MO824A/MC859A – Tópicos em Otimização Combinatória
Segundo semestre de 2020

Atividade 4 (extra-classe)
Entrega: 6 de novembro de 2020, até 23:59

Prof. Fábio Luiz Usberti (fusberti@ic.unicamp.br)
Prof. Celso Cavellucci (celsocv@ic.unicamp.br)

---

1 Objetivo

O objetivo desta atividade consiste na implementação (em grupos de **dois** ou **três** alunos) de uma metaheurística "GRASP" (*Greedy Randomized Adaptive Search Procedure*) para a solução de um problema de maximização de uma função binária quadrática ("quadractic binary function" – QBF).

2 GRASP

Para esta atividade é essencial a leitura da seguinte referência:

Título: Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations, and Applications.

Autores: Maurício G. C. Resende e Celso Ribeiro

Capítulo 10 do livro: M. Gendreau, J.-Y. Potvin (eds.), Handbook of Metaheuristics, International Series in Operations Research & Management Science 146, DOI 10.1007/978-1-4419-1665-5 10.

Leituras necessárias para a atividade: Do início até Seção 10.3.6 (inclusive) e Seções 10.7 e 10.8.

3 Problema MAX-QBF

Uma função binária quadrática (QBF) é uma função $f : \mathbb{B}^n \rightarrow \mathbb{R}$ que pode ser expressa como uma soma de termos quadráticos:

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \cdot x_i \cdot x_j$$

Onde $a_{ij} \in \mathbb{R} \;\; (i, j = 1, \ldots, n)$ são os coeficientes da função f. Em notação matricial, uma QBF pode ser expressa como:

$$f(\mathbf{x}) = \mathbf{x}' \cdot \mathbf{A} \cdot \mathbf{x}$$

Por exemplo:

$$
\begin{aligned}
f(\mathbf{x}) &= \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix} \cdot \begin{bmatrix} 0 & 1 & 2 \\ 0 & 0 & 3 \\ 0 & 0 & 4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\
&= \begin{bmatrix} 0 & x_1 & (2x_1 + 3x_2 + 4x_3) \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \\
&= x_1x_2 + 2x_1x_3 + 3x_2x_3 + 4x_3^2
\end{aligned}
$$

O problema de maximização de uma função binária quadrática (MAX-QBF) pode ser expresso como:

$$
Z = \max_{\mathbf{x}} \; f(\mathbf{x}) \, ,
$$

O MAX-QBF é um problema NP-difícil [1], mesmo que nenhuma restrição adicional seja imposta sobre as variáveis binárias $\mathbf{x}$. No entanto, se os coeficientes a_{ij} forem todos não-negativos, o problema torna-se trivial, uma vez que $x_i = 1 \;\; (i = 1, \ldots, n)$ é uma solução ótima.

4 Problema MAX-QBF com triplas proibidas

Uma variante do problema MAX-QBF é definida a seguir:

Problema MAX-QBFPT ("Maximum quadractic binary function with prohibited triples"): Considere o conjunto $\mathcal{T}$ de todas as triplas ordenadas, sem repetição, dos naturais de 1 a n, ou seja, $\mathcal{T} = \{(i, j, k) \in \mathbb{N}^3 : 1 \leqslant i < j < k \leqslant n\}$. No problema **MAX-QBFPT**, dado um conjunto de triplas proibidas $T \subseteq \mathcal{T}$, deseja-se maximizar uma função binária quadrática tal que x_i, x_j e x_k não podem ser todos iguais a 1, para todo $(i, j, k) \in T$. Este problema pode ser formulado da seguinte forma:

$$
\begin{array}{lll}
Max & Z = \displaystyle\sum_{i=1}^{n} \sum_{j=1}^{n} a_{ij} \cdot x_i \cdot x_j & \\
s.a. & & \\
 & x_i + x_j + x_k \leqslant 2 & \forall (i, j, k) \in T \\
 & x_i \in \mathbb{B} & \forall i = \{1, \ldots, n\}
\end{array}
$$

Onde $a_{ij} \in \mathbb{R} \;\; (i, j = 1, \ldots, n)$ são parâmetros do problema.

Para esta atividade, o conjunto de triplas proibidas T será definido da seguinte forma. Para cada natural $u \in [1, n]$ serão aplicadas duas funções $g, h : [1, n] \to [1, n]$ para gerar dois novos números que formarão uma tripla proibida. As funções g e h serão definidas tomando como base a seguinte função linear congruente l, tipicamente usada para a geração de números pseudo-aleatórios:

$$
l(u) = \;\; 1 + ((\pi_1 \cdot (u - 1) + \pi_2) \mod n)
$$

Onde π_1 e π_2 são normalmente escolhidos como números primos.

Para impedir que $g(u) = u$, define-se a função g da seguinte forma:

$$
g(u) = \begin{cases} l(u) & \text{se } l(u) \neq u \\ 1 + (l(u) \mod n) & \text{caso contrário} \end{cases}
$$

Para impedir que $h(u) = u$ ou $h(u) = g(u)$, define-se a função h da seguinte forma:

$$h(u) = \begin{cases} l(u) & \text{se } l(u) \neq u \wedge l(u) \neq g(u) \\ 1 + (l(u) \mod n) & \text{se } (1 + (l(u) \mod n)) \neq u \wedge (1 + (l(u) \mod n)) \neq g(u) \\ 1 + ((l(u) + 1) \mod n) & \text{caso contrário} \end{cases}$$

Para esta atividade, os números primos utilizados para a função g são $\pi_1 = 131$ e $\pi_2 = 1031$; para a função h os números primos são $\pi_1 = 193$ e $\pi_2 = 1093$.

A partir das informações acima, torna-se possível definir o conjunto de triplas proibidas desta atividade como $T = \{(i, j, k) \in \mathcal{T} : \forall u \in [1, n], (i, j, k) = sort(\{u, g(u), h(u)\})\}$.

5 Requisitos da atividade

Esta atividade envolve a implementação de uma metaheurística GRASP como um método de solução para o MAX-QBFPT. Para esta atividade você pode utilizar como base o Framework GRASP em Java, disponível no ensino aberto, desenvolvido pelos docentes desta disciplina.

Para esta atividade é necessário a implementação de pelo menos dois *métodos de construção alternativos* discutidos no artigo de referência do GRASP [2]:

1. *Random plus greedy*
2. *Sampled greedy construction*
3. *Reactive GRASP*
4. *Cost perturbations*
5. *Bias functions*
6. *Intelligent construction*
7. *POP in construction*

A atividade exige a entrega do código-fonte e de um relatório (até 5 páginas) descrevendo brevemente as seguintes informações sobre a metaheurística desenvolvida:

- Descrição do problema: variáveis de decisão e modelo matemático.
- Metodologia: descrição da lista restrita de candidatos (RCL – "restricted candidate list"), heurística construtiva, método de construção alternativos, operadores de busca local, métodos de busca (*first-improving* e *best-improving*), critérios de parada.
- Resultados: tabela de resultados e análise dos desempenhos obtidos para cada metodologia.

Devem ser avaliados dois métodos de busca (*first-improving* e *best-improving*), dois valores para o parâmetro $\alpha \in [0, 1]$ da lista RCL e três métodos de construção (padrão e alternativo). Desse modo, uma sugestão de possíveis configurações são:

1. `PADRÃO`: GRASP com parâmetro α_1, *first-improving* e heurística construtiva padrão.
2. `PADRÃO+ALPHA`: GRASP `PADRÃO` mas com parâmetro α_2.
3. `PADRÃO+BEST`: GRASP `PADRÃO` mas com *best-improving*.

4. `PADRÃO+HC1`: GRASP `PADRÃO` mas com método de construção alternativo 1.

5. `PADRÃO+HC2`: GRASP `PADRÃO` mas com método de construção alternativo 2.

Procure organizar os resultados em uma tabela, avaliando qual a estratégia obteve o melhor desempenho.

6 Instâncias

Testes computacionais devem ser realizados com um conjunto de sete instâncias disponíveis no ambiente ensino aberto. Adote um tempo de execução para cada instância de 30 minutos. Os nomes das instâncias, suas dimensões e os valores das soluções ótimas (quando conhecidos) são fornecidos a seguir:

Instância	$\lvert\mathbf{x}\rvert$	MAX-QBF (Z^*)	MAX-QBFPT (Z^*)
qbf020	20	**151**	**125**
qbf040	40	**429**	**366**
qbf060	60	**576**	$[508, 576]$
qbf080	80	**1000**	$[843, 1000]$
qbf100	100	$[1468, 1539]$	$[1263, 1539]$
qbf200	200	$[5385, 5826]$	$[3813, 5826]$
qbf400	400	$[14826, 16625]$	$[9645, 16625]$

Para fins de conferência, seguem as triplas proibidas para a instância qbf020:

```
[1, 12, 14]
[2, 3, 7]
[3, 14, 20]
[4, 5, 13]
[5, 6, 16]
[6, 7, 19]
[7, 12, 18]
[5, 8, 9]
[9, 18, 20]
[10, 11, 12]
[2, 4, 11]
[12, 13, 17]
[4, 10, 13]
[3, 14, 15]
[6, 15, 16]
[9, 16, 17]
[2, 8, 17]
[15, 18, 19]
[8, 10, 19]
[1, 2, 20]
```

7 Referências

1. Kochenberger, et al. The unconstrained binary quadratic programming problem: a survey. **J Comb Optim** (2014). 28:58–81. DOI:10.1007/s10878-014-9734-0.

2. Resende, M. G. C. e Ribeiro, C. Greedy Randomized Adaptive Search Procedures: Advances, Hybridizations, and Applications. In: M. Gendreau, J.-Y. Potvin (eds.), **Handbook of Metaheuristics**, International Series in Operations Research & Management Science 146, DOI: 10.1007/978-1-4419-1665-5.