

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

Atividade 1

Entrega: 2 de outubro até 23:59

Prof. Fábio Luiz Usberti (fusberty@ic.unicamp.br)

Prof. Celso Cavellucci (celsocv@ic.unicamp.br)

1 Objetivo

O objetivo desta atividade consiste na modelagem de um problema de programação linear inteira e a solução desse modelo utilizando o software Gurobi¹.

A atividade deve ser realizada em equipes de 2 a 3 alunos. Os docentes vão sortear as equipes aleatoriamente. As equipes com 2 alunos ganharão um bônus na nota em virtude do número menor de alunos.

2 Descrição do Problema

Uma grande companhia de papel e celulose possui F fábricas produtoras de papel para atender a demanda de J clientes. Cada fábrica pode escolher dentre L máquinas e M tipos de matéria-prima para produzir P tipos de papel. A companhia precisa desenvolver um plano de produção e transporte com o objetivo de minimizar os custos totais. Mais especificamente, a companhia deve determinar a quantidade de cada tipo de papel a ser produzida em cada máquina de cada fábrica e a quantidade que deve ser transportada de cada tipo de papel partindo de cada fábrica para cada consumidor. Os parâmetros do problema encontram-se abaixo:

- $D_{j,p}$ = demanda do cliente j em unidades de papel do tipo p ;
- $r_{m,p,l}$ = unidades de matéria-prima m necessárias para produzir uma unidade de papel do tipo p na máquina l ;
- $R_{m,f}$ = unidades de matéria-prima m disponíveis na fábrica f ;
- $C_{l,f}$ = capacidade disponível de produção, em unidades de papel, da máquina l na fábrica f ;
- $p_{p,l,f}$ = custo unitário de produção do papel tipo p utilizando a máquina l na fábrica f ;
- $t_{p,f,j}$ = custo unitário de transporte do papel tipo p partindo da fábrica f até o cliente j ;

3 Requisitos da atividade

3.1 Formulação do problema

Faça a formulação em programação linear inteira do problema de produção de papel.

¹Baixe o software no site www.gurobi.com. Esse software possui licença acadêmica gratuita e vem acompanhado de múltiplos códigos de exemplo em diversas linguagens de programação.

3.2 Geração de instâncias

Gere 10 instâncias aleatórias variando a quantidade de clientes $|J| = \{50, 100, 150, 200, 250, 300, 350, 400, 450, 500\}$ e os demais parâmetros de maneira aleatória e uniforme conforme os seguintes intervalos inteiros:

- $|F| \in [|J|, 2|J|]$
- $|L| \in [5, 10]$
- $|M| \in [5, 10]$
- $|P| \in [5, 10]$
- $D_{j,p} \in [10, 100]$
- $r_{m,p,l} \in [1, 10]$
- $R_{m,f} \in [100, 1000]$
- $C_{l,f} \in [10, 100]$
- $p_{p,l,f} \in [10, 100]$
- $t_{p,f,j} \in [10, 20]$

3.3 Execução de experimentos

Resolva as 10 instâncias propostas no solver Gurobi, anotando o tempo de execução e o custo da solução. Repita o processo, mas relaxe as restrições de integralidade das variáveis, ou seja, permita que todas as variáveis possam assumir valores contínuos. Você deve limitar o tempo de execução em 30 minutos.

3.4 Entrega

A atividade exige a entrega do código-fonte, das instâncias geradas e de um relatório (até 3 páginas) contendo:

- Modelo matemático: apresente o modelo e descreva o significado das variáveis de decisão e das restrições.
- Resultados: tabela de resultados contendo, para cada instância: quantidade de variáveis do modelo, custo da solução para o problema inteiro, tempo de execução para o problema inteiro, custo da solução para o problema relaxado (variáveis contínuas), tempo de execução para o problema relaxado (variáveis contínuas).
- Análise: avalie os resultados quanto aos custos obtidos e os tempos computacionais.

4 Referências

1. Guia para instalação e inclusão de licença no Gurobi:
https://www.gurobi.com/documentation/9.0/quickstart_linux/index.html
2. Exemplos de código para a solução de modelos no Gurobi:
<https://www.gurobi.com/resource/functional-code-examples/>