

O problema da elaboração de grade de horários: estudo de caso em uma Universidade brasileira

Rodrigo Leppaus de Araujo

Graduando do Curso de Bacharelado em Matemática, CEUNES, UFES
29930-000, São Mateus, ES
E-mail: lemao.ara@hotmail.com

Leonardo Delarmelina Secchin

Departamento de Matemática Aplicada, CEUNES, UFES
29930-000, São Mateus, ES
E-mail: lsecchin@ceunes.ufes.br

Palavras-chave: *problema de horários, programação inteira, branch-and-cut, timetabling.*

Resumo: *O problema da elaboração de grade de horários (timetabling problem) é objeto de vasto estudo. É frequente na literatura a abordagem via programação inteira para sua resolução. Neste trabalho, estudamos o problema de grade de horários escolares no Departamento de Matemática Aplicada da UFES. Modelamos o problema no contexto da programação inteira, baseado no trabalho de MirHassani [4], de forma a balancear fidelidade à realidade e eficiência computacional na resolução. Um método clássico branch-and-cut foi utilizado na resolução. Testes mostram que a estratégia desenvolvida é adequada e pode ser utilizada na prática mediante pequenas modificações.*

1 Introdução

O problema da elaboração de grade de horários (*timetabling problem*) - PGH - consiste, no contexto escolar, no escalonamento de um conjunto de disciplinas entre professores e salas de aula, durante um período de tempo pré-definido. É objeto de estudo de vários autores. Dentre eles, MirHassani [4] relata o problema na Faculdade de Matemática da Universidade de Tecnologia de Sharood, no Irã, onde um modelo de programação inteira é proposto. Tais modelos são frequentes na descrição do problema (outro exemplo interessante é o trabalho de Daskalaski e Birbas [2]). MirHassani propõe uma formulação simplificada e eficiente do ponto de vista computacional, já que algumas formulações de programação inteira requerem muito esforço computacional em sua resolução, por se tratar de um problema NP-difícil [4].

Várias técnicas foram utilizadas na resolução do PGH. Dentre elas, Simulating annealing, Busca Tabu, relaxação Lagrangeana e o método de Geração de Colunas. Para uma referenciação adequada, consulte [4, 1]. Em nossa abordagem, utilizamos um algoritmo *Branch-and-cut* para programação inteira.

Nossa proposta é contextualizar o problema à realidade do Departamento de Matemática Aplicada (DMA) da Universidade Federal do Espírito Santo. Com base no trabalho de MirHassani [4], utilizamos um modelo de programação inteira, propondo a adição de novas restrições e variáveis.

2 Testes computacionais

No trabalho de MirHassani [4], questões como a distribuição balanceada de carga horária entre os docentes, e a condensação de horários (diminuição das horas vagas entre aulas) não são contempladas. É de se observar que tais questões são fundamentais na realidade analisada. Propomos então a inserção de novas variáveis e restrições no modelo. Com basicamente duas simples novas restrições, o problema da distribuição de carga horária foi contornado, sem aumento significativo no tempo computacional em relação ao modelo original de MirHassani. O problema da condensação dos horários foi amenizado com a adição de restrições que impediam certos horários ocorrerem simultaneamente. Neste ponto, precisamos balancear o número desses impedimentos com a eficiência na resolução.

Os dados colhidos no DMA, e utilizados na simulação, têm as seguintes características, referentes ao semestre letivo 2012/1:

Descrição	Quantidade
Disciplina	46
Sala	20
Professor Efetivo em exercício	16
Professor Substituto	3

Tabela 1: Características dos dados coletados.

Das 20 salas de aula, 16 são salas de aula normal, 2 laboratórios de informática, 1 laboratório de matemática e 1 laboratório de ensino. Os 16 professores do quadro efetivo em exercício estavam disponíveis para aulas, chefia de Departamento e duas coordenações de cursos. Quase a totalidade dos professores disponíveis podiam ministrar aulas em qualquer horário da semana, a menos dos horários da reunião departamental.

Utilizamos nos testes um computador Intel Core 2 Quad 2.33 GHz, 2Gb de memória RAM e sistema operacional GNU/Linux 64 bits. O modelo, construído com auxílio da ferramenta ZIMPL [3], foi resolvido utilizando o IBM CPLEX 12.4. Realizamos testes trocando algumas restrições por outras equivalentes, de modo a avaliar o desempenho computacional nas diferentes configurações. Sem especificar aqui quais restrições, apresentamos os dados globais dos testes na Tabela 2.

Descrição	Teste 1	Teste 2	Teste 3	Teste 4
Número de variáveis	34.089	34.089	34.089	34.089
Número de restrições	4.464	4.464	4.464	4.464
Elementos não nulos	343.291	375.371	371.123	339.043
Tempo da resolução	152.06sec	161.95sec	175.56sec	78.20sec

Tabela 2: Testes computacionais

Observamos que o número de restrições e variáveis em todos os testes são os mesmos. No entanto, o número de elementos não nulos na matriz das restrições do problema varia. Nota-se que o Teste 4 tem melhores resultados, e é o de menor número de elementos não nulos. O melhor desempenho do Teste 4 deve-se, a nosso ver, principalmente às restrições desse tipo

$$\sum_{i=1}^n y_i = 1, \quad (1)$$

onde $y_i \in \{0, 1\}$ para todo i : existem ramificações computacionalmente eficientes sobre restrições do tipo (veja o tópico *Generalized Upper Bound* em [5]).

3 Discussão e Conclusões

Diante dos testes e tendo em vista a realidade do Departamento de Matemática Aplicada da UFES, nossa adaptação do modelo proposto por MirHassani [4] apresenta-se como uma boa alternativa ao trabalho manual que hoje é empregado. Em relação ao trabalho de MirHassani, avançamos no equilíbrio da carga horária entre os docentes e na condensação dos horários de cada professor.

No entanto, uma limitação em nosso trabalho é que consideramos somente disciplinas de cargas horárias 30, 60 e 90 horas, que podem ser divididas em janelas de 2 horas na semana. Disciplinas de 45 e 75 horas não foram contempladas. Para tanto, devemos considerar janelas de tempo de diferentes tamanhos, e logo, a quantidade de variáveis e complexidade das restrições aumenta.

Outra questão é que não tratamos da alocação de salas de aula. O artigo proposto por Daskalaki e Birbas [2] propõe uma maneira interessante de contornar essa questão. Os autores resolvem o problema em duas fases. A primeira é similar ao presente trabalho, ou seja, aloca disciplinas para os professores e estabelece os horários das mesmas, levando em consideração somente a capacidade total de espaço físico. A segunda fase cuida da alocação de salas, onde tenta-se minimizar, dentre outras coisas, a variedade de salas de aula que uma mesma turma usa. É relatado neste trabalho que esta segunda fase é de fácil resolução frente à primeira. Acreditamos, portanto, ser fácil o uso dessa técnica em nossa proposta.

Referências Bibliográficas

- [1] E.K. Burke, S. Petrovic, Recent research directions in automated timetabling. *European Journal of Operational Research*, 140 (2002), 266-280.
- [2] S. Daskalaki, T. Birbas, Efficient solutions for a university timetabling problem through integer programming. *European Journal of Operational Research*, 160 (2005), 106-120.
- [3] T. Koch, “ZIMPL User Guide 3.2.0”, 2011.
- [4] S.A. MirHassani, A computational approach to enhancing course timetabling with integer programming, *Applied Mathematics and Computation*, 175 (2006), 814-822.
- [5] L.A. Wolsey, “Integer Programming”, Series in Discrete Mathematics and Optimization, John Wiley & Sons, 1998.