Relatório da Atividade 1 de MO824/MC859

Everton Romanzini Colombo (257234)
Filipe Caetano Oliveira de Resende (266879)

Agosto de 2025

[0]**Nota ao professor:** O terceiro integrante desta equipe, Gabriel Antunes Rodrigues (260442), indicou que desistiria da disciplina e, portanto, não teve participação nesta atividade.

Descrição dos Geradores de Instâncias

Fizemos três geradores de instâncias: `instance_gen1`, `instance_gen2` e `instance_gen3`. Todas as instâncias geradas obedecem $n \in \{25, 50, 100, 200, 400\}$ e $0 \leq |S_i| \leq n$ para cada subconjunto S_i $(i = 1, 2, \ldots, n)$.

As instâncias gen1 são completamente aleatórias. As instâncias gen2 possuem todos os seus subconjuntos com tamanho pelo menos $0.8n$. Como estes subconjuntos são muito grandes, é bem provável que a solução do problema sem as restrições de Set Cover seja uma cobertura. Para cada um dos geradores, os coeficientes da função objetivo são valores de ponto flutuante de -10.00 a 10.00, com duas casas decimais. No caso das instâncias gen3, pelo menos 80% dos coeficientes são não-negativos (de 0.00 a 10.00). A ideia por trás dessa estratégia é aproximar os problemas ao caso trivial de quando todos os coeficientes são positivos. Geramos 5 instâncias com cada um dos geradores, e os resultados e suas respectivas análises são apresentados na última seção deste relatório.

Modelo Matemático

O problema foi modelado da seguinte forma:

Parâmetros

- Matriz triangular superior $A \in \mathbb{R}^{n \times n}$
- Família $\mathcal{F}$ de n subconjuntos do universo $\mathcal{U} = [n]$: $\mathcal{F} = \{S_i \subseteq \mathcal{U} | i \in [n]\}$

Índices

$$\mathcal{I} := \{1, 2, \ldots, n\}, \qquad \mathcal{P} := \{(i, j) \in \mathcal{I} \times \mathcal{I} \mid i \leq j\},$$

(definidos para simplificar a notação)

Variáveis

- Variáveis indicadoras $x_i \in \mathbb{B}$, $i \in \mathcal{I}$. Quando $x_i = 1$, entende-se que o conjunto S_i foi selecionado.
- Variáveis ajudantes $y_{ij} \in \mathbb{B}$, $(i, j) \in \mathcal{P}$. Foram utilizadas para linearizar o problema, e serão tais que $y_{ij} = 1 \iff x_i = 1 \wedge x_j = 1$.

Objetivo

$$\max \quad \sum_{(i,j) \in \mathcal{P}} A_{ij}\, y_{ij}.$$

que é equivalente ao objetivo mostrado no PDF da ativdade (mostrado abaixo), mas linearizado.

$$\max \quad x\, A\, x^T$$

Restrições

1. Restrições de definição linear dos y_{ij}:

$$y_{ij} \leq x_i \tag{1}$$
$$y_{ij} \leq x_j \tag{2}$$
$$y_{ij} \geq x_i + x_j - 1 \tag{3}$$
$$\forall (i,j) \in \mathcal{P}.$$

2. Restrições de Set-Cover:

$$\sum_{i \in \mathcal{I}:\, u \in S_i} x_i \;\geq\; 1 \qquad \forall u \in \mathcal{U}. \tag{4}$$

3. Restrições de Integralidade:

$$x_i \in \{0,1\} \quad \forall i \in \mathcal{I}, \qquad y_{ij} \in \{0,1\} \quad \forall (i,j) \in \mathcal{P}.$$

Especificação do Ambiente de Execução

O problema foi modelado em `Python 3.10` com a biblioteca `pyomo` na versão `6.9.3`, e foi resolvido utilizando o solver `Gurobi` na versão `12.0.3`.

Os experimentos foram rodados em uma máquina com as seguintes especificações:

```
Gurobi Optimizer version 12.0.3 build v12.0.3rc0
(mac64[arm] - Darwin 24.6.0 24G84)

CPU model:  Apple M1
Thread count:  8 physical cores, 8 logical processors,
using up to 8 threads
```

Arquivos de código, de instâncias, e de resultados estão contidos no seguinte repositório do GitHub: `https://github.com/Everton-Colombo/MO824-A1`.

Resultados dos Experimentos & Análise

Instâncias gen1

As instâncias 3 e 4 foram resolvidas rapidamente por serem pequenas (valores de n de 50 e 25, respectivamente). As instâncias 1,2 e 5 possuem valores de n de 200, 200 e 100, respectivamente. Por serem grandes, o Gurobi não conseguiu resolvê-las dentro de 10 minutos, e pelos gaps possivelmente nem chegou perto. No Gurobi, os gaps relativos são dados por:

$$MIP\ Gap = \frac{|Incumbent\ Objective|}{|Best\ Objective\ Bound - Incumbent\ Objective|}$$

Instância	Valor do Objetivo	Gap (relativo)	Tempo do Solver (seg)
instance1.txt	4288.58	226.7549%	600.03
instance2.txt	4508.92	208.1693%	600.03
instance3.txt	645.62	0.0031%	7.41
instance4.txt	261.06	0.0000%	0.10
instance5.txt	1478.35	47.7979%	600.01

Tabela 1: Resultados para as instâncias de gen1.

- *Incumbent Objective*: melhor valor objetivo de solução inteira encontrado até então.
- *Best Objective Bound*: melhor limitante dual encontrado pra o valor objetivo de soluções inteiras até então.

Instâncias gen2

Instância	Valor do Objetivo	Gap (relativo)	Tempo do Solver (seg)
instance1.txt	575.87	0.0087%	7.13
instance2.txt	9694.43	827.3087%	600.08
instance3.txt	210.16	0.0048%	0.10
instance4.txt	7419.65	1137.8072%	600.08
instance5.txt	1550.49	38.7555%	600.01

Tabela 2: Resultados para as instâncias de gen2.

Na Tabela 2 temos os resultados das execuções do Gurobi para instâncias de gen2. Novamente, o solver resolveu rapidamente instâncias com $n \leq 50$ (instâncias 1 e 3). Para as demais instâncias, as execuções de 10 minutos terminaram com gaps bem grandes, e portanto é provável que as melhores soluções obtidas estejam bem longe do ótimo.

Instância	Valor do Objetivo	Gap (relativo)	Tempo do Solver (seg)
instance1.txt	575.87	0.0000%	0.09
instance2.txt	14472.64	13.0731%	600.01
instance3.txt	210.16	0.0000%	0.02
instance4.txt	13589.69	14.9315%	600.01
instance5.txt	1550.49	13.5480%	600.01

Tabela 3: Resultados para as instâncias de gen2 sem restrições de Set Cover.

Rodamos todas as instâncias de gen2 removendo as restrições de Set Cover para comparar os resultados (veja a Tabela 3). Como os subconjuntos das instâncias de gen2 são bem grandes, é provável que haja uma solução ótima do MAX-QBF tradicional que seja uma cobertura. Isto aconteceu para as instâncias 1 e 3, nas quais o valor ótimo não mudou. Quanto às demais instâncias, até mesmo a 5, não há como saber pois a execução foi interrompida após 10 minutos.

Instâncias gen3

Antes de analisar os resultados de gen3, é importante destacar o fato de que, para qualquer instância com solução viável do MAX-QBF com Set Cover, temos que:

- A soma dos coeficientes da função objetivo é um limitante primal. Isto se dá pelo fato de que a escolha de todos os subconjuntos forma uma cobertura.
- A soma dos coeficientes não-negativos é um limitante dual.

No caso de instâncias com alto percentual de coeficientes não-negativos na função objetivo, estas duas somas dão valores bem próximos, e portanto o Branch-and-Bound já começa com um gap apertado (assumindo que o Gurobi perceba a existência destes excelentes limitantes). Muitas vezes, apertado o suficiente para que esta solução viável inicial (todos os subconjuntos) seja imediatamente retornada como solução ótima. Isto pode acontecer pois o Gurobi, por padrão, para sua execução quando o gap relativo é menor que certa tolerância. Por esta razão, o Gurobi resolveu rapidamente as instâncias de gen3 (veja os tempos na Tabela 4). De fato, os valores objetivo na Tabela 4 são exatamente as somas dos coeficientes das instâncias (veja a Tabela 5).

Instância	Valor do Objetivo	Gap (relativo)	Tempo do Solver (seg)
instance1.txt	238058.87	0.0000%	18.88
instance2.txt	14887.43	0.0066%	0.22
instance3.txt	59765.97	0.0000%	1.93
instance4.txt	981.05	0.0000%	0.01
instance5.txt	766.71	0.0000%	0.01

Tabela 4: Resultados para as instâncias de gen3.

Instância	Soma dos Coeficientes
instance1.txt	238058.87
instance2.txt	14887.43
instance3.txt	59765.97
instance4.txt	981.05
instance5.txt	766.71

Tabela 5: Soma dos coeficientes da função objetivo para instâncias de gen3.