: Bookman, 2004, Capítulo 1.

Bibliografias

Recomendadas:

1. CORMEN, LEISERSON, RIVEST, STEIN, **Algoritmos: Teoria e Prática**. 2ª ed. Rio de Janeiro: Campus, 2002: Capítulo 15.