

# Greedy Randomized Adaptive Search Procedures (GRASP)

Marcone Jamilson Freitas Souza<sup>1,2,3</sup>

Puca Huachi Vaz Penna<sup>1</sup>

<sup>1</sup> Departamento de Computação

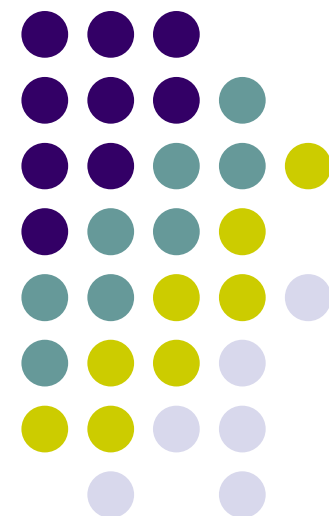
<sup>1</sup> Programa de Pós-Graduação em Ciência da Computação  
Universidade Federal de Ouro Preto

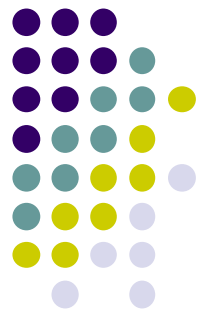
<sup>2</sup> Programa de Pós-graduação em Modelagem Matemática e  
Computacional / CEFET-MG

<sup>3</sup> Programa de Pós-graduação em Instrumentação, Controle e  
Automação de Processos de Mineração / ITV/UFOP

[www.decom.ufop.br/prof/marcone](http://www.decom.ufop.br/prof/marcone), [www.decom.ufop.br/puca](http://www.decom.ufop.br/puca)

E-mail: {marcone,puca}@ufop.edu.br

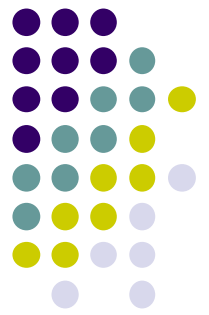




# Fundamentação do método

- Proposto por Feo e Resende (1995):
  - Feo, T. A. e Resende, M. G. C. Greedy Randomized Adaptive Search Procedures. Journal of Global Optimization, 6:109-133, 1995.
- Constituído por duas fases que são aplicadas iterativamente até um critério de parada ser atendido:
  - Construção:
    - constrói uma solução de forma parcialmente gulosa
  - Refinamento:
    - melhora a solução construída por meio da aplicação de um método de busca local

# Fundamentação da fase de construção do método



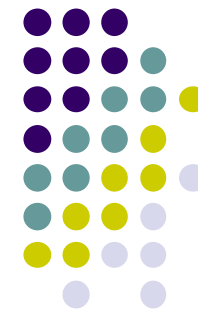
- A fase de construção combina os pontos positivos das estratégias de construção gulosa e aleatória de uma solução
- Pontos positivos da construção gulosa:
  - Gera soluções de boa qualidade
  - Soluções convergem rapidamente para um ótimo local
- Pontos positivos da construção aleatória:
  - Diversifica o espaço de soluções do problema
  - É capaz de gerar, pelo menos teoricamente, todo o espaço de soluções do problema

# Fundamentação da fase de construção do método



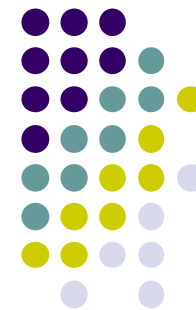
- Ideia da fase de construção GRASP:
  - Cria-se uma lista de candidatos (LC)
  - Avalia-se cada candidato por meio de uma função gulosa  $g$ , que mensura a importância da inserção de cada candidato na solução parcial
  - Enquanto houver candidatos da LC:
    - Forma-se uma lista restrita de candidatos (LRC) com os melhores candidatos segundo a função  $g$
    - Escolhe-se aleatoriamente um elemento candidato dessa LRC
    - Insere-se esse elemento da LRC na solução parcial  $s$
    - Atualiza-se a LC, excluindo-se o elemento inserido na solução parcial  $s$
  - A fase de construção termina retornando uma solução  $s$

# Formação da Lista Restrita de Candidatos (LRC)



- Seja LC uma lista de candidatos
- Seja  $g(t)$  uma função que avalia a inserção do candidato  $t \in LC$  na solução parcial  $s$
- $LRC = \{t \in LC \mid g(t) \leq g_{\min} + \alpha \times (g_{\max} - g_{\min})\}$
- $g_{\min}$  é o menor valor de  $g$  dentre os candidatos  $t \in LC$
- $g_{\max}$  é o maior valor de  $g$  dentre os candidatos  $t \in LC$
- $\alpha \in [0, 1]$  é um parâmetro que mensura o grau de aleatoriedade/gulosidade da escolha do candidato  $t \in LC$ 
  - Se  $\alpha = 0$ , a escolha é gulosa
  - Se  $\alpha = 1$ , a escolha é totalmente aleatória

# Exemplo de formação da LRC para o PCV

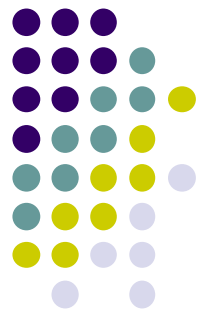


- Depósito na origem: 0
- $LC = \{ 1, 2, 3, 4, 5, 6, 7 \}$
- $g(t)$  = distância do candidato  $t \in LC$  à última cidade inserida na rota. No caso,  $g(t) = d_{0t}$

t	$g(t) = d_{0t}$
1	20
2	9
3	13
4	17
5	10
6	18
7	23

- $g_{\min} = 9$ ;  $g_{\max} = 23$ ; Seja  $\alpha = 0,4$
- Solução parcial  $s = \{0\}$
- $LRC = \{t \in LC \mid g(t) \leq g_{\min} + \alpha \times (g_{\max} - g_{\min})\}$
- $LRC = \{t \in LC \mid g(t) \leq 9 + 0,4 \times (23 - 9)\}$
- $LRC = \{t \in LC \mid g(t) \leq 14,6\}$
- $LRC = \{2, 3, 5\}$
- $s \leftarrow s \cup \{3\} = \{0, 3\}$
- $LC \leftarrow LC \setminus \{3\} = \{ 1, 2, 4, 5, 6, 7 \}$

# Exemplo de formação da LRC para o PCV

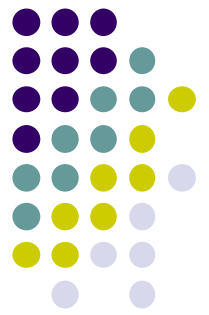


- Solução parcial  $s = \{0, 3\}$
- $LC = \{1, 2, 4, 5, 6, 7\}$
- $g(t)$  = distância do candidato  $t \in LC$  à última cidade inserida na rota. No caso,  $g(t) = d_{3t}$

t	$g(t) = d_{3t}$
1	12
2	4
4	14
5	16
6	13
7	8

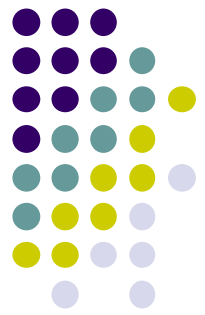
- $g_{\min} = 4$ ;  $g_{\max} = 16$ ; Seja  $\alpha = 0,4$
- Solução parcial  $s = \{0, 3\}$
- $LRC = \{t \in LC \mid g(t) \leq g_{\min} + \alpha \times (g_{\max} - g_{\min})\}$
- $LRC = \{t \in LC \mid g(t) \leq 4 + 0,4 \times (16 - 4)\}$
- $LRC = \{t \in LC \mid g(t) \leq 8,8\}$
- $LRC = \{2, 7\}$
- $s \leftarrow s \cup \{2\} = \{0, 3, 2\}$
- $LC \leftarrow LC \setminus \{2\} = \{1, 4, 5, 6, 7\}$

# Exemplo de formação da LRC para o PCV



- Solução parcial  $s = \{0, 3, 2\}$
- $LC = \{1, 4, 5, 6, 7\}$
- $g(t)$  = distância do candidato  $t \in LC$  à última cidade inserida na rota. No caso,  $g(t) = d_{2t}$
- A construção termina quando  $LC = \emptyset$
- A cada aplicação da fase de construção GRASP novas soluções parcialmente gulosas são geradas, dada a aleatoriedade do processo construtivo

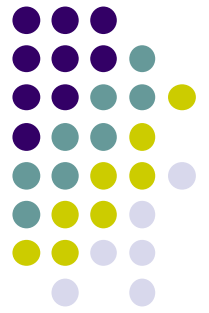




# Algoritmo GRASP

```
procedimento  $GRASP(f(.), g(.), N(.), GRASP_{max}, s)$   
1   $f^* \leftarrow \infty$ ;  
2  para ( $Iter = 1, 2, \dots, GRASP_{max}$ ) faça  
3       $Construcao(g(.), \alpha, s)$ ;  
4       $BuscaLocal(f(.), N(.), s)$ ;  
5      se ( $f(s) < f^*$ ) então  
6           $s^* \leftarrow s$ ;  
7           $f^* \leftarrow f(s)$ ;  
8      fim-se;  
9  fim-para;  
10  $s \leftarrow s^*$ ;  
11 Retorne  $s$ ;  
fim  $GRASP$ 
```

# Parâmetros do método GRASP



- Há somente dois parâmetros no método:
  - GRASPmax: número de iterações em que o método é aplicado:
  - $\alpha$ : fator de aleatoriedade / gulosidade da fase de construção
- Esses parâmetros devem ser calibrados apropriadamente