

## **PROBLEMA DE ALOCAÇÃO DE AULAS: O CASO DA CENTRAL DE AULAS DA UFPB**

**Eduardo Vieira Queiroga, Teobaldo Leite Bulhões Júnior,  
Lucídio dos Anjos Formiga Cabral**

Centro de Informática - Universidade Federal da Paraíba  
Campus V, Av. dos Escoteiros - Mangabeira, 58055-000, João Pessoa - PB  
{eduardo.queiroga, teobaldoleite, lucidio}@ci.ufpb.br

**Luciano Carlos Azevedo da Costa, Anand Subramanian**

Departamento de Engenharia de Produção - Centro de Tecnologia - UFPB  
Campus I, Bloco G, Cidade Universitária, 58051-970, João Pessoa, PB  
{lucianocosta, anand}@ct.ufpb.br

### **RESUMO**

O Problema de Alocação de Aulas a Salas (PAAS) consiste na alocação de aulas (com horários preestabelecidos) a salas, respeitando os conflitos por choques de horário e evitando que as capacidades das salas sejam excedidas. Para a resolução do problema, um modelo baseado em Programação Linear Inteira foi proposto. Esse modelo tem por objetivo minimizar o número de aulas de uma mesma turma alocadas em salas distintas. Devido às características da instância do problema, uma adaptação do modelo foi realizada. Buscando-se a redução do número de restrições implementadas no modelo, um algoritmo que associa o problema de listar as Cliques Maximais de um Grafo com a construção do conjunto de choque de horários foi utilizado. Os resultados mostram que, diferentemente do método manual atualmente utilizado, o modelo proposto realizou a alocação para o caso estudado em um baixo tempo computacional e respeitando ao máximo as exigências do problema.

**PALAVRAS CHAVE.** Alocação de Aulas, Otimização, Programação Linear Inteira.

**Área Principal:** PO na Educação (EDU)

### **ABSTRACT**

The Classroom Assignment Problem (CAP) consists in allocating courses (with schedules previously defined) to classrooms, avoiding the overlaps of courses schedules and respecting the classroom capacities. For solving this problem, a mathematical model based on Integer Linear Programming has been proposed. This model aims at minimizing the number of meetings of a same course assigned to different classrooms. Due to the instance features, an adaptation of this model has been performed. Seeking to reduce the number of implemented constraints in the model, an algorithm that associates the problem of listing the maximal cliques in a graph with the problem of construction the set of schedule overlaps has been used. The results show that, differently of the manual method currently used, the proposed model was able to make the assignment for the studied case spending a low computational time and respecting as much as possible the problem requirements.

**KEYWORDS.** Classroom Assignment, Optimization, Integer Linear Programming.

**Main Area:** OR in Education (EDU)

## 1. Introdução

O Problema de Alocação de Aulas a Salas (PAAS) consiste na alocação de aulas (com horário de início e término preestabelecidos) a salas, respeitando as suas capacidades (Schaerf, 1999). Esse tipo de problema pode ser encontrado nos mais diversos tipos de instituições de ensino, surgindo mais comumente nas de ensino superior, onde a complexidade deste problema pode aumentar, dado o elevado número de aulas a serem alocadas. Elloumi *et al.* (2014) provam a partir de uma redução do problema *NMTS* (do inglês, *Numerical Matching with Target Sum*), que o problema de alocação de aulas é  $\mathcal{NP}$ -Difícil.

Apesar da complexidade desse tipo de problema, tem-se utilizado modelos matemáticos para a sua resolução. Segundo Phillips *et al.* (2015), muitas formulações matemáticas são capazes de considerar diversos critérios de avaliação da solução, destacando-se: maximização do número de aulas alocadas; maximização do número de horas/aula cursadas pelos alunos; maximização da utilização das salas; maximização das preferências associadas à alocação de um dado curso a uma sala; minimização do número de aulas de um mesmo curso alocadas em salas diferentes (estabilidade aula-sala); e maximização da robustez da alocação.

Mesmo que a utilização de modelos matemáticos não seja aconselhada para a resolução de problemas de alocação de grande escala, dependendo de como eles são implementados e/ou concebidos, eles podem ser utilizadas na prática. Uma das principais vantagens da utilização desse tipo de abordagem diz respeito a simplicidade de sua execução. Ao contrário do que acontece na implementação de métodos aproximativos (heurísticas / meta-heurísticas), não se faz necessária a utilização de estruturas de dados complexas e de técnicas de programação avançadas.

Diante do exposto, este trabalho tem por objetivo utilizar um modelo de programação inteira mista para resolver o problema de alocação de aulas na Central de Aulas (CA) da Universidade Federal da Paraíba. Atualmente, essa tarefa é realizada manualmente. Além consumir diversas horas de trabalho dos responsáveis por essa atividade, no geral, as soluções obtidas são sub-ótimas e repletas de inconsistências, como: choques de horário entre as aulas alocadas; desconsideração das capacidades das salas; alocação de mais de uma aula na mesma sala e no mesmo horário, entre outros.

O restante deste artigo está organizado como segue: a Seção 2 apresenta e discute alguns trabalhos realizados no contexto do Problema de Alocação de Aulas; na Seção 3, é feita uma descrição do problema e são apresentadas suas principais características e peculiaridades. Na Seção 4, é apresentada uma formulação de programação inteira mista que permite a resolução do Problema de Alocação de Salas. Ainda nessa seção, uma formulação alternativa é discutida. A Seção 5 apresenta e discute os principais resultados obtidos através da aplicação dos modelos utilizados. As conclusões e propostas de trabalhos futuros são apontadas na Seção 6.

## 2. Trabalhos Relacionados

Muitos trabalhos na literatura têm proposto heurísticas para a resolução do PAAS. No trabalho de Martinez-Alfaro e Flores-Terán (1998), é proposto um método de resolução baseado em *Simulated Annealing* para a determinação da alocação de salas em uma instituição de ensino superior. O algoritmo considerou aspectos como: capacidade das salas com relação às quantidades de alunos em cada turma, distância entre as salas e os departamentos aos quais pertenciam as disciplinas, duração das aulas e choques de horário.

Nos métodos apresentados em Constantino *et al.* (2010) e Kripka *et al.* (2011), busca-se realizar a alocação de aulas de modo que os descolamentos dos alunos sejam minimizados. Enquanto Constantino *et al.* (2010) propõem e estudam três algoritmos para a resolução do problema, Kripka *et al.* (2011) propõem um método baseado em um *Simulated Annealing*.

Subramanian *et al.* (2011) e Prado e Souza (2014) propõem algoritmos que consideram penalizações decorrentes do não atendimento das restrições do problema e dos requisitos de qualidade das soluções. O primeiro trabalho utiliza um método baseado em Busca Tabu, ao passo que o

segundo propõe um algoritmo heurístico com uma construção gulosa e uma meta-heurística baseada em *Simulated Annealing* como mecanismo de Busca Local.

No problema de alocação de aulas, os horários de início e término das aulas devem ser conhecidos *a priori*. Dessa forma, muito comumente esse problema pode ser resolvido em conjunto com o problema de *timetabling*. Os métodos que englobam esses dois problemas seguem uma estratégia de “*times first, rooms second*”, i.e., primeiro define-se os horários das aulas, para então realizar-se a alocação das aulas (Phillips *et al.*, 2015). Um exemplo dessa abordagem é encontrada no trabalho de Ueda *et al.* (2001), que propõe um Algoritmo Genético de duas fases para resolver ambos os problemas em sequência.

Apesar da grande utilização de métodos aproximativos, alguns trabalhos têm utilizado uma abordagem baseada em modelos matemáticos para a resolução do PAAS. Dammak *et al.* (2006) apresentam formulações para o problema de alocação de exames a salas. Nesse trabalho, são apresentados dois modelos lineares inteiros. No primeiro, cada sala não pode conter mais de um exame alocado. O segundo modelo possui essa restrição relaxada e foi formulado como um problema de transporte.

Cirino *et al.* (2013) resolvem o problema de alocação de aulas no ICMC da USP. Os autores propuseram um modelo cuja função objetivo, a ser minimizada, foi baseada em dois critérios: evitar que as aulas de uma mesma turma sejam ministradas em salas diferentes; e maximizar a ocupação relativa das salas, isto é, uma aula deve acontecer em uma sala cuja capacidade seja próxima da demanda da aula.

No trabalho de Phillips *et al.* (2015), é apresentado um modelo de programação inteira baseado no problema do máximo empacotamento de conjuntos. Essa formulação permite que instâncias de tamanho elevado sejam resolvidas. Diferentemente da maioria das formulações propostas, nesse trabalho, as variáveis de decisão representam padrões de alocação. O problema é resolvido utilizando-se um algoritmo de otimização lexicográfica, onde se busca ao máximo o atendimento dos critérios de avaliação propostos. Dentre os indicadores propostos no trabalho, destaca-se a Estabilidade Aulas-Sala, que visa reduzir o número de aulas de uma turma alocadas em salas diferentes.

### 3. Descrição do Problema

O problema descrito nesta seção consiste na alocação de aulas no prédio da Central de Aulas da UFPB (CA-UFPB). A CA é um setor da UFPB formado por diversos blocos de salas que recebem aulas dos diversos Centros da universidade.

Na UFPB, em boa parte dos casos, a alocação das aulas é feita em duas etapas: em um primeiro momento, os gestores de cada centro tentam alocar suas demandas de aulas utilizando as instalações (salas de aula/laboratórios) que lhes são disponíveis. Caso a quantidade de aulas a serem alocadas exceda o número de salas disponíveis em cada centro, a demanda excedente é encaminhada para ser alocada na Central de Aulas. Assim, no geral, as salas da CA recebem a demanda de aulas excedente que não pôde ser alocada nas instalações dos seus respectivos centros de origem.

O estudo de caso em questão foi realizado durante o semestre letivo 2014.2. O número de turmas presentes na instância do problema foi 645, representando uma demanda em aulas de 1131 aulas. O número de salas disponíveis na CA no período em questão foi 60 salas, cujas capacidades, em número de alunos, são apresentadas na Tabela 1.

Antes da apresentação das informações relacionadas à disponibilidade e utilização das salas, apresenta-se o conceito de horas-aula (HA). Este se aplica pois nem sempre uma aula tem duração de 1 hora, como é o caso das aulas no turno da noite, que possuem duração de 50 minutos. O número de HA disponíveis para cada sala foi definido segundo os horários de término e início de cada turno: Manhã (07:00 - 12:00) = 5 horas-aula; Tarde (12:00 - 18:00) = 6 horas-aula; Noite (18:00 - 22:20) = 5 horas aula.

Tabela 1: Distribuição das salas por bloco na CA-UFPB, com suas respectivas capacidades

Bloco A		Bloco B		Bloco D		Bloco E		Bloco F		Bloco G		Bloco H	
Sala	Cap.	Sala	Cap.	Sala	Cap.	Sala	Cap.	Sala	Cap.	Sala	Cap.	Sala	Cap.
CAA101	60	CAB101	60	CAD101	40	CAE101	40	CAF101	40	CAG101	40	CAH101	40
CAA102	60	CAB102	60	CAD102	40	CAE102	40	CAF102	40	CAG102	40	CAH102	40
CAA103	60	CAB103	60	CAD103	40	CAE103	40	CAF103	40	CAG103	40	CAH103	40
CAA104	60	CAB104	60	CAD104	40	CAE104	40	CAF104	40	CAG104	40	CAH104	40
CAA105	80	CAB105	40	CAD105	40	CAE105	40	CAF105	40	CAG105	40	CAH105	40
CAA106	80	CAB106	60	CAD106	40	CAE106	40	CAF106	40	CAG106	60	CAH106	40
		CAB107	50	CAD107	40	CAE107	40	CAF107	40			CAH107	40
				CAD108	50	CAE108	50	CAF108	50			CAH108	50
				CAD109	30	CAE109	30	CAF109	30				
				CAD110	60	CAE110	60	CAF110	60				
				CAD111	60	CAE111	60	CAF111	60				

As Tabelas 2 e 3 apresentam a taxa de ocupação e a disponibilidade das salas de aula da CA-UFPB, por turno e por turno/dia da semana, respectivamente. Essa taxa foi calculada dividindo-se a quantidade de horas-aula a serem alocadas no turno pela disponibilidade das salas, que também foi calculada em horas-aula. Esta última foi obtida multiplicando-se o número de salas disponíveis pelo número de horas-aula disponíveis para cada sala.

Tabela 2: Disponibilidade e Taxa de Ocupação por turno

Turno	Disponibilidade	Demanda	% Porcentagem
Manhã (07:00 - 12:00)	1500	995	66.33
Tarde (12:00 - 18:00)	1800	873	46.50
Noite (18:00 - 22:20)	1500	870	58.00

Tabela 3: Disponibilidade e Taxa de Ocupação por turno e dia

Dia	Manhã	% Ocupação	Tarde	% Ocupação	Noite	% Ocupação
Segunda	300	76.33	360	54.44	300	65.00
Terça	300	63.33	360	44.17	300	52.33
Quarta	300	77.33	360	57.22	300	59.78
Quinta	300	60.33	360	43.89	300	56.67
Sexta	300	54.33	360	32.78	300	56.22

Foi observado durante o cálculo da demanda horária por turno que a duração de algumas aulas transpassava o horário de término do turno. Foram encontrados os seguintes casos de início e término das aulas: 09:00 - 13:00; 11:00 - 13:00; 16:00 - 19:00; 17:00 - 19:00. Nesses casos, no cálculo da demanda de cada turno, foram consideradas apenas as horas de aula no respectivo turno. No caso das aulas com início às 9h e término às 13h, 3 horas foram computadas na carga horária do turno manhã e 1 hora no turno da tarde.

Uma característica da instância que torna a tentativa de se encontrar uma solução viável para o problema uma tarefa árdua, diz respeito à grande quantidade de aulas em um mesmo horário. Como o número de aulas é muito maior do que o de salas, logo que são consideradas todas as restrições do problema (apresentadas no final da seção), muitas vezes uma solução viável não pode ser encontrada.

O problema descrito poderia ser evitado através da utilização de um modelo de *timetabling* para a determinação dos horários de início e término de cada aula. Como apresentado na seção anterior, em diversos problemas da literatura, os problemas de alocação de salas e determinação de horários são resolvidos em conjunto. Vale ressaltar, contudo, que no caso estudado, essa prática não seria viável, a menos que houvesse um compartilhamento de informações entre os diversos centros na definição dos horários das disciplinas. Isso demandaria a integração das decisões de alocação, inclusive na primeira fase do processo. Embora o procedimento descrito represente ganhos consideráveis do ponto de vista da gestão de recursos, sabe-se que os horários das aulas sempre acabam

sendo definidos de acordo com as preferências das pessoas envolvidas no processo de tomada de decisão.

A Tabela 4 apresenta a quantidade de aulas por dia da semana e horário. Pode ser observada uma grande concentração de aulas nas segundas e quartas-feiras, nos horários de 10:00 – 12:00, 14:00 – 16:00 e 19:00 – 20:40.

Tabela 4: Quantidade de aulas por dia e horário

Horário Turno																
	07:00 – 08:00	08:00 – 09:00	09:00 – 10:00	10:00 – 11:00	11:00 – 12:00	12:00 – 13:00	13:00 – 14:00	14:00 – 15:00	15:00 – 16:00	16:00 – 17:00	17:00 – 18:00	18:00 – 19:00	19:00 – 19:50	19:50 – 20:40	20:40 – 21:30	21:30 – 22:20
Segunda-Feira	20	53	44	56	56	5	18	50	48	40	35	10	58	58	53	53
Terça-Feira	14	49	48	42	37	5	15	34	35	37	33	12	45	45	42	42
Quarta-Feira	14	51	53	58	56	5	18	52	50	42	39	11	52	52	49	49
Quinta-Feira	13	43	46	42	37	5	14	35	35	36	33	15	47	47	46	46
Sexta-Feira	14	38	41	38	32	8	11	22	28	25	24	7	47	47	50	50

Diante da problemática exposta, as principais restrições do problema em estudo podem ser sintetizadas como segue:

- Todas as aulas devem ser alocadas a uma sala em um determinado horário;
- Duas aulas não podem ser alocadas a uma mesma sala em um mesmo horário/dia da semana;
- A quantidade de alunos em uma turma não pode exceder a capacidade da sala (restrição pode ser relaxada dependendo da situação).
- O indicador de qualidade considerado para análise da solução foi o número de aulas de uma mesma turma alocadas em salas diferentes.

#### 4. Formulação Matemática

Esta seção apresenta e discute duas formulações para o PAAS. A primeira considera todas as restrições do problema apresentadas na Seção 3. A segunda formulação, por sua vez, propõe uma adaptação da primeira formulação, relaxando as restrições de capacidade das salas. Essa adaptação tem por objetivo tornar possível a obtenção de soluções viáveis. Como apresentado na Tabela 4, o número de aulas em cada *slot* da grade horária não é balanceado. Além disso, para algumas turmas, a demanda de alunos é superior à capacidade de qualquer sala disponível. Dessa forma, dadas as limitações de recurso, encontrar uma solução viável pode não ser possível.

A escolha da restrição para ser relaxada se deu por três razões:

- Em conversa com os gestores, percebeu-se que esta seria a restrição cujo não atendimento levaria a um menor transtorno. Além de o número de alunos matriculados em cada turma poder variar (a alocação é realizada antes do término das matrículas), pequenas adaptações como a adição e/ou retirada de cadeiras poderiam ser facilmente realizadas;
- Essa prática também é adotada pelos gestores durante a realização da alocação manual;
- Na própria instância, existem turmas que apresentam uma demanda superior à capacidade de todas as salas disponíveis, que é de 80 alunos.

##### 4.1. Modelo Proposto (Formulação F1)

Diante da problemática exposta, é apresentado abaixo um modelo de Programação Linear Inteira para a resolução do PAAS:

##### Notação

### Conjuntos

$T$	Conjunto das turmas $t$ a serem alocadas
$H_t$	Conjunto das aulas (horários) $i$ de cada turma $t$
$S$	Conjunto das salas $j$ disponíveis
$Ch$	Conjunto dos choques de horário $c$ entre as aulas
$H_c$	Conjunto das aulas (horários) $k$ do choque de horário $c$

### Dados de Entrada

$Q_j$	Capacidade da sala $j$
$D_t$	Demanda da turma $t$

### Variáveis de Decisão

$$x_{ij} = \begin{cases} 1, & \text{se a aula } i \text{ for alocada na sala } j \\ 0, & \text{caso contrário} \end{cases}$$

$$y_t = \begin{cases} 1, & \text{se as aulas da turma } t \text{ não forem alocadas na mesma sala} \\ 0, & \text{caso contrário} \end{cases}$$

### Modelo

$$\begin{aligned} \min \sum_{t \in T} y_t & \quad (1) \\ \text{s.a. } \sum_{j \in S} x_{ij} = 1 & \quad \forall i \in H_t, t \in T \quad (2) \\ \sum_{k \in H_c} x_{kj} \leq 1 & \quad \forall j \in S, \forall c \in Ch \quad (3) \\ x_{ij} D_t \leq Q_j & \quad \forall j \in S, \forall i \in H_t, t \in T \quad (4) \\ y_t \geq |x_{ij} - x_{kj}| & \quad \forall t \in T, \forall i, k \in H_t, \forall j \in S \quad (5) \\ x_{ij} \in \{0, 1\} & \quad \forall i \in P, \forall j \in T_i \quad (6) \\ y_t \in \{0, 1\} & \quad \forall t \in T \quad (7) \end{aligned}$$

A função objetivo (1) visa minimizar o número de aulas de uma mesma turma alocadas em salas diferentes. As restrições (2) obrigam que todas as aulas de todas as turmas sejam alocadas a uma única sala. Os choques de horário são evitados por meio das restrições (3), que impedem que duas ou mais aulas sejam alocadas em uma mesma sala no mesmo dia e horário. As capacidades das salas são respeitadas através das restrições (4) e as restrições (5), além de relacionar as variáveis em consideração, fazem a contagem das turmas cujas aulas são alocadas em salas distintas. Por fim, as restrições (6) e (7) definem a natureza das variáveis.

#### 4.2. Modelo Ajustado para o Caso Estudado (Formulação F2)

A utilização da formulação F1 para o caso estudado neste trabalho leva a uma solução inviável. Como já descrito, na instância considerada, o atendimento de todas as restrições impostas ao modelo não é possível.

Na alocação manual utilizada atualmente na UFPB, é comum se encontrar problemas relacionados ao não atendimento de algumas restrições. Alocação de múltiplas aulas em uma mesma sala, estouro da capacidade das salas ou mesmo a não alocação de algumas turmas são algumas das inconsistências que podem ser encontradas.

Dado que a instituição aceita durante o processo de alocação de aulas a violação de algumas restrições do problema, uma modificação na Formulação F1 é proposta. Essa adaptação diz respeito à relaxação das restrições de capacidade, penalizando-se na função objetivo o seu não atendimento.

*Penalidades*

$$p_{ij} = \begin{cases} D_t - Q_j, & \text{se } D_t - Q_j > 0 \\ 0, & \text{caso contrário.} \end{cases}$$

**Modelo**

$$\min \sum_{t \in T} y_t + \sum_{i \in H_t} \sum_{t \in T} \sum_{j \in S} p_{ij} * x_{ij} \quad (8)$$

$$\text{s.a } \sum_{j \in S} x_{ij} = 1 \quad \forall i \in H_t | t \in T \quad (9)$$

$$\sum_{k \in H_c} x_{kj} \leq 1 \quad \forall j \in S, \forall c \in Ch \quad (10)$$

$$y_t \geq |x_{ij} - x_{kj}| \quad \forall t \in T | \forall i, k \in H_t, \forall j \in S \quad (11)$$

$$x_{ij} \in \{0, 1\} \quad \forall i \in P, \forall j \in T_i \quad (12)$$

$$y_t \in \{0, 1\} \quad \forall t \in T \quad (13)$$

Quando uma aula  $i$  for alocada em uma sala  $j$  de capacidade insuficiente, uma penalidade  $p_{ij}$  será adicionada à função objetivo. Essa penalidade é proporcional ao excesso provocado. Embora essa nova formulação permita que soluções “inviáveis” sejam geradas, como será visto nos resultados, o não atendimento das restrições de capacidade ocorreu em poucos casos. Além disso, como será visto nos resultados, a formulação proposta também foi capaz de aumentar a taxa de utilização das salas, isto é, as aulas foram alocadas, na maioria das vezes, a salas com valores de capacidade próximos dos valores das demandas.

## 5. Resultados Computacionais e Análise Comparativa

O modelo proposto foi implementado no CPLEX 12.6, utilizando como *framework* a linguagem C++. Os testes foram realizados em um computador com processador Intel Core i5-3210M CPU 2.5 GHz, 8 GB de memória RAM, sob o sistema operacional Linux Mint 17 de 64 bits.

O modelo executado possui restrições de capacidade relaxadas e penalizadas na função objetivo, como foi explicado na seção 4.2; já a alocação manual foi realizada por um funcionário da UFPB.

### 5.1. Algoritmo de detecção de choques de horários

O conjunto de choques de horário  $Ch$  presente nas formulações descritas na Seção 4 é formado conforme apresentado no Algoritmo 1. Cada elemento do conjunto é formado por todas as turmas existentes que possuem algum choque de horário. Uma vez que esse conjunto tenha sido gerado, as restrições de choque de horário são geradas. Deve-se notar que, diferentemente do que é feito em alguns trabalhos na literatura, os choques de horário não são computados para cada par de turmas. Todas as turmas que possuem algum tipo de choque de horário são colocadas em uma mesma restrição. Essa escolha busca reduzir a quantidade de restrições necessárias para tratar os choques de horário, possivelmente melhorando o desempenho do modelo matemático.

O algoritmo que computa os choques de horário baseia-se na ideia de construção dos conjuntos de choques maximais, podendo ser comparado ao problema de se listar todas as Cliques Maximais de um Grafo, definido em Bron e Kerbosch (1973). A Figura 1 exemplifica essa analogia para um grafo em que a aresta  $\{v_i, v_j\}$  indica que a aula  $i$  (vértice  $i$ ) possui choque com a aula  $j$  (vértice  $j$ ). Nesse caso, existem dois choques de horário (Figura 1(a)) em correspondência com os cliques maximais do grafo (Figura 1(b)). Esta visão demonstra a eficiência do algoritmo em reduzir ao máximo o número de restrições de choque utilizados pelo modelo matemático proposto.

No Algoritmo 1, os conjuntos  $ch$  e *choques* são inicializados. O algoritmo adiciona ao conjunto de choques todas as turmas que iniciam no horário indicado (Linha 9). Na primeira

---

**Algoritmo 1:** Algoritmo de Detecção de Choques Maximais - ADCM
 

---

```

1 Procedimento ADCM ()
2   aulas : conjunto de aulas a ser analisado
3   horarios : conjunto de horários relacionados com as aulas (considerando início
4     e término indistintamente) ordenado cronologicamente e sem repetição (como
5     no exemplo da Figura 1(a))
6   ch  $\leftarrow \emptyset$  : choque de horário em construção
7   choques  $\leftarrow \emptyset$  : conjunto global de todos os choques encontrados
8   chAtivo  $\leftarrow false$  : indica se ch é ativo e pode ser adicionado a choques
9   chAtivo  $\leftarrow true$ 
10  As funções aulasi(h) e aulasf(h) retornam as aulas com início e fim no
11  horário h, respectivamente
12  para cada h  $\in$  horarios fazer
13    se chAtivo == false então
14       $\lfloor$  Vá para linha 19
15    se  $|aulas_f(h)| > 0$  então
16      se chAtivo == true &&  $|ch| > 1$  então
17         $\lfloor$  choques.add(ch); // guarda choque maximal encontrado
18        remova aulasf(h) de ch;
19        chAtivo = false;
20    se  $|aulas_i(h)| > 0$  então
21      ch.add(aulasi(h)); // adiciona aulas que começam em h ao choque em
22      construção chAtivo = true;
23  retorne choques;
  
```

---

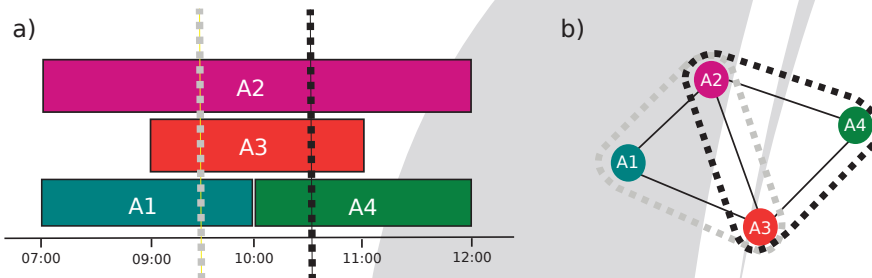


Figura 1: a). Choques de horários em visão cronológica. b) Choques de horários como cliques maximais em Grafo.

iteração do laço, a variável  $h$  corresponde ao horário de início de uma ou mais aulas. Caso a condição da Linha 17 seja satisfeita, as aulas que tem início no horário  $h$  são adicionadas ao conjunto  $ch$ , que corresponde ao conjunto dos choques em construção, e a variável  $chAtivo$  recebe o valor  $true$ , que indica que o conjunto ainda pode receber elementos (conjunto segue ativo). Nas iterações seguintes, para os próximos horários na sequência,  $h$  pode ser também o horário de término de uma ou mais aulas no conjunto de choque (Linha 12). Caso a condição na linha 12 seja satisfeita, a linha 13 verifica se o conjunto ainda está ativo e se possui mais de um elemento. Caso a condição na linha 13 seja satisfeita, o conjunto em construção é considerado maximal e o conjunto de choques  $ch$  (choques correntes) é armazenado ao conjunto  $choques$ . Cada um dos elementos em  $choques$  representará uma restrição do modelo. As aulas que possuem término no  $h$  corrente são



removidas do conjunto  $ch$ . O algoritmo continua verificando se existem turmas aptas a entrarem no conjunto  $ch$ . O algoritmo termina quando no teste da Linha 10 o conjunto  $ch$  não estiver mais ativo.

No exemplo da Figura 1, o algoritmo inicia a análise no horário 07:00 e então adiciona ao conjunto  $ch$  as aulas A1 e A2 e ativa a variável  $chAtivo$ . No horário 09:00 a aula A3 é adicionada ao conjunto que se torna  $ch = \{A1, A2, A3\}$ . No horário 10:00 a aula A1 termina e o conjunto  $ch$  é adicionado ao conjunto global de *choques*. A1 é então removido  $ch$  e A4 é adicionado, resultando em  $ch = \{A2, A3, A4\}$ . A variável  $chAtivo$  que recebeu *false* quando ocorreu a remoção de A1, recebe *true*, reativando o conjunto  $ch$ . No horário 11:00, o conjunto de choque  $ch$  é adicionado a *choques* pois A3 termina, e assim  $chAtivo$  recebe *false*. A menos que uma nova aula seja adicionada nessa iteração,  $chAtivo$  permanece inativo. Para o último horário 12:00 o conjunto  $ch$  atual que é  $ch = \{A2, A4\}$  permanece inativo. Assim, no teste da Linha 10, o algoritmo termina. O algoritmo retorna então o conjunto  $choques = \{\{A1, A2, A3\}, \{A2, A3, A4\}\}$ , com dois choques maximais.

## 5.2. Análise dos Resultados

Nas Tabelas 5, 6 e 7 apresentam-se exemplos da alocação realizada pelo modelo para salas com capacidades de 30, 60 e 80 alunos, respectivamente. Nessas tabelas, as turmas são representadas pelo seu código em um *slot* da grade de horários durante a semana.

Tabela 5: Alocação da sala CAE-109.

Horário \ Dia	Seg	Ter	Qua	Qui	Sex
07:00 às 08:00	–	–	–	–	1404256/01
08:00 às 09:00	1404256/01	2201017/01	1404304/01	2201017/01	1404256/01
09:00 às 10:00	1404256/01	2201017/01	1404304/01	2201017/01	2203005/01
10:00 às 11:00	2203014/01	–	1404304/01	–	2203005/01
11:00 às 12:00	2203014/01	–	1404304/01	–	2203005/01
12:00 às 13:00	–	–	–	–	–
13:00 às 14:00	–	–	2203014/01	2203005/01	–
14:00 às 15:00	–	–	2203014/01	2203005/01	–
15:00 às 16:00	–	–	–	2203005/01	–
16:00 às 17:00	–	–	–	–	–
17:00 às 18:00	–	1404004/11	–	1404004/11	–
18:00 às 19:00	–	1404004/11	–	1404004/11	–
19:00 às 20:40	1403331/02	1403327/02	1302372/03	1302298/03	1302298/03
20:40 às 22:20	1403331/02	1403327/02	1302372/03	–	–

Uma comparação entre a alocação realizada pelo modelo matemático e a alocação manual pode ser vista na Tabela 8. A alocação automática, assim como na alocação manual, alocou todas as aulas, o que já era previsto devido a obrigatoriedade dessa restrição no modelo. Na violação I, que diz respeito ao número de turmas alocadas em salas distintas, o modelo automático se mostrou bem mais eficiente, pois descumpriu apenas uma vez este requisito. A alocação manual, por sua vez, violou esse requisito 58 vezes. Na violação II, relacionado às aulas alocadas em uma sala com capacidade insuficiente, a alocação automática apresentou um melhor desempenho quando comparada à alocação manual. No que diz respeito aos choques de horário, a alocação automática não violou nenhuma restrição de choques de horário (violação III), o que não pode ser visto no resultado da alocação manual. Na alocação manual, 23 choques de horário foram observados.

Tabela 6: Alocação da sala CAA-101.

<b>Dia</b> <b>Horário</b>	<b>Seg</b>	<b>Ter</b>	<b>Qua</b>	<b>Qui</b>	<b>Sex</b>
07:00 às 08:00	1403182/01	–	1403182/01	–	1401134/03
08:00 às 09:00	1403182/01	1108100/02	1403182/01	1103118/10	1401134/03
09:00 às 10:00	1401134/03	1108100/02	–	1103118/10	1401134/03
10:00 às 11:00	1101164/04	1108104/01	1101164/04	–	–
11:00 às 12:00	1101164/04	1108104/01	1101164/04	–	–
12:00 às 13:00	–	–	–	–	–
13:00 às 14:00	–	–	–	–	–
14:00 às 15:00	1101157/14	1103118/12	1101157/14	1103118/12	–
15:00 às 16:00	1101157/14	1103118/12	1101157/14	1103118/12	1401348/32
16:00 às 17:00	1103178/10	1107174/02	–	1107174/02	1401348/32
17:00 às 18:00	1103178/10	1107174/02	–	1107174/02	1401348/32
18:00 às 19:00	–	–	–	–	–
19:00 às 20:40	1101157/16	1205020/01	1205020/01	1403747/03	1401134/05
20:40 às 22:20	–	–	1205020/01	1403747/03	1401134/05

Tabela 7: Alocação da sala CAA-106.

<b>Dia</b> <b>Horário</b>	<b>Seg</b>	<b>Ter</b>	<b>Qua</b>	<b>Qui</b>	<b>Sex</b>
07:00 às 08:00	–	–	–	–	–
08:00 às 09:00	1103178/05	1103179/05	1103178/05	1103179/05	–
09:00 às 10:00	1103178/05	1103179/05	1103178/05	1103179/05	1404289/01
10:00 às 11:00	1404138/01	1403153/02	1404138/01	1403153/02	1404289/01
11:00 às 12:00	1404138/01	1403153/02	1404138/01	1403153/02	1404289/01
12:00 às 13:00	–	–	–	–	1404289/01
13:00 às 14:00	–	–	–	–	–
14:00 às 15:00	1101157/10	1103118/13	1101157/10	1103118/13	–
15:00 às 16:00	1101157/10	1103118/13	1101157/10	1103118/13	–
16:00 às 17:00	1103232/06	1403153/04	1103232/06	1403153/04	–
17:00 às 18:00	1103232/06	1403153/04	1103232/06	1403153/04	–
18:00 às 19:00	–	–	–	–	–
19:00 às 20:40	1404138/03	1403153/07	1101157/16	1403153/07	1204233/02
20:40 às 22:20	1404138/03	1204138/02	1103227/02	1204138/02	1103227/02

- Violação I: Número de turmas com aulas alocadas em salas diferentes
- Violação II: Número de aulas com estouro da capacidade
- Violação III: Número de choques de horário entre duas ou mais aulas

O somatório dos estouros é uma métrica que de modo geral indica o nível de violação com respeito à capacidade das salas. O modelo matemático também apresentou um melhor resultado para esse indicador. Ele apresentou, aproximadamente, um valor 5 vezes menor de estouros do que a alocação manual. Uma outra perspectiva da violação de capacidade pode ser vista na Figura 2. O estouro máximo apresentado pela alocação manual foi consideravelmente maior do que o obtido pela alocação automática.

A alocação automática foi realizada em apenas alguns segundos, contra um tempo de alocação desconhecido para a alocação manual devido a complexidade da tarefa.

Tabela 8: Comparação Alocação Manual x Alocação Automática

Método	Aulas Alocadas	Aulas Não Alocadas	I	II	III	Somatório dos Estouros	Tempo de Alocação
Manual	1131	0	58	213	23	2432	Desconhecido
Automático	1131	0	1	81	0	552	15 – 20 segundos

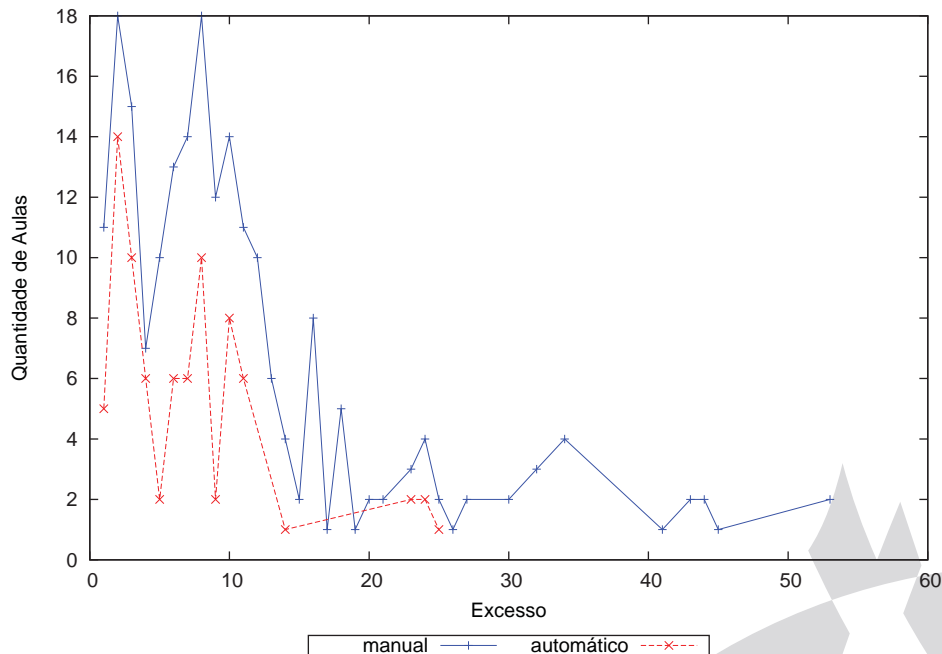


Figura 2: Histograma de estouro de capacidade na solução manual e automática.

## 6. Conclusões

Neste trabalho foi proposto um modelo matemático de Programação Linear Inteira para o Problema de Alocação de Aulas a Salas (PAAS) com objetivo de minimizar o número de turmas com aulas alocadas em salas distintas. Um algoritmo de definição de conjuntos de choques de horário maximais também foi proposto para minimizar o número de restrições no modelo. Foi utilizada uma estratégia de penalização para o controle das violações das capacidades das salas.

O modelo matemático se mostrou bastante eficiente na resolução do PAAS. Quando comparado com os resultados obtidos pela alocação manual, a obtida pela alocação automática apresentou poucas inconsistências, pois: proibiu a existência de choques de horário; reduziu o estouro das capacidades existentes; fez uma melhor utilização da capacidade das salas; além de respeitar o critério considerado, isto é, alocar em uma mesma sala aulas de uma mesma turma. Além disso, a alocação automática pôde ser realizada em poucos segundos, diferente do que é observado para uma alocação manual dessa natureza.

Como trabalhos futuros, sugere-se a utilização de novos objetivos e configurações, buscando uma melhor alocação. Essas novas configurações podem englobar outros critérios não abordados neste trabalho. Além disso, métodos heurísticos podem ser aplicados com a intuito de resolver instâncias maiores dentro da UFPB.

## Referências

**Bron, C. e Kerbosch, J.** (1973), Algorithm 457: Finding all cliques of an undirected graph. *Commun. ACM*, v. 16, n. 9, p. 575–577.

- Cirino, R. B. Z., Costa, A. M. e Santos, M. O.** Um modelo matemático para resolução do problema de alocação de salas no Instituto de Ciências Matemáticas e de Computação da Universidade de São Paulo. *XLV Simpósio Brasileiro de Pesquisa Operacional*, p. 2555–2565, 2013.
- Constantino, A. A., Filho, W. M. e Landa-Silva, D.** Iterated heuristic algorithms for the classroom assignment problem. *Proceedings of the: 2010 International Conference on the Practice and Theory of Automated Timetabling (PATAT 2010)*, p. 152–166, 2010.
- Dammak, A., Elloumi, A. e Kamoun, H.** (2006), Classroom assignment for exam timetabling. *Advances in Engineering Software*, v. 37, n. 10, p. 659–666.
- Elloumi, A., Kamoun, H., Jarboui, B. e Dammak, A.** (2014), The classroom assignment problem: Complexity, size reduction and heuristics. *Applied Soft Computing*, v. 14, p. 677–686.
- Kripka, R. M. L., Kripka, M. e Silva, M. C.** Formulação para o problema de alocação de salas de aula com minimização de deslocamentos. *XLIII Simpósio Brasileiro de Pesquisa Operacional*, p. 1941–1951, 2011.
- Martinez-Alfaro, H. e Flores-Terán, G.** (1998), Solving the Classroom Assignment Problem With Simulated Annealing. p. 3703–3708.
- Phillips, A. E., Waterer, H., Ehrgott, M. e Ryan, D. M.** (2015), Integer programming methods for large-scale practical classroom assignment problems. *Computational & Operations Research*, n. 53, p. 42–53.
- Prado, A. S. e Souza, S. R.** Problema de Alocação de Salas em Cursos Universitários: Um Estudo de Caso. *XLVI Simpósio Brasileiro de Pesquisa Operacional*, p. 2054–2065, 2014.
- Schaerf, A.** (1999), A Survey of Automated Timetabling. *Artificial Intelligence Review*, v. 13, p. 87 – 127.
- Subramanian, A., Medeiros, J. M. F., Formiga, L. d. A. e Souza, M. J. F.** (2011), Alocação de aulas a salas em uma instituição universitária. *Revista Produção Online*, v. 11, n. 1, p. 54–75.
- Ueda, H., Ouchi, D., Takahashi, K. e Miyahara, T.** A co-evolving timeslot/room assignment genetic algorithm technique for university timetabling. Burke, E. e Erben, W. (Eds.), *Practice and Theory of Automated Timetabling III*, volume 2079 of *Lecture Notes in Computer Science*, p. 48–63. Springer Berlin Heidelberg, 2001.