MO824/MC859 – Tópicos em Otimização Combinatória
Segundo semestre de 2025

Atividade 4
Entrega: 10 de outubro de 2025, até meio-dia

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 equipes de três de uma metaheurística fundamentada em "Algoritmo Genético" (*Genetic Algorithm*) para a solução de um problema de maximização de uma função binária quadrática ("quadractic binary function" – QBF).

2 Algoritmo Genético

Para esta atividade é essencial a leitura da seguinte referência:
Título: Genetic Algorithms
Autores: Colin R. Reeves
Capítulo 5 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.

3 Problema MAX-QBF

Uma função binária quadrática (QBF) é uma função $f : \mathbb{B}^n \to \mathbb{R}$ 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 Set Cover

No problema MAX-SC-QBF, desejamos maximizar uma função binária quadrática sujeita a restrições de cobertura de conjuntos, onde o universo a ser coberto é o próprio conjunto de variáveis da QBF.

Seja $N = \{1, \ldots, n\}$ o *conjunto de variáveis* da QBF. Seja $\mathcal{S} = \{S_1, \ldots, S_n\}$ uma coleção de subconjuntos $S_i \subseteq N$, representando as variáveis que o subconjunto i cobre. Cada subconjunto i está associado a uma variável binária x_i indicando sua seleção. Para cada par (i, j) de subconjuntos, temos um coeficiente $a_{ij} \in \mathbb{R}$ que representa o ganho (positivo ou negativo) por selecionar ambos simultaneamente.

Nosso objetivo é selecionar subconjuntos de forma que:

- todas as variáveis da QBF sejam cobertas, ou seja, para todo $k \in N$, exista ao menos um S_i tal que $k \in S_i$ e $x_i = 1$;
- seja maximizado o ganho quadrático total derivado das interações entre subconjuntos selecionados.

5 Requisitos da atividade

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

Para esta atividade é necessário a implementação de pelo menos duas *estratégias evolutivas alternativas* discutidas em aula:

1. *Latin hypercube*
2. *Stochastic universal selection*
3. *Uniform crossover*
4. *Adaptive mutation*
5. *Steady-state*

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

- **Distribuição de tarefas**: incluir um parágrafo redigido por cada aluno descrevendo suas contribuições para a realização da atividade.
- **Descrição do problema**: variáveis de decisão e modelo matemático.

- **Metodologia**: codificação de uma solução, geração da população inicial, método de seleção de indivíduos, método de recombinação ("crossover"), método de mutação, critérios de parada, estratégias evolutivas alternativas.
- **Resultados**: tabela de resultados e análise dos desempenhos obtidos para cada metodologia.

Devem ser avaliados dois tamanhos de população (constante ou em função do tamanho da instância), duas taxas de mutação (constante ou em função do tamanho da instância) e três estratégias evolutivas (padrão e duas alternativas). Desse modo, uma sugestão de possíveis configurações são:

1. `PADRÃO`: Algoritmo Genético com tamanho de população P_1, taxa de mutação M_1 e construção aleatória da população.
2. `PADRÃO+POP`: Algoritmo Genético `PADRÃO` mas com tamanho de população P_2.
3. `PADRÃO+MUT`: Algoritmo Genético `PADRÃO` mas com taxa de mutação M_2.
4. `PADRÃO+EVOL1`: Algoritmo Genético `PADRÃO` mas com estratégia evolutiva alternativa 1.
5. `PADRÃO+EVOL2`: Algoritmo Genético `PADRÃO` mas com estratégia evolutiva alternativa 2.

Procure organizar os resultados em tabelas/gráficos, avaliando o desempenho de cada estratégia.

6 Instâncias

Testes computacionais devem ser realizados com um conjunto de pelo menos 15 instâncias que foram desenvolvidas na Atividade 1. Adote um tempo de execução para cada instância de pelo menos 30 minutos.

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. Colin R. Reeves. Genetic Algorithms. 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.