



- DCC136 - Algoritmos Construtivos

– Heurísticas de Construção –
(CONTINUAÇÃO....)



Conceitos Comuns

- Roteiro
 - Algoritmo de Construção
 - Algoritmo Guloso
 - Algoritmo Guloso randomizado, adaptativo, reativo

Métodos Construtivos

- Algoritmo GULOSO
 - Seleciona-se o elemento que minimiza o incremento no custo da solução parcial de tal forma que ao final se obtenha uma solução viável.
 - Algoritmo míope: enxerga somente o custo ou benefício associado ao próximo passo. Considera apenas a próxima decisão.
 - Algoritmo ALEATÓRIO
-

Métodos Construtivos

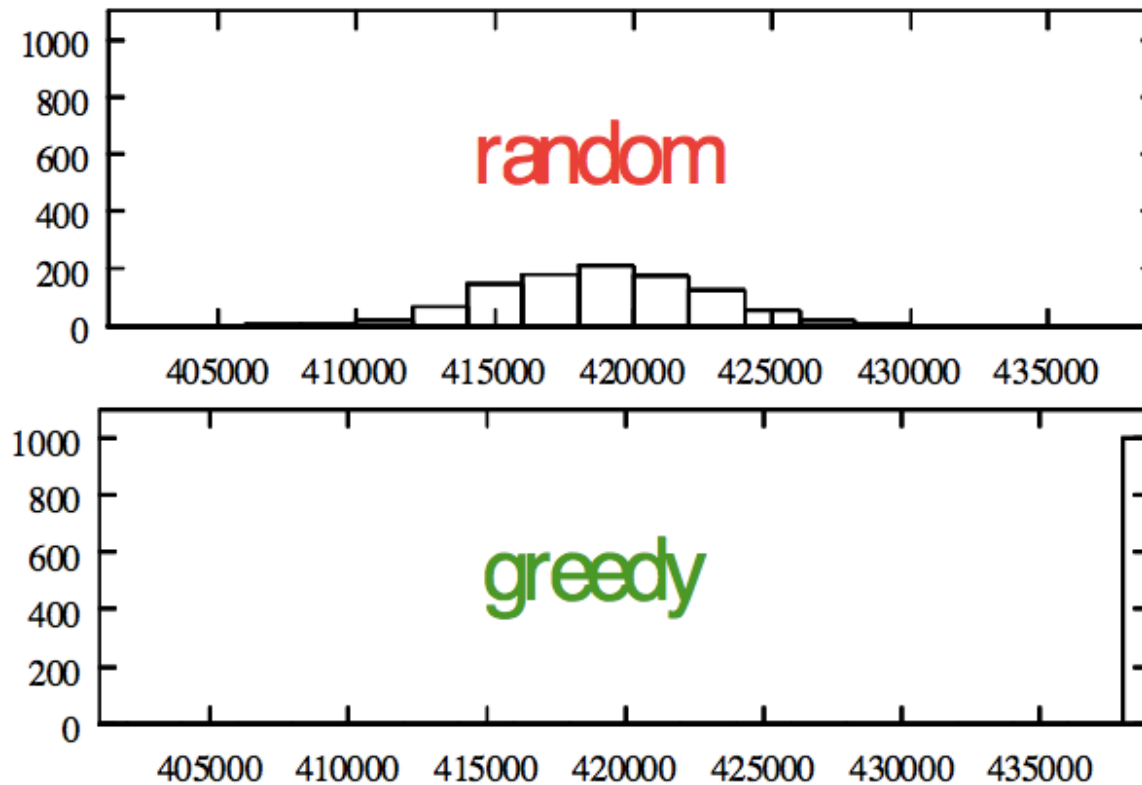
```
procedimento ConstrucaoAleatoria( $g(.)$ ,  $s$ );  
1   $s \leftarrow \emptyset$ ;  
2  Inicialize o conjunto  $C$  de elementos candidatos;  
3  enquanto ( $C \neq \emptyset$ ) faça  
4      Escolha aleatoriamente  $t_{escolhido} \in C$ ;  
5       $s \leftarrow s \cup \{t_{escolhido}\}$ ;  
6      Atualize o conjunto  $C$  de elementos candidatos;  
7  fim-enquanto;  
8  Retorne  $s$ ;  
fim ConstrucaoAleatoria;
```

Figura 2: Heurística de construção aleatória de uma solução inicial

Métodos Construtivos

- Algoritmo GULOSO x ALEATÓRIO
 - Qualidade x Tempo processamento

Métodos Construtivos



Métodos Construtivos

• Problema da Sequência mais Próxima

Considere Σ um alfabeto com k caracteres, sendo que Σ_m representa todas as possíveis sequências de tamanho m usando os caracteres de Σ . Para duas sequências s e t pertencentes a Σ_m , a distância de Hamming entre estas sequências é definida como o número de posições diferentes entre s e t , representada por $d_H(s, t)$. Ou seja, $d_H(s, t) = \sum_{i=1}^m \delta(s_i, t_i)$, onde s_i e t_i representam, respectivamente, o i -ésimo caractere das sequências s e t , sendo que $\delta(s_i, t_i) = 0$ se $s_i = t_i$; ou $\delta(s_i, t_i) = 1$, caso contrário ($s_i \neq t_i$).

Para o Problema da Sequência Mais Próxima, dado um conjunto $S_c = \{s_1, s_2, \dots, s_n\}$ contendo n sequências onde $S_c \subseteq \Sigma_m$, o objetivo é determinar uma sequência x de tamanho m sobre o alfabeto Σ que possua a menor diferença em relação às cadeias de S_c , ou seja, obter x tal que $\operatorname{argmin}_{x \in \Sigma_m} \max_{s^i \in S_c} d_H(x, s^i)$.

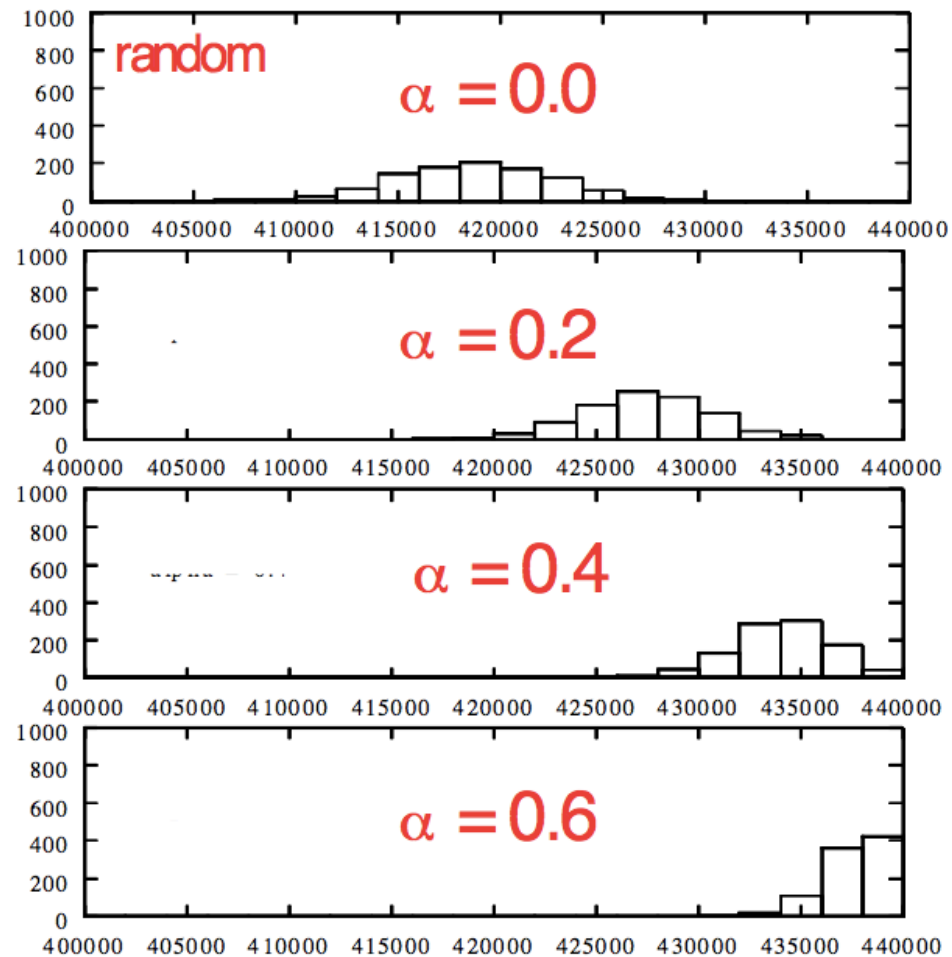
Por exemplo, dado o conjunto $S_c = \{ACGTA, TTACA, CCGCA, GGGGA\}$, onde $\Sigma = \{CTGA\}$ e $m=5$. Para uma sequência $x = TCGCA$, considerando cada sequência $s^i \in S_c$, tem-se que $d_H(x, s^1) = 2$, $d_H(x, s^2) = 2$, $d_H(x, s^3) = 1$, $d_H(x, s^4) = 3$. Logo, a distância máxima entre a sequência x e as cadeias de S_c é definida por $d_c = \max_{i=1..4} \{d_H(x, s^i)\} = 3$.

Métodos Construtivos

- Para o PSP
 - Guloso
 - Aleatório
 - Adaptativo

Métodos Construtivos

- Semi-guloso



Métodos Construtivos

- Guloso Aleatorizado
 - Algoritmo guloso encontra sempre a mesma solução para um dado problema, exceto por eventuais empates.
 - A aleatorização permite alcançar diversidade nas soluções encontradas.
 - Criar uma lista de candidatos e forçar uma escolha aleatória a cada iteração.

Métodos Construtivos

- Guloso Aleatorizado
 - A **qualidade** da solução obtida depende da qualidade dos elementos na lista de candidatos
 - A **diversidade** das soluções encontradas depende da cardinalidade da lista

Métodos Construtivos

- Guloso Aleatorizado
 - Casos extremos:
 - algoritmo guloso puro
 - solução gerada aleatoriamente

Métodos Construtivos

- Guloso Aleatorizado
 - Na estratégia gulosa os elementos candidatos são classificados segundo uma heurística gulosa.
 - No guloso aleatorizado um subconjunto dos melhores elementos são inseridos na **lista restrita de candidatos - LRC**.
 - Para determinar elementos a serem inseridos na LRC pode-se considerar *Qualidade* ou *Quantidade*.

Métodos Construtivos

- Guloso Aleatorizado

- *Qualidade*

- Parâmetro p que representa número máximo de elementos na LRC
 - p melhores elementos são inseridos na LRC

- *Quantidade*

- Para ser incluído na lista, verifica-se qualidade do candidato.
 - Considerando c_max e c_min os valores máximo e mínimo associado aos candidatos, o limite pode ser definido como:

$$\{e_i/c_i \leq c^{\min} + \alpha(c^{\max} - c^{\min})\}$$

Métodos Construtivos

Algorithm 2.18 Template of the greedy randomized algorithm.

$s = \{ \} ; /* \text{Initial solution (null)} */$

Evaluate the incremental costs of all candidate elements ;

Repeat

Build the restricted candidate list RCL ;

$/* \text{select a random element from the list } RCL */$

$e_i = \text{Random-Selection}(RCL) ;$

If $s \cup e_i \in F$ **Then** $/* \text{Test the feasibility of the solution} */$

$s = s \cup e_i ;$

Reevaluate the incremental costs of candidate elements ;

Until Complete solution found.

Métodos Construtivos

- Estratégia Reativa
 - Estratégia para ajuste de parâmetro (grau de aleatoriedade).

Dúvidas

- Perguntas ou comentários?



Luciana Brugiolo Gonçalves

lbrugiolo@ice.ufjf.br

Referências

- EL-GHAZALI TALBI. Metaheuristics: From Design to Implementation, Wiley, 2009.
- Notas de aula do prof Celso Ribeiro (UFF).
- Notas do prof. Maurício Resende (AT&T).

TRABALHO

- TRABALHO 1 (continuação):
 - The capacitated team orienteering problem
 - Propor um algoritmo guloso adaptativo aleatorizado para o CTOP.