

# PROBLEMA DE ATRIBUIÇÃO DE VIGILÂNCIAS DE EXAMES A DOCENTES

João Paulo das Neves Silva



Mestrado em Gestão de Processos e Operações

Departamento de Engenharia Mecânica

Instituto Superior de Engenharia do Porto

2012

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de Tese/Dissertação, do 2º ano, do Mestrado em Gestão de Processos e Operações

Candidato: João Paulo das Neves Silva, N° 1100123, jns@eu.ipp.pt

Orientação científica: Professor Doutor Manuel Pereira Lopes, mpl@isep.ipp.pt



Mestrado em Gestão de Processos e Operações

Departamento de Engenharia Mecânica

Instituto Superior de Engenharia do Porto

Novembro 2012

À família e aos amigos.



## *Agradecimentos*

Quero agradecer, em primeiro lugar, ao Instituto Superior de Engenharia do Porto e ao meu orientador, o professor Manuel Pereira Lopes com quem tive o privilégio de aprender muitos ensinamentos durante o período de mestrado bem como durante a elaboração deste trabalho.

Aos professores João Bastos e João Pinho Ribeiro, pela disponibilidade e ajuda prestada.

Aos amigos que me apoiaram e a todos os que direta ou indiretamente contribuíram para o desenvolvimento do meu trabalho.

## *Resumo*

O presente trabalho foi realizado com o intuito de resolver o problema de alocação de vigilantes a exames do Instituto Superior de Engenharia do Porto, no departamento de Engenharia Mecânica.

O modelo apresentado faz a atribuição das vigilâncias de uma forma hierárquica, utilizando vários critérios, desde a regência da unidade curricular até à simples vigilância.

Devido ao facto de estar implementado informaticamente, apresenta reduzidos tempos na formulação e obtenção de uma solução, o que o torna uma boa ferramenta para a criação de cenários alternativos.

Em suma, o modelo proposto neste trabalho apresenta soluções de melhor qualidade, em que a distribuição de afetações é proporcional entre os docentes, e o seu tempo de obtenção é muito reduzido em comparação com a alternativa atual.

### *Palavras-Chave*

Problemas de horários, programação inteira binária, alocação de docentes a exames.

## *Abstract*

The present study was performed in order to solve the problem of allocating vigilantes to examinations in the Instituto Superior de Engenharia do Porto, in the department of Mechanical Engineering.

The model presented makes the allocation of surveillances in a hierarchical manner, using various criteria, since the regency of course to the simple vigilance.

Due to it being implemented on a computer, the model presents reduced time in formulating and obtaining a solution, which makes it a good tool for the creation of alternative scenarios.

In sum, the proposed model presents better quality solutions, in which the distribution of the affectations is equitable among teachers, and require much less time to obtain compared to the current alternative.

### ***Keywords***

Scheduling problems, University Exam Scheduling, Binary Integer Programming

# Índice

<b>AGRADECIMENTOS.....</b>	<b>I</b>
<b>RESUMO .....</b>	<b>II</b>
<b>ABSTRACT.....</b>	<b>III</b>
<b>ÍNDICE .....</b>	<b>IV</b>
<b>ÍNDICE DE FIGURAS.....</b>	<b>VI</b>
<b>ÍNDICE DE TABELAS .....</b>	<b>VII</b>
<b>1. INTRODUÇÃO .....</b>	<b>1</b>
1.1. OBJETIVOS .....	1
1.2. CONTEXTUALIZAÇÃO .....	2
1.3. ORGANIZAÇÃO DO RELATÓRIO.....	2
1.4. CONTRIBUIÇÕES .....	2
<b>2. REVISÃO BIBLIOGRÁFICA .....</b>	<b>3</b>
2.1. INTRODUÇÃO.....	3
2.2. MÉTODOS EXATOS.....	4
2.3. HEURÍSTICAS E META-HEURÍSTICAS .....	9
2.3.1. <i>Heurísticas</i> .....	9
2.3.2. <i>“Tabu Search”</i> .....	11
2.3.3. <i>Algoritmos genéticos</i> .....	11
2.3.4. <i>“Simulated Annealing”</i> .....	12
2.3.5. <i>“Particle Swarm Optimization”</i> .....	12
2.3.6. <i>Metodologias híbridas</i> .....	13
2.3.7. <i>Outras metodologias</i> .....	14
<b>3. DESCRIÇÃO DO PROBLEMA .....</b>	<b>15</b>
3.1. INTRODUÇÃO.....	15
3.2. DESCRIÇÃO DO PROBLEMA .....	16
3.3. DESCRIÇÃO DO MÉTODO DE RESOLUÇÃO MANUAL .....	18
<b>4. FORMULAÇÃO DO PROBLEMA E IMPLEMENTAÇÃO COMPUTACIONAL.....</b>	<b>23</b>
4.1. FORMULAÇÃO DO PROBLEMA .....	23
4.2. IMPLEMENTAÇÃO COMPUTACIONAL .....	27
4.3. RESULTADOS COMPUTACIONAIS.....	34
4.4. ANÁLISE COMPARATIVA DAS SOLUÇÕES .....	37
<b>5. CONCLUSÕES .....</b>	<b>40</b>

<b>REFERÊNCIAS DOCUMENTAIS .....</b>	<b>41</b>
<b>ANEXO A- SOLUÇÃO ATUAL .....</b>	<b>45</b>
<b>ANEXO B – SOLUÇÃO DO MODELO .....</b>	<b>47</b>

## Índice de Figuras

Figura 1 - Tabela descritiva dos departamentos e dos tempos de computação (modelo PI) (Daskalaki and Birbas 2005) .....	7
Figura 2 - Tabela comparativa dos tempos de computação entre as duas metodologias (Daskalaki and Birbas 2005) .....	7
Figura 3 - Tabela comparativa dos níveis de satisfação entre as duas metodologias (Daskalaki and Birbas 2005) .....	8
Figura 4 - Exemplo de calendário de exames .....	17
Figura 5 - Fluxograma do procedimento seguido para efetuar a alocação de docentes a exames ...	19
Figura 6 - Folha exemplo da alocação das vigilâncias a exames .....	20
Figura 7 - Folha exemplo da distribuição de vigilâncias .....	21
Figura 8 - Exemplo de um ficheiro de submissão no portal .....	22
Figura 9 - Exemplo de um calendário de exames na sua versão final .....	22
Figura 10 - Sequência de funcionamento da implementação computacional do modelo .....	28
Figura 11 - Folha Excel com o mapa de exames .....	29
Figura 12 - Folha Excel com os dados das unidades curriculares .....	29
Figura 13 - Exemplo da matriz de preferências .....	30
Figura 14 - Folha Excel com a informação relativa aos docentes .....	31
Figura 15 - Exemplo da folha Excel com todos os dados de <i>input</i> para a aplicação C++ .....	31
Figura 16 - Exemplo da informação de INPUT referente às salas de exame .....	32
Figura 17 - Fluxograma do processo iterativo para obtenção da solução ótima .....	34
Figura 18 - Exemplo do ficheiro para carregamento no portal (solução do modelo) .....	34
Figura 19 - Tempo de computação para obtenção da solução na aplicação em C++ .....	36
Figura 20 - Pormenor dos rácios mínimos na solução manual .....	38
Figura A. 1 - Distribuição das vigilâncias (solução atual) 1/2 .....	45
Figura A. 2 - Distribuição das vigilâncias (solução atual) 2/2 .....	46
Figura B. 1 - Distribuição das vigilâncias (solução do modelo) 1/2 .....	47
Figura B. 2 - Distribuição das vigilâncias (solução do modelo) 2/2 .....	48

## *Índice de Tabelas*

Tabela 1 - Níveis de preferência e respetivo coeficiente.....	25
Tabela 2 - Legenda do fluxograma .....	34
Tabela 3 - Tempos de computação (solução do modelo).....	35
Tabela 4 - Resultados do processo iterativo .....	36
Tabela 5 - Rácios e desvios (solução do modelo) .....	37
Tabela 6 - Tabela de comparação de resultados.....	37
Tabela 7 - Tempo despendido (solução atual).....	38



# 1. INTRODUÇÃO

A gestão da calendarização de horários é um aspeto a ter consideração nos dias de hoje em qualquer instituição, visto ser um processo muito moroso e em alguns casos de elevada complexidade. Para facilitar e acelerar o processo de obter uma solução que utilize de forma eficaz e eficiente os recursos disponíveis, respeitando um grande número e variedade de restrições, recorre-se à criação de modelos matemáticos que simulam a realidade, em que a sua solução é obtida, normalmente, de forma mais rápida que a alternativa (encontrar uma solução manualmente).

Neste trabalho é apresentada uma formulação matemática para a solução do problema de afetação de docentes à vigilância de exames e descrita a sua implementação numa plataforma informática amigável para o utilizador.

## 1.1. OBJETIVOS

O objetivo deste trabalho é resolver o problema de alocação de docentes à vigilância de exames, garantindo que a distribuição das mesmas é feita de forma proporcional, respeitando ainda as restrições de número máximo de vigilâncias e limite de vigilâncias diário.

## **1.2. CONTEXTUALIZAÇÃO**

O problema tratado neste trabalho ocorre várias vezes durante o ano no Instituto Superior de Engenharia do Porto e ocupa durante várias horas um docente. Com o intuito de minimizar esse tempo, e libertar esse colaborador foi desenvolvido este trabalho.

Em parceria com os responsáveis que realizam os horários, conseguiu-se ter a percepção das dificuldades sentidas durante a realização dos mesmos, e tentou-se ajustar o modelo proposto de modo a ultrapassar essas dificuldades e se possível evitar as situações que levavam à sua ocorrência.

## **1.3. ORGANIZAÇÃO DO RELATÓRIO**

A organização deste trabalho será feita da seguinte forma:

- No capítulo 1 será feita uma breve introdução ao problema em estudo.
- No capítulo 2 será apresentado o estado da arte neste âmbito.
- No capítulo 3 será apresentada a descrição do problema, as variáveis envolvidas e o processo pelo qual é resolvido atualmente.
- No capítulo 4 é apresentada a formulação do problema, a sua implementação e a comparação da solução obtida com a solução atual.
- No capítulo 5 são apresentadas as conclusões bem como são indicadas algumas considerações que possam ser tomadas como referência para trabalhos futuros.

## **1.4. CONTRIBUIÇÕES**

As principais contribuições deste trabalho são a formulação de um modelo matemático para a solução de um problema real complexo e a sua implementação numa plataforma informática amigável para o utilizador.

## 2. REVISÃO BIBLIOGRÁFICA

Neste capítulo apresenta-se o estado da arte referente ao tema abordado.

### 2.1. INTRODUÇÃO

O problema tratado neste trabalho é um problema comum nos dias de hoje, a criação de horários, neste caso uma versão mais restrita e simples, a alocação de docentes à vigilância de exames.

De um modo genérico o problema consiste em alocar um conjunto de docentes a um conjunto de exames, que ocorre num pequeno espaço temporal, de modo a que não haja sobreposição de um ou do outro, ou seja, que num determinado período de tempo não se repitam nem docentes, nem exames. Sendo esta a condição, normalmente, obrigatória em todas as instituições, cada instituição acrescenta um número de considerações secundárias que melhor adapta o problema à sua realidade. A dimensão de qualquer uma destas variáveis, docentes e exames, varia de instituição para instituição.

Segundo Marco P. Carrasco (2001), esta tarefa é morosa, trabalhosa e realizada por pessoas que trabalham com o seu conhecimento dos recursos e limitações/restrições da instituição para a qual estão a realizar o calendário.

A procura de soluções para este tipo de problemas já vem sendo estudada à mais de meio século e continua ainda hoje a ser alvo de estudos, a cada ano que passa existem mais documentos com ideias inovadoras e novos estudos aplicados a casos reais, continuando assim a ser um tema muito discutido.

## 2.2. MÉTODOS EXATOS

Welsh e Powell (1967), desenvolveram um algoritmo para resolver a criação de calendários, que utiliza a coloração dos vértices de um grafo de modo a que não existissem dois vértices adjacentes da mesma cor e o número de cores utilizadas fosse o mínimo possível.

O artigo de Dimopoulou e Miliotis (2001) documenta a criação e implementação de um algoritmo de programação linear, para realizar a alocação de exames a períodos de tempo e a salas de aula. Devido ao elevado grau de variabilidade nas escolhas das disciplinas por parte dos alunos e ao limite de salas para realização de exames, este algoritmo tenta solucionar o problema criando grupos de exames alocados a períodos de tempos sujeitos a restrições criadas pelo utilizador. A solução obtida inicialmente por este algoritmo é depois tratada com um algoritmo heurístico que melhora a solução até que esta seja viável. Este algoritmo é muito flexível e permite a fácil construção e teste de novos calendários, variando a sua composição mediante as condições impostas pelo utilizador. O sucesso ou insucesso deste algoritmo reside na experiência do utilizador e no seu conhecimento do problema e das suas condicionantes.

Já Daskalaki, Birbas et al. (2004) apresentam artigos em que o modelo utilizado é um modelo em programação linear em que as variáveis de decisão são binárias.

Nesta abordagem são considerados seis elementos estruturais básicos:

- o dia da semana, em que se pode alocar uma disciplina, denotado pela letra  $I$ , por exemplo  $I = \{1, 2, 3, 4, 5\}$ ;

- o período de tempo do dia, em que se pode alocar uma disciplina, no caso em estudo, das 8:00AM até às 09:00PM, sendo que a cada período são considerados 45m para exame e 15m para troca de docentes/alunos, denotado pela letra  $J$ , por exemplo  $J = \{1, 2, \dots, 14\}$ ;
- o grupo de alunos, para qual os calendários são realizados, designado pela letra  $K$ , por exemplo  $K = \{turma\#1, turma\#2, \dots, turma\#|K|\}$ ;
- o docente, que irá lecionar uma determinada disciplina, denotada pela letra  $L$ , por exemplo  $L = \{docente\#1, docente\#2, \dots, docente\#|L|\}$ ;
- a disciplina, para um determinado grupo de alunos, denotada pela letra  $M$ , por exemplo  $M = \{disciplina\#1, disciplina\#2, \dots, disciplina\#|M|\}$ ;
- a sala de aulas, disponíveis para serem alocadas a realização de disciplinas, denotada pela letra  $N$ , por exemplo  $N = \{sala\#1, sala\#2, \dots, sala\#|N|\}$ .

Sendo as variáveis definidas como  $X_{i,j,k,l,m,n}$  onde  $i \in I, j \in J, k \in K, l \in L, m \in M, n \in N$ , toma o valor 1 quando a disciplina  $m$ , lecionada pelo docente  $l$  para o grupo de alunos  $k$ , está prevista para o período do dia  $j$  no dia  $i$  na sala de aula  $n$  e  $Y_{i,p_v,k,h_v,m,n}$  onde  $i \in I, k \in K, m \in M, n \in N$  e  $h_v$  e  $p_v$  são número naturais.  $Y_{i,p_v,k,h_v,m,n}$  toma o valor de 1 quando a disciplina  $m$ , o que requer uma sessão de  $h_v$  períodos consecutivos, está prevista para o dia  $i$  para o grupo de alunos  $k$  na sala de aula  $n$ , sendo que  $p_v$  toma o valor do número de repetições da disciplina.

Sujeito aos seguintes tipos de restrições:

- restrições de unicidade, garantem que o um docente num determinado período de tempo apenas tem uma disciplina para lecionar ou que um determinado grupo de alunos, num determinado período de tempo apenas têm uma aula para frequentarem;
- restrições de integralidade, garantia de que os horários são completos;
- restrições de consecutividade, garantem que o modelo consegue aceitar alocações de disciplinas que requerem múltiplo períodos;

- restrições de repetibilidade, conjunto de restrições ligado com a ideia de consecutividade no sentido de assegurarem a existência da quantidade certa de períodos de um determinado tipo de aula, bem como a quantidade correta de repetições de um dado tipo de aula em caso de aulas teóricas ou trabalho de laboratório;
- restrições de pré-atribuição, que garantem que uma disciplina fica alocada num determinado período, garantindo assim a sua atribuição ou então usada para facilitar os problemas computacionais.

O modelo representa um problema de otimização cujo objetivo é minimizar uma função linear de custos, sendo esses custos as preferências expressas sobre períodos de tempo, dias da semana ou até mesmo especificar uma determinada sala/docente para uma determinada disciplina, sendo que a definição apropriada destes coeficientes pode levar a um menor espaço de solução e a um tratamento mais fácil da mesma.

Segundo Qualizza e Serafini (2005), é apresentada uma abordagem em programação linear inteira baseada no algoritmo de geração de colunas para a resolução do problema. Neste modelo, cada coluna representa o calendário semanal de uma unidade curricular, ou seja, a cada coluna estão associadas as restrições e preferências do horário semanal de uma unidade curricular. O procedimento de geração de colunas neste modelo passa pela utilização de uma matriz em que se faz a relação entre a ocupação das salas e a não sobreposição de tempo de cada disciplina, sendo que a atribuição de uma coluna a uma disciplina passa pela seleção dos melhores períodos de tempo atualmente disponíveis no calendário.

Também no mesmo ano, Avella e Vasil'Ev (2005) apresentam um modelo de programação linear inteira formulado com um problema de empacotamento com restrições adicionais. De modo a reduzirem a dimensão do espaço de pesquisa, é definido um número máximo de dias em que os docentes devem comparecer na instituição. No entanto esta formulação apresenta uma grave desvantagem devido ao ponto anteriormente referido, caso a definição do número de dias de cada docente não seja a melhor o problema pode não ter solução, ou então a solução pode não ser viável.

Ainda no mesmo ano, Daskalaki e Birbas (2005) apresentam uma nova abordagem para problemas de criação de calendários universitários. O modelo apresentado é composto por

um processo de duas etapas que resolve eficientemente o problema. A primeira etapa faz uma relaxação das restrições no que diz respeito à realização sucessiva de disciplinas com múltiplos períodos, sendo que na segunda etapa estas restrições são recuperadas, visto serem computacionalmente mais pesadas que as restantes, e são gerados e resolvidos um conjunto de sub-problemas, um por cada dia da semana, de modo a otimizar o problema localmente. Os autores compararam esta abordagem com uma abordagem que resolve o problema em apenas uma etapa (modelo PI), utilizando dois casos de estudo diferentes, referentes a dois departamentos distintos (Figura 1) e verificou-se que a nova abordagem apresenta uma diminuição significativa no tempo de obtenção da solução (Figura 2) sem perda significativa de qualidade (perda de aproximadamente 1,5 a 2% na taxa de alocações preferenciais) como se pode ver na Figura 3.

Table 1  
Computation times for problems of different size

	Problem size				Computation
	Courses (#)	Lab courses (#)	Required teaching periods	Rooms/labs(#)	CPU time (min)
Problem #1	25	8	139	3/6	2.5
Problem #2	47	19	187	4/10	18.5
Problem #3	92	27	319	6/12	96.0

Figura 1 - Tabela descritiva dos departamentos e dos tempos de computação (modelo PI)  
(Daskalaki and Birbas 2005)

Table 2  
Computation times for the two solution approaches

	Problem size				Computation time	
	Student groups (#)	Teachers (#)	Courses/labs (#)	Teaching periods	Single-stage IP model (minutes)	Relaxation approach (seconds)
Problem #1	3	30	25/8	139	2.5	16
Problem #3	11	73	92/27	319	96	85

Figura 2 - Tabela comparativa dos tempos de computação entre as duas metodologias (Daskalaki and Birbas 2005)

Table 3  
Satisfaction levels achieved with the two different solution approaches

Teachers satisfaction	Problem #1		Problem #3	
	Single-stage IP model	Relaxation approach	Single-stage IP model	Relaxation approach
Successes	133 (95.68%)	131 (94.24%)	296 (92.79%)	290 (90.91%)
Failures	6 (4.32%)	8 (5.76%)	23 (7.21%)	29 (9.09%)
Total # of periods	139	139	319	319

Figura 3 - Tabela comparativa dos níveis de satisfação entre as duas metodologias (Daskalaki and Birbas 2005)

Utilizando (Daskalaki, Birbas, and Housos 2004) como referência, MirHassani (2006) resolve o problema recorrendo a programação linear inteira minimizando uma função de não preferências. O método utilizado é superior ao atualmente utilizado na universidade, e apresenta um bom desempenho em comparação com outras metodologias conhecidas.

Schimmelpfeng e Helber (2007) apresentam um artigo em que documenta a aplicação de um modelo em programação inteira para a resolução dum problema real de horários da “School of Economics and Management at Hannover University”, na Alemanha. Neste artigo são apresentados dois softwares para a resolução do problema, sendo que a versão “open source” demora minutos a resolver o problema, enquanto a versão comercial do software CPLEX demora apenas segundos. Segundo os autores é possível obter soluções de elevada qualidade mesmo utilizando software “open source”.

Baseando-se no artigo de (Daskalaki and Birbas 2005), Bakır e Aksop (2008) adaptam o modelo em programação inteira para criar horários para o Departamento de Estatística da Universidade de Gazi, Turquia, mudando a função objetivo de modo a minimizar a insatisfação total de todos os envolvidos e tornando as variáveis de decisão em variáveis binárias. A solução obtida por este método minimiza a insatisfação total e adere a todas as restrições.

Uma outra abordagem desenvolvida por Birbas, Daskalaki et al. (2009), onde apresentam dois modelos para a resolução do problema de horários. Um modelo agrega a componente unidade curricular à componente docente da variável, sendo atribuído um conjunto de penalidades que foi adicionado à função objetivo. O outro modelo a componente unidade curricular desaparece por completo, visto que a cada unidade curricular está associado um docente, assim a função objetivo só tem que minimizar o custo associado a cada professor.

Mais recentemente McCollum, McMullan et al. (2012) apresentam um modelo de programação inteira que utiliza duas variáveis de decisão, alocação de exames a um período e alocação de exames a salas de aula. O modelo apresentado representa o problema real proposto em “*2nd International Timetabling Competition*” (ITC2007).

Também no mesmo ano Santos, Uchoa et al. (2012) apresentam uma formulação de programação inteira para resolver o problema de criação de horários utilizando um algoritmo de corte e geração de colunas para resolver a relaxação linear. A utilização deste algoritmo deve-se ao grande número de variáveis. A metodologia seguida consiste em atribuir docentes a grupos de alunos tendo em consideração uma distribuição apropriada de aulas ao longo da semana, bem como as preferências dos professores.

### **2.3. HEURÍSTICAS E META-HEURÍSTICAS**

Para além de abordagens utilizando métodos exatos, existem outras metodologias adaptadas a este tipo de problema, como por exemplo, as heurísticas e meta-heurísticas, “*Tabu Search*”, “*Simulated Annealing*”, algoritmos genéticos entre outras.

O principal objetivo de uma heurística é encontrar soluções de boa qualidade e de forma rápidas. Quanto mais restrições forem fornecidas ao problema, maior é a probabilidade de obter uma solução ótima que satisfaça o problema.

Os métodos meta-heurísticos podem incluir várias heurísticas capazes de encontrar a uma solução para o problema de forma rápida e eficiente. O método de melhoria inicia-se com uma solução viável e vai sendo melhorado duma forma repetitiva até que nenhuma melhoria possa ser encontrada ou o critério de paragem seja satisfeito.

#### **2.3.1. HEURÍSTICAS**

Burke (2004) desenvolve um modelo que tem por base métodos de ordenação semelhantes aos utilizados nos métodos de coloração de grafos. Este método apresenta resultados tão bons ou melhores quando comparados com outras heurísticas devido ao facto de ter por base um método geral e adaptativo que o torna mais rápido e de fácil manuseamento. Neste

trabalho o autor apresenta casos de teste em que demonstra a adaptabilidade do modelo, aplicando-o a casos reais, em que o modelo resolveu um problema de ordenação aleatória dos exames e demonstrou que a capacidade adaptativa do método foi capaz de transformar uma má ordenação dos exames numa ordenação ótima.

Usando como base o trabalho desenvolvido por (Carter 2001), Murray, Müller et al. (2007) desenvolvem um algoritmo onde aplicam novas técnicas, que têm como propósito anular os efeitos de favoritismo, fazendo com a solução seja o mais equilibrada possível, ou seja, a distribuição dos recursos seja proporcional.

A metodologia apresentada no artigo de Burke, Mareček et al. (2010) consiste na divisão do problema de criação de horários em sub-problemas, fazendo com as soluções sejam encontradas mais rapidamente, no entanto estas são parciais. Os autores utilizaram programação inteira para implementar a heurística que resolve o problema estudado no artigo com o objetivo de encontrar uma solução viável para o problema de atribuição de eventos a salas e a períodos do dia. A atribuição de eventos a períodos do dia é realizada através da coloração de grafos associada a uma combinação linear que minimiza o número de violações das restrições suaves.

Ainda no mesmo ano Kahar e Kendall (2010), desenvolvem um novo modelo para a resolução do problema de criação de horários para exames na universidade “*Universiti Malaysia Pahang*” (UMP), Malásia. Deste novo modelo constam um novo grupo de restrições, que não tinham sido modeladas anteriormente, adicionando assim novos desafios na criação de horários exequíveis. Essas restrições contemplam a distância entre salas de exame e a divisão dos exames por várias salas. O novo modelo apresenta soluções melhores em comparação com o software da UMP, e adere a todas as restrições, coisa que o software atual não consegue.

Sabar, Ayob et al. (2012) apresenta um artigo onde apresenta um algoritmo que se baseia no processo de acasalamento de abelhas (“*honey-bee mating optimization algorithm*”). Este algoritmo é baseado na natureza, simulando o processo de acasalamento das abelhas, sendo testado em dois casos distintos, o primeiro caso aplicado a exames, o segundo caso aplicado a disciplinas. Os resultados desta aplicação apresentam, em alguns casos, melhoria em relação às outras abordagens, sendo que nos restantes estão dentro da média.

### **2.3.2. “TABU SEARCH”**

Utilizando a meta-heurística (“*Tabu Search*”) Glover (1989), e com a colaboração de Manuel Laguna em (1993) publicou um artigo a demonstrar a eficácia da heurística.

Ainda utilizando a metodologia “*Tabu Search*”, Santos, Ochi et al. (2004) resolvem o problema de horários definindo-o como programação semanal de aulas para que todas as restrições do problema sejam satisfeitas. A nova metodologia apresentada demonstra resultados melhores em comparação com a metodologia híbrida anteriormente em uso.

Dando seguimento a esta heurística foram publicados vários artigos a demonstrar os seus resultados e a sua eficácia, como por exemplo White, Xie et al. (2004) utilizando esta heurística com o objetivo de minimizar o desconforto geral dos recursos envolvidos apresenta algumas melhorias em relações a outras metodologias utilizadas na altura.

A metodologia “*Tabu Search*” aplicada nos casos anteriores é um processo iterativo que utiliza estruturas de memória para guiar um processo de pesquisa descendente, não sendo confundida com a ausência de movimentos de melhoria. Este processo parte de uma solução inicial e compara-a com a sua vizinhança (outras soluções), escolhendo sempre o melhor movimento possível, a nova solução será então a solução anterior mais o movimento. Quando não existe um movimento que apresente melhorias é escolhido um que penalize a solução. Devido a este facto existe a possibilidade de o algoritmo entrar em ciclos, para que isso não aconteça é implementado um pequeno mecanismo, que se designa de memória de curta duração, onde são registados os movimentos, evitando que o algoritmo os utilize de novo, esses movimentos são os que ficam com o estatuto de “tabu”.

### **2.3.3. ALGORITMOS GENÉTICOS**

Ghaemi, Vakili et al. (2007), usa a metodologia dos algoritmos genéticos para resolver o problema de criação de horários com o intuito de minimizar os conflitos. Os autores apresentam e testam duas versões de algoritmos genéticos, um algoritmo genético modificado e um algoritmo genético cooperativo. O algoritmo genético modificado é uma junção de um algoritmo genético com alguns operadores genéticos básicos, esta adição contribui com uma melhoria do desempenho geral do algoritmo. Para além destes

operadores são ainda adicionados operadores genéticos inteligentes que aumentam ainda mais o desempenho do algoritmo. Quando estas adições são transmitidas ao algoritmo genético cooperativo a evidencia dessas melhorias é ainda mais visível e significativa.

A abordagem deste problema realizada por Pillay e Banzhaf (2010), apresenta como alternativa um algoritmo genético. Este algoritmo trabalha em duas fases, na primeira fase procura encontrar solução válidas que não violem as restrições difíceis (“*hard constraints*”), na segunda fase otimiza a solução através da otimização das restrições fáceis (“*soft constraints*”). Os resultados obtidos, quando aplicados a modelos de teste, são dos melhores de entre as restantes metodologias existentes, e o seu desempenho é superior à maioria das alternativas.

#### **2.3.4. “SIMULATED ANNEALING”**

O artigo publicado por Pongcharoen, Promtet et al. (2008), apresenta uma nova ferramenta para o tratamento deste tipo de problemas denominada “*Stochastic Optimisation Timetabling Tool*”. Esta ferramenta utiliza diferentes tipos de algoritmos, entre eles estão algoritmos genéticos, algoritmos de “*Simulated Annealing*” e algoritmos de pesquisa aleatória. Para além disso, os algoritmos estão dotados de um procedimento de reparação que garante que as soluções impossíveis são retificadas, impedindo assim conflitos e garantindo que as salas têm capacidade para acomodar as disciplinas/exames.

#### **2.3.5. “PARTICLE SWARM OPTIMIZATION”**

Um outro tipo de abordagem é o método *PSO* (“Particle Swarm Optimization”), apresentado por Shu-Chuan, Yi-Tin et al. (2006). Este método otimiza numa forma iterativa o problema tentando encontrar a melhor solução.

Mais tarde Tassopoulos e Beligiannis (2012), realizaram um conjunto de estudos onde comparavam o método *PSO* com outras formas de resolver este tipo de problemas, demonstrando a sua superioridade.

### 2.3.6. METODOLOGIAS HÍBRIDAS

Azimi (2005) apresenta um trabalho em que aplica diversas metodologias para resolver o problema de criação de horários de exames. As metodologias utilizadas passam por quatro bem estudadas e conhecidas, “*Simulated Annealing*” (SA), “*Tabu Search*” (TS), “*Genetic Algorithm*” (GA), e “*Ant Colony System*” (ACS), e outras três metodologias híbridas que são a combinação de algumas das atrás mencionadas, nomeadamente um algoritmo sequencial TS-ACS, um algoritmo híbrido ACS / TS, e um algoritmo sequencial ACS-TS.

Al-Yakoob, Sherali et al. (2010), apresentam modelo bietápico para resolução do problema de criação de horário na universidade “*Kuwait University*” (KU), Kuwait. Numa primeira fase, foi desenvolvido um modelo de programação inteira mista para resolver o problema da atribuição de exames a períodos de tempo e a salas de aula tendo em conta as restrições relacionadas com conflitos. Na segunda fase, após a obtenção da solução do problema na primeira fase, é resolvido o problema da atribuição de vigilantes a exames onde é tido em conta as restrições de preferências dos vigilantes para períodos e dias específicos de exames. Para além disso, foi desenvolvido um procedimento heurístico iterativo que tem como função minimizar a insatisfação dos vigilantes dos exames. Os resultados obtidos superam em muito a abordagem existente (manual) quer dum ponto de visto individual quer do ponto de vista de equidade.

A metodologia híbrida desenvolvida por Turabieh e Abdullah (2011), incorpora operadores heurísticos eficazes no algoritmo “*Great Deluge*”. Segundo os autores a escolha deste algoritmo deve-se ao facto de ter apresentado bons resultados dentro do âmbito de resolução de problemas de criação de horários de exames. A aplicação das heurísticas a este algoritmo melhoram ainda mais o seu desempenho geral. As heurísticas utilizadas têm por base metodologias que aparecem na literatura com a designação “mecanismos do tipo eletromagnético”. O objetivo deste modelo é encontrar solução de elevada qualidade evitando soluções ótimas localmente, utilizando para isso um valor de força calculado. Esse valor, calculado dinamicamente, é utilizado como uma taxa de decaimento na determinação do nível utilizado pelo algoritmo “*Great Deluge*”. A metodologia foi comparada com outras metodologias nos casos de estudo, “Toronto benchmark” e ITC07, e apresenta soluções competitivas na maiorias dos casos, sendo que em dois casos a solução obtida através desta metodologia é a melhor.

Gunawan, Ng et al. (2012) apresentam uma metodologia para a resolução do problema de criação de horários através da divisão em dois sub-problemas, o problema de atribuição de professores e o problema de atribuição das disciplinas. Relativamente ao primeiro sub-problema, os docentes são alocados a disciplinas sem considerar a programação dos horários das disciplinas e dos seus períodos de tempo. No segundo sub-problema as disciplinas são alocadas a períodos de tempo assumindo que a alocação de docentes a disciplinas já tinha sido realizada. Para a obtenção da solução utilizando esta metodologia é utilizado um método híbrido que consiste na aplicação sequencial de dois algoritmos. O primeiro algoritmo aplica uma abordagem baseada numa relaxação Lagrangeana para a obtenção de uma solução inicial viável para o problema. O segundo algoritmo usado é um algoritmo de “*Simulated Annealing*” que é aplicado para melhorar a solução inicial obtida anteriormente.

### **2.3.7. OUTRAS METODOLOGIAS**

O artigo publicado por Mumford (2010) apresenta uma metodologia para a resolução do problema de criação de horários para exames utilizando um quadro múltiplo-objetivos. Neste quadro são apresentados como objetivos: a minimização da dispersão dos exames e a minimização do tamanho do calendário. São apresentados dois algoritmos, um para ordenar os exames e outro para criar soluções para os horários tendo como função melhorar as ordenações de exames numa forma iterativa usando heurísticas de ordenamento e respeitando os objetivos. Esta metodologia apresenta como pontos fortes: uma estrutura de multiobjectivo simples, soluções apresentadas com ordenações, um algoritmo que otimiza o comprimento e a proximidade do calendário em simultâneo, etc.

Outra abordagem seguida é a utilização do método GRASP (*Greedy Randomized Adaptive Search Procedure*) para resolver o problema por parte de Gogos, Alefragis et al. (2010). Este método utiliza um processo iterativo para a determinação da solução ótima, a cada iteração são criadas soluções aleatórias e encontrada a solução localmente ótima. Das soluções localmente ótimas é aplicado novamente um processo iterativo para encontrar a solução ótima global. Esta metodologia garantiu aos autores o segundo lugar na “*International Timetabling Competition*” no ano 2007 (ITC07).

# 3. DESCRIÇÃO DO PROBLEMA

Neste capítulo apresenta-se o problema e a forma como é atualmente resolvido.

## 3.1. INTRODUÇÃO

Este trabalho pretende analisar o problema de alocação de docentes a vigilâncias de exames, um problema que ocorre várias vezes durante o ano no Instituto Superior de Engenharia do Porto, que requiere a utilização de vários recursos por parte dos departamentos que o realizam, neste caso em estudo, o departamento de Engenharia Mecânica.

Deste departamento fazem parte duas licenciaturas e sete mestrados, sendo que por cada um deles existem sempre dois semestres por ano curricular e vários anos curriculares por curso. Por cada semestre, por cada ano e por cada curso existe um calendário de exames diferente, no entanto, como os docentes são os mesmos, a realização da alocação dos docentes tem que ser feita em simultâneo. Como essa tarefa é realizada manualmente, é

difícil e morosa e, frequentemente, após concluída pela primeira vez, é sujeita a muitos ajustes devido a reclamações por parte dos envolvidos.

No sentido de resolver estes problemas surgiu este trabalho, que tem como propósito desenvolver uma ferramenta, através da criação de um modelo matemático que se adeque ao problema e seja passível de implementar informaticamente, de forma a acelerar o processo de obtenção de soluções e o processo de efetuar alterações devido a reclamações.

### **3.2. DESCRIÇÃO DO PROBLEMA**

A época de exames tem uma duração de quatro semanas, cada uma com 5 dias, segunda-feira a sábado, sendo que ao sábado apenas existem períodos para exames de manhã. De segunda-feira a sexta-feira, cada dia de exame está dividido em cinco períodos de tempo (09:00, 11:00, 14:00, 18:00 e 19:00 horas), aos sábados existem apenas três períodos (08:30, 09:00 e 11:00 horas).

Dos períodos anteriormente referidos, o período das 18:00 e o período das 19:00, assim como os períodos das 8:30 e 9:00 ao sábado, são considerados como um só, visto que os exames são pré-atribuídos, e para o mesmo curso é impossível haver um exame num período quando já existe no outro, ou seja, quando existe um exame no período das 18:00, não pode existir um exame no período das 19:00, e vice-versa, o mesmo se passa nos períodos das 8:30 e 9:00 ao sábado.

A Figura 4 exemplifica o calendário referente à época de exames do segundo semestre de 2011/2012, referente ao curso de Licenciatura em Engenharia Mecânica.

Mapa de Exames

Curso: Licenciatura em Engenharia Mecânica  
Ciclo: 1.º

Data de afixação: 01-06-2012

Ano Lectivo: 2011/2012

Semestre: 2.º



Ano	Segunda-feira		Terça-feira		Quarta-feira		Quinta-feira		Sexta-feira		Sábado	
	25-Jun	Hora	26-Jun	Hora	27-Jun	Hora	28-Jun	Hora	29-Jun	Hora	30-Jun	Hora
1	MATE1	14:00			IENG2	09:00			MTMET	14:00		
2			ESTAT	14:00							ELTRO	08:30
3	AUTO2	18:00							MTERM	18:00		
<b>02-Jul</b>												
1	FISIC	14:00			DESET	8.30			MECA1	09:00		
2			MEMAT	14:00			MEFLU	09:00			PFAB1	08:30
3	MFLUX	18:00							ORG12	18:00		
<b>09-Jul</b>												
1	MATE1	14:00			IENG2	14:00			MTMET	09:00		
2			ESTAT	14:00			ANPR2	14:00			ELTRO	08:30
3	AUTO2	18:00							MTERM	18:00		
<b>16-Jul</b>												
1	FISIC	14:00			DESET	14:00			MECA1	09:00		
2			MEMAT	14:00			PFAB1	14:00			MEFLU	08:30
3	MFLUX	18:00			PROJ2	18:00			ORG12	18:00		

Figura 4 - Exemplo de calendário de exames

Pretende-se simular a alocação de docentes a vigilâncias de exames, tendo em conta um conjunto de restrições que condicionam a solução, criando um modelo o mais próximo possível da realidade.

Esta alocação deve respeitar as seguintes restrições obrigatórias (“*hard constraints*”):

- o número máximo de vigilâncias que cada docente pode efetuar num determinado período de exames, que varia de docente para docente em função da sua carga horária, não pode ser ultrapassado;
- cada docente só pode realizar uma vigilância por dia;
- cada sala de exame deve ter dois vigilantes;
- o mesmo docente não pode ser escalado para vigilâncias com sobreposição de horário.

Está ainda sujeita, na medida do possível, a uma outra série de restrições que, não sendo obrigatórias (“*soft constraints*”), melhoram a qualidade da solução. Uma restrição que garante a distribuição proporcional das vigilâncias pelos docentes, e uma outra que obriga

a uma afetação hierárquica por preferências. Essas preferências vão desde: regente, docente que deve sempre estar presente na vigilância do exame da unidade curricular que rege, a sem preferências; passando por docente da cadeira, docentes alocados a vigilâncias logo após aos regentes; exclusividades, docentes que apenas podem vigiar exames de um determinado curso ou conjunto de cursos; e impedimentos por parte dos docentes.

### **3.3. DESCRIÇÃO DO MÉTODO DE RESOLUÇÃO MANUAL**

A metodologia utilizada na sua forma atual, é composto por duas grandes etapas, a alocação de vigilantes aos exames e a criação de ficheiros para carregar o portal. Os tempos fornecidos pelo responsável pela realização dos calendários para estas etapas são 6 horas e 12 horas respetivamente, sendo que no tempo de criação de ficheiros está contemplado o tempo de submissão que é cerca de 1% do tempo total dessa etapa. O fluxograma que se segue (Figura 5) exemplifica o procedimento seguido pelo responsável pela criação dos exames.

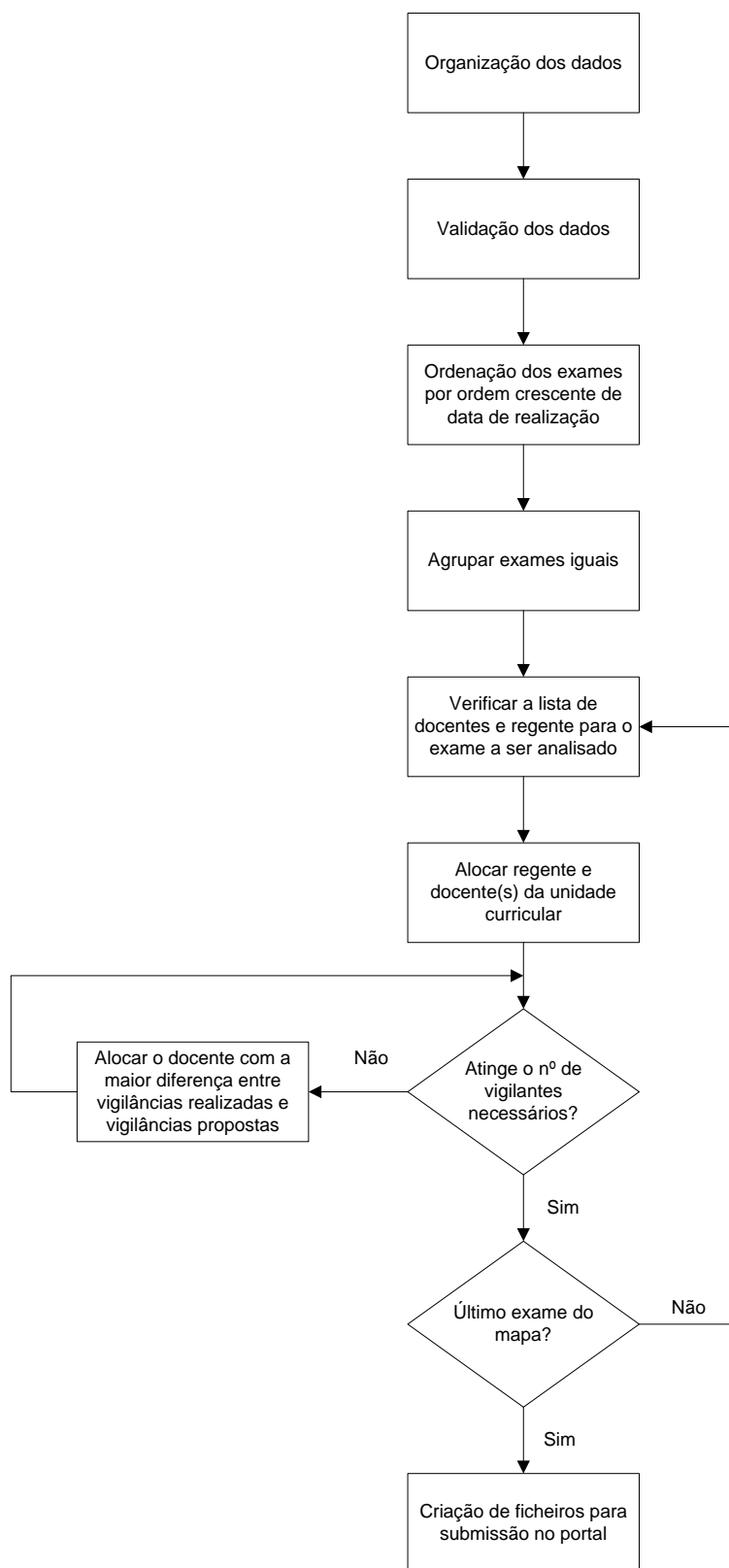


Figura 5 - Fluxograma do procedimento seguido para efetuar a alocação de docentes a exames

Numa primeira fase o responsável pela criação dos horários organiza os seguintes dados:

- Lista do número máximo de vigilâncias por docente;
- Mapas de exames com indicação de dia, hora e sala por exame;
- Lista de docentes e as unidades curriculares que lecionam;
- Lista de salas por exame;
- Lista de impedimentos por docente.

Estes dados são dispostos em folhas de cálculo sobre as quais o responsável efetua a alocação de vigilantes a exames. Na folha da Figura 6 constam a data, hora, exame, responsável pela unidade curricular do lado esquerdo; do lado direito a base que o responsável utiliza para fazer a alocação dos docentes.

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1								35		35		60		60		50		50
2								1	2	3	4	5	6	7	8	9	10	11
3	Data	Hora	U.C.	Curso	Época	Inscritos	Responsável	F202	F202	F203	F203	F204	F204	F207	F207	F208	F208	F209
4	23-Jan	09:00	ALGAN	Licenciatura Engenharia Mecânica	N	354	MGM	MGM	MJO	ATM	HMM	ADS	AFS	AGP	AGR	ALT	ANT	FBP
5	23-Jan	14:00	ALGEB	Licenciatura Engenharia Mecânica A	N	63	ATM	ATM	JAM	MMA	MJO							
6	23-Jan	18:00	PFAB2	Licenciatura Engenharia Mecânica	N	160	LLM				LLM	AAL	MJP	OMP	RFS	PAM		
7	23-Jan	18:00	MCINT	Licenciatura Engenharia Mecânica A	N	16	LSR	LSR	RFR									
8	23-Jan	19:00	TEOES	Mestrado Eng. Mecânica - Construçã	N	26	JSD			JSD	JSB							
9	23-Jan	19:00	TEOES	Mestrado Construções Mecânicas	N	5	JSD											
10	23-Jan	19:00	TECAI	Mestrado Eng. Mecânica - Gestão In	N	14	MIL											
11	23-Jan	19:00	TECAI	Mestrado Gestão de Processos e Op	N	1	MIL											
12	23-Jan	19:00	PROAF	Mestrado Eng. Mecânica - Materials	N	30	FGS											FGS
13	23-Jan	19:00	CLIMA	Mestrado Energias Sustentáveis	N	19	ISP											
14																		
15	24-Jan	09:00	TERMO	Licenciatura Engenharia Mecânica	N	473	MIC	MIC	ANT	APM	AQF	ASB	ATA	ATM	CFA	DAB	EAM	FAC
16	24-Jan	09:00	TERMO	Licenciatura Engenharia Mecânica A	N	41	PAA											
17	24-Jan	19:00	CONSM	Mestrado Eng. Mecânica - Construçã	N	19	FJF	FJF	LAM									
18	24-Jan	19:00	CONSM	Mestrado Construções Mecânicas	N	5	FJF											
19	24-Jan	19:00	FENTR	Mestrado Eng. Mecânica - Energia	N	14	ASR			ASR	FAC							
20	24-Jan	19:00	FENTR	Mestrado Energias Sustentáveis	N	51	ORC					ORC	AAL					
21																		
22	25-Jan	09:00	DEGER	Licenciatura Engenharia Mecânica	N	260	AGS	AGS	JFS	MMA	JPP	FCB	ALS	ALT	OMP	JPR	RFS	
23	25-Jan	09:00	DEGER	Licenciatura Engenharia Mecânica A	N	59	LMT											LMT
24	25-Jan	14:00	GOASE	Licenciatura Engenharia Mecânica A	N	16	AGR	AGR	CFC									
25	25-Jan	18:00	ORGL1	Licenciatura Engenharia Mecânica	N	259	MMS					MMS	MIL	ANT	PAM	PAC	FBP	HOC
26	25-Jan	18:00	ORGL1	Mestrado Eng. Mecânica - Construçã	N	16	AGP	AGP	CFE	RFR	FCB							

Figura 6 - Folha exemplo da alocação das vigilâncias a exames

Após a ordenação desta folha de cálculo são verificados quais os exames que podem ser considerados como iguais para efeito de alocação de vigilantes. Este processo conta não só com a experiência do responsável, conhecimento do histórico de realização de outros calendários, mas também com os dados dos regentes das unidades curriculares quando se tratam de disciplinas que variam o seu método de avaliação de ano para ano.

Sigla	Departamento	Proposto	Total	Diferença	Restrições	Compensações	Observações
AAL	DEM	6	4	2			
ADS	DEM	6	9	-3			
AFS	DEM	10	9	1			
AGM	DEM	8	3	5			
AGP	DEM	10	4	6			
AGR	DEM	10	9	0		1	
AGS	DEM	7	6	0		1	
AJC	DEM	8	3	5			
ALS	DEM	6	4	2			
ALT	DEM	6	3	3			
ANT	DEM	12	9	3			
APM	DEM	6	3	3			
AQF	DEE	10	0	10			
ASB	DEMA	10	0	10			
ASR	DEQ	6	3	3	MÊS		
ATA	DEE	10	2	8			
ATM	DMA	12	4	8			
CFA	DEQ	6	4	2			
CFC	DEM	6	2	4			
CMI	DEM	6	3	3			
DAB	DEM	8	3	5			
EAM	DEM	6	3	3			
FAC	DEM	8	7	1			
FBP	DEM	12	0	11		1	
FCB	DEM	6	4	1		1	
FGS	DEM	12	8	4			
FJF	DEM	8	3	5			
HDA	DEM	6	3	3			
HMM	DEI	6	0	6			
HJM	DEM	6	4	1		1	
HOC	DEM	8	4	4			

Figura 7 - Folha exemplo da distribuição de vigilâncias


O processo de atribuição de docentes às vigilâncias passa por atribuir a cada exame o regente da unidade curricular, seguidamente os docentes da cadeira até que o número de vigilantes requeridos para esse exame seja alcançado. Quando o número de docentes da cadeira não é o suficiente, os restantes docentes são alocados consoante a proporcionalidade de vigilâncias efetuadas perante o seu número máximo de vigilâncias proposto (Figura 7).

Concluída a alocação dos docentes, procede-se à criação dos ficheiros para serem lançados no portal (Figura 8), onde constam o dia, a hora, as salas e os docentes para cada exame, existindo um ficheiro por cada curso e por cada época (normal e recurso).

	A	B	C	D	E	
1	<b>Sigla</b>	<b>Disciplina</b>	<b>Data (aaaa-mm-dd)</b>	<b>Hora (hh:mm)</b>	<b>Siglas das Salas (separadas por vírgula)</b>	<b>Siglas dos Docentes (separadas por vírgula)</b>
2	DESET	04-07-2012	08:30	F202,F203,F204,F207,F208,F209,F309,F317	AGS,ALS,ALT,FAC,FCB,GEM,JDM,JFS,JPP,LMD,LMT,MJP,MMS,NFO,RFR,SEC	
3	FISIC	02-07-2012	14:00	F202,F203,F204,F207,F208,F209,F302,F303,F309,F317	AFL,ALG,APA,FLC,JDM,JFJ,JJV,JLL,JSD,LMO,LMT,MJP,SEC,VML	
4	IENG2	27-06-2012	09:00	F202,F203,F204,F207,F208,F209	AFG,CAS,ISP,JPR,JSB,JSD,LMO,MGM,MIL,NAL,PAA,PAC	
5	MATE1	25-06-2012	14:00	F202,F203,F204,F207,F208,F209,F309,F317	APM,ATM,JDM,LAM,MGM,MMA,PAC,PRF	
6	MECA1	06-07-2012	09:00	F202,F203,F204,F207,F208,F209,F303,F309,F317,F409,F515	AFL,ASA,ATM,EAM,FAC,GEM,HDA,ISP,JFJ,JJV,JSD,LLM,MDC,MGM,MPL,PRF,RA	
7	MTMET	29-06-2012	14:00	F202,F204,F207,F208,F303,F309,F317	AFL,AGM,ATM,CMI,FAC,HOC,JDM,JLL,MDJ,MIL,OCF,RHR	
8	ELTRO	30-06-2012	08:30	F204,F207,F208,F209	ALS,ANT,APM,ASA,LLM,MMA,NFO,RAR	
9	ESTAT	26-06-2012	14:00	F202,F203,F204,F207,F208,F209,F317	AGC,DAB,IPF,JSD,LMD,LMO,LOC,MDC,MDJ,MIL,MMS,NAL,OCF,RHR	
10	MEFLU	05-07-2012	09:00	F202,F203,F204,F207,F208,F209,F303,F309,F317,F409	AGM,AJC,ASR,CAS,CFA,CMI,FAC,FGS,JJV,LMD,LMT,LNH,MDJ,MGM,MIC,MMA,	
11	MEMAT	03-07-2012	14:00	F202,F203,F204,F207,F208,F209,F309,F317,F409	AJC,FGS,IPF,JPR,JSB,JSD,LAM,LMT,MDC,MDJ,MIL,MMA,RDS,RFS,RHR,VHN	
12	PFAB1	07-07-2012	08:30	F202,F203,F204,F207,F208,F209	ADS,AGR,ASA,FGS,HJM,HOC,JSD,MDJ,MGM,MIL,MJP,RFS	
13	AUTO2	25-06-2012	18:00	F204,F207,F209	ADS,AFS,AHA,LMO,RFS,VML	
14	MFLUX	02-07-2012	18:00	F204,F207,F208,F209	AFS,AHA,AJC,CFA,HOC,IPF,LLM,LOC	
15	MTERM	29-06-2012	18:00	F204,F207,F208,F209	ALG,ALS,ALT,ANT,APA,CAS,EAM,LSR,VMM	
16	ORGI2	06-07-2012	18:00	F202,F204,F207,F209	AGC,ANT,HOC,LNH,MIL,MMS,NFO,VML	
17						

Figura 8 - Exemplo de um ficheiro de submissão no portal

O mapa de exames final, submetido no portal tem o aspeto que se pode ver na Figura 9.

<b>Mapa de Exames</b>		<b>Data de afixação:</b> 15-07-2012			
<b>Curso:</b>	Mestrado Gestão de Processos e Operações	<b>Ano Lectivo:</b>	2011/2012		
<b>Ciclo:</b>	2.º	<b>Semestre:</b>	Época Especial		

Ano	Segunda-feira 03-Set	Hora	Terça-feira 04-Set	Hora	Quarta-feira 05-Set	Hora	Quinta-feira 06-Set	Hora	Sexta-feira 07-Set	Hora	Sábado 08-Set	Hora
1	PROAC F204	14:00	TECAI F204	19:00	GESEM F204	19:00	GQASE F204	19:00	FIAMA F204	19:00	GESAP F204	9:00
					AUTCI F204	14:00						
2	LOGIS F204	19:00					TEMOP F204	14:00				
<b>10-Set</b>												
<b>11-Set</b>												
<b>12-Set</b>												
<b>13-Set</b>												
<b>14-Set</b>												
1	TECFQ F204	14:00	ESEAD F204	19:00	PLANP F204	19:00	PLAPP F204	19:00	SMPFA F204	19:00		
	MODAD F409	19:00										
2							GESTP F204	14:00				
1												
2												
3												
1												
2												
3												

Figura 9 - Exemplo de um calendário de exames na sua versão final

# 4. FORMULAÇÃO DO PROBLEMA E IMPLEMENTAÇÃO COMPUTACIONAL

Neste capítulo apresenta-se a formulação do problema, a sua implementação e a comparação da solução obtida com a atual. O artigo de (Daskalaki, Birbas, and Housos 2004), “*An integer programming formulation for a case study in university timetabling*”, serviu de base a este trabalho visto que a abordagem apresentada é semelhante e muito abrangente, sendo ela convertida/restringida a um problema de alocação de docentes a exames, simplificando-o.

## 4.1. FORMULAÇÃO DO PROBLEMA

Seja  $P = (1, 2, \dots, n)$  o conjunto de docentes do departamento,  $D = (1, 2, \dots, l)$  o conjunto de dias do mapa de exames,  $E_j = (1, 2, \dots, m)$  o conjunto de exames num determinado dia  $j$

$\in D$ ,  $Vmax_i$  o número máximo de vigilâncias por docente  $i$ ,  $i \in P$  e  $Salas_{jk}$  o número de salas por exame  $j$ ,  $j \in D$  no dia  $k$ ,  $k \in E_j$ .

Cada docente  $i$ ,  $i \in P$ , tem um limite máximo de vigilâncias por época de exame  $Vmax_i$  que não pode ser ultrapassado, sendo que ainda tem um limite de vigilâncias máximo por dia igual à unidade. Por exame, são sempre necessários dois vigilantes por sala, logo cada exame  $k$ ,  $k \in E_j$ , no dia  $j$ ,  $j \in D$ , deve ter um número de vigilantes igual a  $Vig_{jk}$ , em que  $Vig_{jk} = 2 \times Salas_{jk}$ .

Para garantir uma distribuição equilibrada das vigilâncias pelos docentes, foi criado um rácio, calculado através do quociente entre o número total de vigilantes por época de exames e o número total de vigilâncias propostas para os docentes por época de exames (Equação 1).

$$\text{Rácio} = \frac{\sum \text{N}^\circ \text{ de vigilancias necessárias por exame}}{\sum \text{Vigilâncias propostas por docente}} = \frac{\sum_{j=1}^l \sum_{k=1}^m Vig_{jk}}{\sum_{i=1}^n Vmax_i} \quad (1)$$

Todos os docentes  $i$ ,  $i \in P$ , devem ter um número mínimo de vigilâncias efetuadas de modo a aproximar-se o mais possível desse rácio. Para que isso aconteça o número mínimo de vigilâncias de cada docente é calculado através da multiplicação do número máximo de vigilâncias propostas pelo rácio acima mencionado,  $Vmin_i = Vmax_i \times \text{Rácio}$ .

Por forma a fazer uma distinção entre os vários níveis hierárquicos para atribuição de vigilâncias são considerados todos os docentes e todas as unidades curriculares, sendo atribuído a cada par um coeficiente,  $Pref_{ijk}$ , na função objetivo consoante o seu nível observado na Tabela 1.

Tabela 1 - Níveis de preferência e respectivo coeficiente

Nível	Preferência	Coeficiente
1	Regente	100
2	Docente da cadeira	50
3	Exclusividade	20
4	Sem preferência	10
5	Impedimento	2

Estes níveis têm ordem de importância decrescente, sendo que o regente de uma cadeira, nível 1, deve sempre vigiar o exame da mesma, e o nível 5, que representa um impedimento por parte do docente, só deve ser utilizado em último recurso, ou seja, só mesmo quando não existe qualquer outra possibilidade.

Cada nível representa um coeficiente e uma condição diferentes; a sua relação é descrita abaixo:

- Nível 1 - “ Regente”: refere-se ao regente de uma unidade curricular, sendo este obrigado a estar presente na vigilância do exame da unidade curricular correspondente;
- Nível 2 - “ Docente da cadeira”: refere-se aos docentes das unidades curriculares, havendo a necessidade de um exame precisar de mais vigilantes, estes são os próximos a ser chamados a seguir aos regentes;
- Nível 3 - “ Exclusividade”: refere-se aos docentes que são exclusivos a uma licenciatura ou um mestrado, sendo que apenas podem vigiar exames desse mesmo mestrado ou licenciatura;
- Nível 4 - “ Sem preferência”: refere-se aos docentes que não lecionam as unidades curriculares mas podem efetuar vigilâncias aos exames das mesmas;

- Nível 5 - “ Impedimento”: refere-se aos docentes que, preferencialmente, não devem ser alocados para vigiar exames, sendo que a sua alocação pode ser feita apenas como último recurso.

Sejam as variáveis de decisão definidas como:

$$\mathbf{X}_{ijk} = \begin{cases} 1 - \text{Afectar o professor } i \text{ no dia } j \text{ ao exame } k \\ 0 - \text{caso contrário} \end{cases}$$

$$\mathbf{Y}_i = \begin{cases} 1 - \text{Caso o professor } i \text{ não consiga cumprir o rácio} \\ 0 - \text{caso contrário} \end{cases}$$

o problema pode ser definido como o seguinte modelo de programação matemática (P):

(P):

$$\text{Max } Z = \sum_{i=1}^n \sum_{j=1}^l \sum_{k=1}^l \text{Pref}_{ijk} \times \mathbf{X}_{ijk} - \sum_{i=1}^n 200 \times \mathbf{Y}_i \quad (2)$$

$$\text{s.a.: } \sum_{j=1}^l \sum_{k=1}^m \mathbf{X}_{ijk} \leq Vmax_i, i \in P \quad (3)$$

$$\sum_{k=1}^m \mathbf{X}_{ijk} \leq 1, i \in P, j \in D \quad (4)$$

$$\sum_{j=1}^l \sum_{k=1}^m \mathbf{X}_{ijk} \leq Vig_{jk}, i \in P \quad (5)$$

$$\sum_{j=1}^l \sum_{k=1}^m \mathbf{X}_{ijk} + \mathbf{Y}_i \geq Vmin_i, i \in P \quad (6)$$

$$\text{com: } \mathbf{X}_{ijk} \in \{0,1\}, \mathbf{Y}_i \in \{0,1\} \quad (7)$$

O coeficiente da variável  $\mathbf{Y}_i$  tem o valor “200”, de modo a evitar que esta variável seja utilizada de forma incorreta, o valor “200” é o dobro do melhor coeficiente que pode ser escolhido (100) que equivale a atribuição de uma vigilância a um regente.

A função objetivo (2) maximiza as preferências. O conjunto de restrições (3) garante que nenhum dos docentes excede o número máximo de vigilâncias proposto e o conjunto de restrições (4) garante que cada docente no máximo efetua uma vigilância por dia. O conjunto de restrições (5) e (6) são restrições que garantem a presença de vigilantes em cada exame e uma distribuição dessas mesmas vigilâncias de forma proporcional, respectivamente. A restrição (7) garante a integralidade binária das variáveis de decisão.

## **4.2. IMPLEMENTAÇÃO COMPUTACIONAL**

Os conjuntos de testes computacionais foram realizados num computador portátil com um processador INTEL (R) CORE (TM) i5 CPU M430 (2,27 GHz) com 4,00 GB de RAM.

A aplicação que corresponde à implementação deste modelo usa duas plataformas diferentes; usa folhas de cálculo do Microsoft Excel como base de dados, e uma aplicação em C++ que utiliza as bibliotecas do programa IBM ILOG CPLEX 12.4 para correr o modelo matemático e obter uma solução.

O fluxograma que se segue representa as etapas de funcionamento da implementação computacional do modelo desenvolvido neste trabalho.

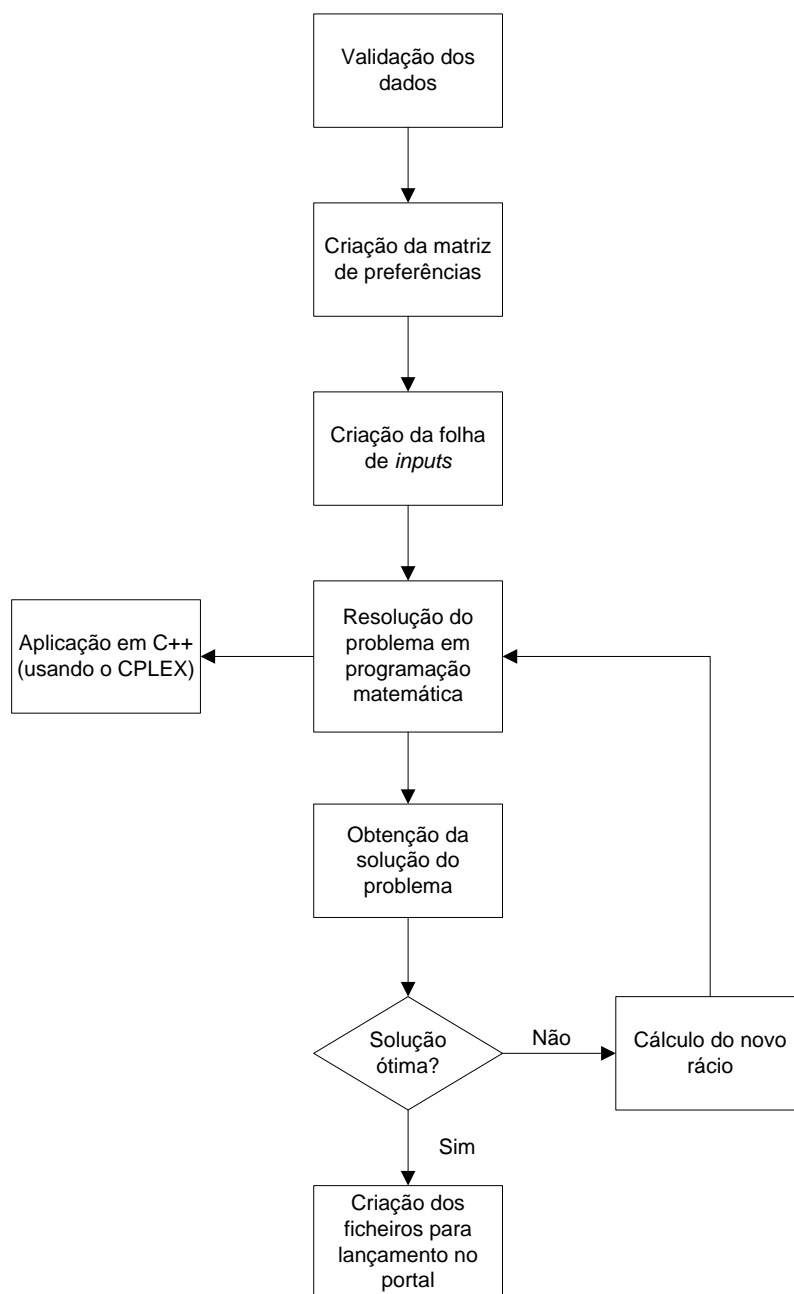


Figura 10 - Sequência de funcionamento da implementação computacional do modelo

A primeira etapa deste processo é a validação dos dados, nesta etapa verifica-se se existe uma correspondência entre os exames presentes na folha de cálculo do mapa de exames (Figura 11) e a folha de cálculo com a informação referente às unidades curriculares (Figura 12).

	A	B	C	D	E	F	G
1	Data	Hora	U.C.	Curso	Época	Regente	Nº Docentes
2	25-06-2012	09:00	MATE1	LEMA	N	JMS	1
3	25-06-2012	14:00	MATE1	LEM	N	JDM	3
4	25-06-2012	18:00	AUTO2	LEM	N	AFS	2
5	26-06-2012	09:00	ESTAT	LEMA	N	MHF	0
6	26-06-2012	14:00	ESTAT	LEM	N	AGC	3
7	26-06-2012	19:00	DEGPS	MEM-MT	N	SMS	0
8	26-06-2012	19:00	DIMES	MEM-CM	N	JFJ	0
9	26-06-2012	19:00	DIMES	MCM	N	JFJ	0
10	26-06-2012	19:00	ENRE2	MES	N	ATA	2
11	26-06-2012	19:00	EQTEH	MEM-EN	N	LSR	0
12	26-06-2012	19:00	ESEAD	MEM-GI	N	LNH	1
13	26-06-2012	19:00	ESEAD	MGPO	N	LNH	1
14	27-06-2012	09:00	IENG2	LEM	N	PAA	5
15	27-06-2012	14:00	ELTRI	LEMA	N	GCA	0
16	27-06-2012	18:00	SIPRA	LEMA	N	LSR	0
17	28-06-2012	14:00	PRFAB	LEMA	N	MDC	0
18	28-06-2012	19:00	AUTCI	MEM-GI	N	AFS	0
19	28-06-2012	19:00	AUTCI	MEM-MT	N	AFS	0
20	28-06-2012	19:00	MECFI	MEM-EN	N	PAA	0
21	28-06-2012	19:00	MECFI	MES	N	PAA	0
22	29-06-2012	09:00	GEOPS	LEMA	N	AGR	1
23	29-06-2012	14:00	MTMET	LEM	N	AGM	2
24	29-06-2012	14:00	MTMET	LEMA	N	OMP	0
25	29-06-2012	18:00	MTERM	LEM	N	LSR	1
26	29-06-2012	09:30	ELTRO	LEMA	N	ASA	2

Figura 11 - Folha Excel com o mapa de exames

Na folha apresentada na Figura 11 é reunida a informação referente aos exames, onde constam a data, hora, época, regente e número de docentes de cada exame/unidade curricular.

	A	B	C	D	E	F	G	H	I
1	Unidade Curricular	Sigla Curso	Curso	Inscritos Normal	Inscritos Recurso	Grupo	Regente	Docente	Docente
2	DESET	LEM	Licenciatura Engenharia Mecânica	330	160		AGS	ALT	LMT
3	FISIC	LEM	Licenciatura Engenharia Mecânica	470	250		APA	FLC	SEC
4	IENG2	LEM	Licenciatura Engenharia Mecânica	260	130		PAA	NAL	CAS
5	MATE1	LEM	Licenciatura Engenharia Mecânica	376	180		JDM	ATM	MGM
6	MECA1	LEM	Licenciatura Engenharia Mecânica	450	220		JSD	HDA	FAC
7	MTMET	LEM	Licenciatura Engenharia Mecânica	350	170		AGM	AGP	OMP
8	ANPR2	LEM	Licenciatura Engenharia Mecânica		5		JFS	MDC	
9	ELTRO	LEM	Licenciatura Engenharia Mecânica	220	100		ASA	NFO	RAR
10	ESTAT	LEM	Licenciatura Engenharia Mecânica	320	150		AGC	RHR	IPF
11	MEFLU	LEM	Licenciatura Engenharia Mecânica	450	220		FAC	ASR	MIC
12	MEMAT	LEM	Licenciatura Engenharia Mecânica	420	200		AJC	MMA	RDS
13	PFAB1	LEM	Licenciatura Engenharia Mecânica	280	140		FGS	MDJ	MJP
14	AUTO2	LEM	Licenciatura Engenharia Mecânica	160	80		AFS	ADS	AHA
15	MFLUX	LEM	Licenciatura Engenharia Mecânica	200	100		LOC	CFA	
16	MTERM	LEM	Licenciatura Engenharia Mecânica	180	90		LSR	VMM	
17	ORGI2	LEM	Licenciatura Engenharia Mecânica	180	90		MIL	ANT	
18	PROJ2	LEM	Licenciatura Engenharia Mecânica		5		ISP	LLM	
19	DESET	LEMA	Licenciatura Engenharia Mecânica Automóvel	70	35		AGS	RFR	OCF
20	ELTRI	LEMA	Licenciatura Engenharia Mecânica Automóvel	66	30		GCA		
21	LABAU	LEMA	Licenciatura Engenharia Mecânica Automóvel		5		DAB	APM	
22	MATE1	LEMA	Licenciatura Engenharia Mecânica Automóvel	76	35		JMS	ALG	
23	MTMET	LEMA	Licenciatura Engenharia Mecânica Automóvel	70	35		OMP		
24	MECA1	LEMA	Licenciatura Engenharia Mecânica Automóvel	87	40		JSD	JFJ	MDC

Figura 12 - Folha Excel com os dados das unidades curriculares

Na folha de dados com a informação das unidades curriculares (Figura 12) consta o nome da mesma, a sigla e nome do curso, o regente e os docentes das disciplinas. Para além disto

nesta folha constam o número de inscritos em ambas as épocas de recurso. De referir que estes dados são fornecidos pelo responsável pela criação dos horários de exames.

Após essa verificação, caso não haja problemas, procede-se à criação da matriz de preferências (Figura 13), para tal é feita a conjugação das folhas com os dados dos docentes (Figura 14) e dados das unidades curriculares (Figura 12), em que a cada docente se atribui um coeficiente mediante o seu nível hierárquico (Tabela 1).

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1				LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM	LEM
2		ID	Docente	DESET LEM	FISIC LEM	ENG2 LEM	MATE1 LE	MECA1 LE	MTMET LE	ANPR2 LE	ELTRO LEM	ESTAT LEM	MEFLU LEM	MEMAT LE	PFAB1 LEM	AUTO2 LEM
3		1	AAL													
4		2	ADS													50
5		3	AFG			50										
6		4	AFL		50											
7		5	AFS													100
8		6	AGC									100				
9		7	AGM						100							
10		8	AGP					50								
11		9	AGR													
12		10	AGS	100												
13		11	AHA													50
14		12	AJC										100			
15		13	ALG													
16		14	ALS	50												
17		15	ALT	50												
18		16	ANT													
19		17	APA		100											
20		18	APD	2	2	2	2	2	2	2	2	2	2	2	2	2
21		19	APM													
22		20	ASA								100					
23		21	ASR									50				
24		22	ATA	2	2	2	2	2	2	2	2	2	2	2	2	2
25		23	ATM				50									
26		24	CAS			50										

Figura 13 - Exemplo da matriz de preferências

Na Figura 14, da folha Excel com a informação dos docentes, constam a sigla do docente, o nome do docente, o departamento ao qual pertence, o número máximo de vigilâncias proposto por docente, as vigilâncias realizadas em épocas anteriores e os impedimentos que possam ter. Para além destes dados, são inseridos o número de vigilâncias que os docentes efetuam na atual época de exames, após a obtenção da solução.



exame, índice  $k$ . A coluna 1 é a coluna que faz a correspondência entre os dados da folha de Excel e a aplicação, pois a aplicação apenas trata as variáveis como números, não sabendo o seu conteúdo, ou seja, na aplicação, por exemplo, a variável 10 corresponde a atribuir o docente AFL à vigilância do exame “MATE1” a ser realizado no dia “25-06-2012”, no entanto na aplicação este é apenas tratado como “variável 10”, não tendo correspondência direta ao Excel, não fosse o caso de existir a coluna 1.

A informação referente às salas por exame, colunas “N” e “O” e coluna “D” dos ficheiros para submissão no portal, são fornecidas por uma outra aplicação, desenvolvida para realizar a alocação das salas a exames, sendo que um exemplo dos resultados dessa aplicação pode ser visto na Figura 16.

	A	B	C	D	E
1	Data	Hora	Exame	Salas	Nº Salas
2	25-06-2012	09:00	MATE1 LEMA	F204,F207	2
3	25-06-2012	14:00	MATE1 LEM	F204,F207,F208,F209	4
4	25-06-2012	18:00	AUTO2 LEM	F204,F208,F209	3
5	26-06-2012	09:00	ESTAT LEMA	F202	1
6	26-06-2012	14:00	ESTAT LEM	F202,F203,F204,F207,F208,F209,F317	7
7	26-06-2012	19:00	DEGPS MEM-MT	F202	1
8	26-06-2012	19:00	DIMES MCM	F208	1
9	26-06-2012	19:00	ENRE2 MES	F207	1
10	26-06-2012	19:00	EQTEH MEM-EN	F204	1
11	26-06-2012	19:00	ESEAD MGPO	F209	1
12	27-06-2012	09:00	IENG2 LEM	F202,F203,F204,F207,F208,F209	6
13	27-06-2012	14:00	ELTRI LEMA	F204,F207	2
14	27-06-2012	18:00	SIPRA LEMA	F202	1
15	28-06-2012	14:00	PRFAB LEMA	F202	1
16	28-06-2012	19:00	AUTCI MEM-MT	F202	1
17	28-06-2012	19:00	MECFI MES	F207	1
18	29-06-2012	09:00	GEOPS LEMA	F202	1
19	29-06-2012	14:00	MTMET LEM	F202,F203,F204,F207,F209,F317	6
20	29-06-2012	14:00	MTMET LEMA	F208,F309	2
21	29-06-2012	18:00	MTERM LEM	F202,F204,F207,F208	4
22	30-06-2012	08:30	ELTRO LEM	F204,F207,F208,F209	4
23	30-06-2012	08:30	MECV2 LEMA	F203	1
24	30-06-2012	11:00	OSENE MEM-EN	F202	1

Figura 16 - Exemplo da informação de INPUT referente às salas de exame

Após a resolução do problema, e com a solução obtida, há a necessidade de averiguar se esta é a solução ótima. Devido aos diferentes níveis hierárquicos da atribuição de alocações, os regentes das unidades curriculares têm sempre que fazer vigilância aos exames dessas mesmas unidades, logo a seguir são os docentes das cadeiras, faz com que

estes dois grupos de docentes tenham um rácio inevitavelmente superior aos restantes docentes, logo a solução obtida pode não ser a ótima. Neste caso de estudo em concreto existem vários docentes que apenas têm duas vigilâncias como máximo proposto, sendo que estes são, ou docentes ou regentes de uma cadeira, o seu rácio será inevitavelmente igual a 1.

Outro fator a ter em conta é o cálculo do rácio por docente só usar números inteiros, os números de vigilâncias realizadas e propostas por docente, que leva a diferenças significativas por vigilância realizada; por exemplo, um docente com 4 vigilâncias propostas, se realizar uma tem um rácio de 0,25, caso realize 2, o seu rácio passa a 0,50, caso o rácio a atingir seja de 0,60, este docente só tem uma hipótese, realizar 3 ou mais vigilâncias, que equivale a um rácio de 0,75 ou rácio de 1,0, caso realizasse apenas uma ou duas vigilâncias não conseguia atingir o rácio pretendido. Devido ao número limitado de vigilâncias por época de exame, o fator anteriormente referido leva a soluções que, apesar de serem válidas, permitem que existam docentes com ocupação total, rácio igual a 1,0, e docentes sem realizar uma única vigilância, com rácio igual a 0.

Assim sendo houve a necessidade de encontrar uma maneira de automatizar o processo de obtenção do melhor rácio que conduzisse a uma solução ótima, evitando assim um processo de tentativa e erro.

O processo desenvolvido para solucionar este problema é descrito no fluxograma abaixo (Figura 17).

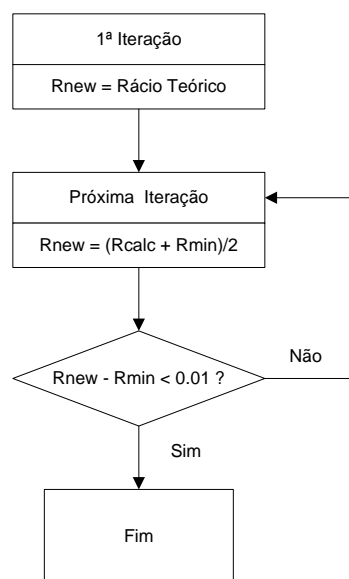


Figura 17 - Fluxograma do processo iterativo para obtenção da solução ótima

Onde:

Tabela 2 - Legenda do fluxograma

Sigla	Significado
Rcalc	Rácio calculado a cada iteração
Rmin	Rácio mínimo (o menor rácio verificado no conjunto dos docentes)
Rnew	Novo Rácio (o novo rácio a ser utilizado na iteração seguinte)

A obtenção da solução ótima é um processo iterativo, em que a cada iteração o novo rácio é calculado através da média entre o rácio calculado na iteração anterior e o rácio mínimo obtido, sendo que na primeira iteração apenas se usa o rácio (teórico) calculado através da equação (1). Este processo pára quando a diferença entre o rácio calculado na iteração anterior e o rácio mínimo, seja inferior a 0.01, ou seja 1% de margem erro.

Com a solução ótima obtida são criados os ficheiros individuais para carregar o portal, com o formato apresentado na Figura 18.

	A	B	C	D	E
1	Sigla Disciplina	Data (aaaa-mm-dd)	Hora (hh:mm)	Siglas das Salas (separadas por vírgula)	Siglas dos Docentes (separadas por vírgula)
2	ESEAD	26-06-2012	19:00	F207	LNH,SFR
3	MODAD	03-07-2012	19:00	F204,F207	AGR,MGM,MJP,MPL
4	PLAPP	07-07-2012	11:00	F207	ANT,JAB
5	TECFQ	05-07-2012	19:00	F207	JJV,MMS
6					

Figura 18 - Exemplo do ficheiro para carregamento no portal (solução do modelo)

### 4.3. RESULTADOS COMPUTACIONAIS

Neste subcapítulo são apresentados os resultados obtidos pela aplicação e tempos de computação.

O modelo matemático foi testado e comparado tomando por base os horários dos exames do segundo semestre de 2011/2012 do departamento de engenharia mecânica do ISEP.

A solução na íntegra pode ser visualizada no Anexo B, onde contém o número total de vigilâncias realizadas e o número de vigilâncias propostas para cada docente.

Na Tabela 3 pode-se observar os tempos de computação para obtenção da solução utilizando a aplicação desenvolvida.

Tabela 3 - Tempos de computação (solução do modelo)

Etapa	Tempo de computação
Carregamento do modelo	2 minutos e 19 segundos
Obtenção da solução ótima	55 segundos
Criação dos ficheiros para lançamento no portal	34 segundos
Total	3 minutos e 48 segundos

A primeira etapa, carregamento do modelo, são realizadas todas as operações necessárias de leitura e tratamento dos dados.

A segunda etapa, obtenção da solução ótima, são carregados os dados na aplicação C++, que configura o CPLEX para a resolução do problema. Após a resolução do problema é criado um ficheiro com a solução que será posteriormente lido no Excel. É também nesta etapa que se realiza o processo iterativo.

A terceira etapa lê o ficheiro solução, correspondente à última iteração do problema, e cria os ficheiros para lançamento no portal.

Como se pode verificar na Tabela 3, o tempo total necessário para obtenção da solução é de 3 minutos e 48 segundos. A obtenção da solução não necessita de tanta rapidez, no entanto fica salientado o reduzido tempo de processamento da aplicação.

Na Tabela 4 podem ser observados os tempos de processamento de cada etapa do processo iterativo até se obter a solução ótima. Através da sua análise pode-se verificar que o processo converge rapidamente para a solução ótima. Pode-se ainda observar que o tempo de computação decresce à medida que se aproxima da solução ótima, devido ao facto de o rácio ser menor, logo não exige tanto tempo de computação para conseguir satisfazer a restrição (6).

Tabela 4 - Resultados do processo iterativo

Iteração	Rácio (utilizado)	Rácio Mínimo	Tempo de computação (segundos)
1	0.6	0.500	4.2
2	0.55	0.500	4.0
3	0.52	0.500	3.8
4	0.51	0.500	3.5
5	0.50	0.500	3

Embora o tempo de computação por iteração seja, em média, 4s, o tempo de computação para obtenção da solução (Figura 19), em média, é de apenas 0.25 segundos, sendo o tempo restante despendido na escrita e criação do ficheiro com a solução.

```

C:\Users\JPNS\Documents\Prototipo.exe
Presolve time = 0.16 sec.
Probing time = 0.03 sec.
Click table members: 2300.
MIP emphasis: balance optimality and feasibility.
MIP search method: dynamic search.
Parallel mode: deterministic, using up to 4 threads.
Root relaxation solution time = 0.11 sec.

      Nodes
      Node Left      Objective  IInf  Best Integer  Cuts/
                                         Best Bound  ItCnt  Gap
*      0+      0          integral      0   8332.0000    17220.0000  638  ---
*      0      0          integral      0   17220.0000    17220.0000  638  0.00%
Elapsed real time = 0.44 sec. (tree size = 0.00 MB, solutions = 2)

Root node processing (before b&c):
  Real time = 0.28
Parallel b&c, 4 threads:
  Real time = 0.00
  Sync time (average) = 0.00
  Wait time (average) = 0.00
-----
Total (root+branch&cut) = 0.28 sec.
Prima qualquer tecla para continuar . . .
  
```

Figura 19 - Tempo de computação para obtenção da solução na aplicação em C++

Em termos de qualidade de solução do programa temos como parâmetros de comparação: rácio teórico, rácio mínimo, desvio máximo (diferença entre o rácio mínimo e o rácio teórico) e o desvio padrão.

Tabela 5 - Rácios e desvios (solução do modelo)

Rácio Teórico	0,6
Rácio Mínimo	0,50
Desvio máximo	0,1
Desvio Padrão	0,207

#### 4.4. ANÁLISE COMPARATIVA DAS SOLUÇÕES

A comparação em termos de distribuição proporcional das vigilâncias será analisada através dos seguintes parâmetros em ambas as soluções (Anexo A e Anexo B), rácio teórico, rácio mínimo, desvio máximo e o desvio padrão. Como referido anteriormente o rácio teórico é calculado através da equação (1) e o rácio mínimo é o menor rácio verificado no conjunto dos docentes. Na Tabela 6 podem ser verificados os rácios obtidos em ambas as soluções.

Tabela 6 - Tabela de comparação de resultados

	Solução manual	Solução do modelo
Rácio Teórico	0,6	0,6
Rácio Mínimo	0	0,50
Desvio Máximo	0,6	0,1
Desvio Padrão	0,215	0,207

Em termos de rácio mínimo, na solução do modelo este encontra-se perto do rácio teórico enquanto na solução manual ele toma o valor de zero, isto deve-se ao facto de haver um docente que não efetua uma única vigilância. O segundo rácio mais baixo na solução manual é 0,2, rácio esse que se encontra muito longe do teórico. Estes rácios são calculados dividindo o número de vigilâncias realizadas pela diferença entre as vigilâncias propostas e as realizadas anteriormente, noutras épocas de exame no mesmo ano curricular, por exemplo, cursos trimestrais ou até mesmo pós-graduações. Tomando como exemplo o

segundo rácio mais baixo da solução manual, 0,2 para o docente VML, este foi calculado através do quociente entre 1, número de vigilâncias realizadas na época de exames em estudo, e 5, diferença entre o número de vigilâncias propostas (6) e as vigilâncias realizadas anteriormente (1), estes valores podem ser verificados na Figura 20.

SIGLA	Dep.	Restrição	Proposto	Total	Diferença	Realizadas
PAA	DEM		10	7	3	
PAC	DEM		6	4	2	
PRF	DEMA		6	2	4	
PSA	DEM		2	0	2	
RAR	DEE		6	2	4	
RDS	DEM		6	2	4	
RFR	DEM		6	3	3	
RFS	DEM		12	8	4	
RHR	DEMA		12	8	4	
RMP	DEQ		6	4	2	
SEC	DEFI		6	2	4	
SFR	DEMA		3	2	1	
SMS	DEQ	MES	2	2	0	
TIS	DEC	MES	2	2	0	
VHN	DEM		6	2	3	1
VML	DEM		6	1	4	1
VMM	DEM		6	3	2	1

Figura 20 - Pormenor dos rácios mínimos na solução manual

Quanto ao desvio padrão podemos constatar que a solução do modelo apresenta o valor mais baixo, indicando uma menor dispersão dos valores, o que evidencia uma melhor distribuição das vigilâncias pelos docentes.

Na Tabela 7 estão indicados os tempos despendidos em cada etapa do processo manual.

Tabela 7 - Tempo despendido (solução atual)

Etapa	Tempo despendido
Alocação de docentes a vigilâncias	6 horas
Criação e lançamento dos ficheiros no portal	12 horas
Total	18 horas

Como podemos verificar pela Tabela 3, o tempo de alocação de docentes a vigilâncias e criação dos ficheiros para submissão no portal é de 3 minutos e 32 segundos, já contabilizando o tempo de leitura de dados de input para o programa, enquanto na solução manual estas duas etapas tem um tempo acumulado de 18 horas.

Para eventuais correções ao plano assume-se que o tempo será semelhante ao de efetuar uma simulação mais o tempo que o utilizador demore a efetuar as alterações dos dados. Tendo em conta a rapidez com que a aplicação encontra a nova solução, o tempo de processamento das reclamações é claramente inferior ao do método atual (manual), que segundo o responsável a troca de docentes tem uma duração média de 30 minutos e a alocação de mais docentes entre 1 a 2 horas.

## 5. CONCLUSÕES

Este trabalho constitui um estudo de um problema real da instituição de ensino ISEP, mais propriamente do departamento de Engenharia Mecânica, incidindo no problema de alocação de docentes à vigilância de exames.

O modelo foi implementado utilizando duas plataformas, as folhas de cálculo do Microsoft Excel como base de dados e meio de apresentação da solução, e uma aplicação desenvolvida em C++, para obtenção da solução, através da resolução do modelo matemático, recorrendo às bibliotecas do CPLEX.

A utilização da ferramenta gerada para este modelo apresenta como vantagens a rapidez na obtenção de solução, a garantia de que a alocação dos docentes é feita de uma forma proporcional e, na eventualidade de ser necessário fazer alterações no calendário, o processo é bastante mais rápido do que no processo atual (manual).

Como trabalho futuro, indica-se a possibilidade de alargar a implementação desta ferramenta a outros departamentos do ISEP.

## *Referências Documentais*

- Al-Yakoob, Salem; Sherali, Hanif and Al-Jazzaf, Mona. 2010. A mixed-integer mathematical modeling approach to exam timetabling. *Computational Management Science* 7 (1):19-46.
- Avella, Pasquale and Vasil'Ev, Igor. 2005. A Computational Study of a Cutting Plane Algorithm for University Course Timetabling. *Journal of Scheduling* 8 (6):497-514.
- Azimi, Z. N. 2005. Hybrid heuristics for examination timetabling problem. *Applied Mathematics and Computation* 163:705–733.
- Bakır, M Akif and Aksop, Cihan. 2008. A 0-1 Integer Programming Approach To a University Timetabling Problem. *Hacettepe Journal of Mathematics and Statistics* Volume 37 (1):41 - 55.
- Birbas, T.; Daskalaki, S. and Housos, E. 2009. School timetabling for quality student and teacher schedules. *J. of Scheduling* 12 (2):177-197.
- Burke, E. K. and Newall, J. P. 2004. Solving Examination Timetabling Problems through Adaption of Heuristic Orderings. *Annals of Operations Research* 129 (1):107-134.
- Burke, Edmund K.; Mareček, Jakub; Parkes, Andrew J. and Rudová, Hana. 2010. Decomposition, reformulation, and diving in university course timetabling. *Computers & Operations Research* 37 (3):582-597.
- Carter, M.W. 2001. A comprehensive course timetabling and student scheduling system at the University of Waterloo. *Burke, E., Erben, W. (eds.) PATAT 2000* vol. 2079:64–82.
- Daskalaki, S. and Birbas, T. 2005. Efficient solutions for a university timetabling problem through integer programming. *European Journal of Operational Research* 160 (1):106-120.

- Daskalaki, S.; Birbas, T. and Housos, E. 2004. An integer programming formulation for a case study in university timetabling. *European Journal of Operational Research* 153 (1):117-135.
- Dimopoulou, M. and Miliotis, P. 2001. Implementation of a university course and examination timetabling system. *European Journal of Operational Research* 130 (1):202-213.
- Ghaemi, S.; Vakili, M. T. and Aghagolzadeh, A. 2007. Using a genetic algorithm optimizer tool to solve University timetable scheduling problem. Paper read at Signal Processing and Its Applications, 2007. ISSPA 2007. 9th International Symposium on.
- Glover, F. 1989. Tabu Search — Part I. *ORSA Journal on Computing* Vol. 1:190-206.
- Glover, F. and Laguna, M. 1993. Tabu search. Modern Heuristics Techniques for Combinatorial Problems. *Blackwell Scientific Publications, Oxford*:70-150.
- Gogos, Christos; Alefragis, Panayiotis and Housos, Efthymios. 2010. An improved multi-staged algorithmic process for the solution of the examination timetabling problem. © *Springer Science+Business Media* 194 203–221.
- Gunawan, Aldy; Ng, Kien Ming and Poh, Kim Leng. 2012. A hybridized Lagrangian relaxation and simulated annealing method for the course timetabling problem. *Computers & Operations Research* 39 (12):3074-3088.
- Kahar, M. N. M. and Kendall, G. 2010. The examination timetabling problem at Universiti Malaysia Pahang: Comparison of a constructive heuristic with an existing software solution. *European Journal of Operational Research* 207 (2):557-565.
- Carrasco, Marco P. and Rato, Margarida V. 2001. A multiobjective genetic algorithm for the class/teacher timetabling problem. *E. Burke and W. Erben (Eds.): 2079*:pp. 3–17.
- McCollum, Barry; McMullan, Paul; Parkes, Andrew; Edmund Burke, and Qu, Rong. 2012. A new model for automated examination timetabling. *Annals of Operations Research* 194 (1):291-315.

- MirHassani, S. A. 2006. A computational approach to enhancing course timetabling with integer programming. *Applied Mathematics and Computation* 175 (1):814-822.
- Mumford, Christine. 2010. A multiobjective framework for heavily constrained examination timetabling problems. *Annals of Operations Research* 180 (1):3-31.
- Murray, Keith; Müller, Tomáš and Rudová, Hana. 2007. Modeling and Solution of a Complex University Course Timetabling Problem Practice and Theory of Automated Timetabling VI. In *Practice and Theory of Automated Timetabling VI*, edited by E. Burke and H. Rudová: Springer Berlin / Heidelberg.
- Pillay, N. and Banzhaf, W. 2010. An informed genetic algorithm for the examination timetabling problem. *Applied Soft Computing* 10 (2):457-467.
- Pongcharoen, P.; Promtet, W.; Yenradee, P. and Hicks, C. 2008. Stochastic Optimisation Timetabling Tool for university course scheduling. *International Journal of Production Economics* 112 (2):903-918.
- Qualizza, Andrea and Serafini, Paolo. 2005. A Column Generation Scheme for Faculty Timetabling. *Practice and Theory of Automated Timetabling V* 3616:161-173.
- Sabar, Nasser R., Ayob, Masri; Kendall, Graham; and Qu, Rong. 2012. A honey-bee mating optimization algorithm for educational timetabling problems. *European Journal of Operational Research* 216 (3):533-543.
- Santos, Haroldo G.; Ochi, Luiz S. and Souza, Marcone J.F. 2004. An Efficient Tabu Search Heuristic for the School Timetabling Problem. In *Proceedings of WEA*:468 - 481.
- Santos, Haroldo; Uchoa, Eduardo; Ochi, Luiz and Maculan, Nelson. 2012. Strong bounds with cut and column generation for class-teacher timetabling. *Annals of Operations Research* 194 (1):399-412.
- Schimmelpfeng, Katja and Helber, Stefan. 2007. Application of a real-world university-course timetabling model solved by integer programming. *OR Spectrum* 29 (4):783-803.

- Shu-Chuan, Chu; Yi-Tin, Chen and Jiun-Huei, Ho. 2006. Timetable Scheduling Using Particle Swarm Optimization. Paper read at Innovative Computing, Information and Control, 2006. ICICIC '06. First International Conference on, Aug. 30 2006-Sept. 1 2006.
- Tassopoulos, Ioannis and Beligiannis, Grigorios. 2012. Using particle swarm optimization to solve effectively the school timetabling problem. *Soft Computing - A Fusion of Foundations, Methodologies and Applications* 16 (7):1229-1252.
- Turabieh, Hamza and Abdullah, Salwani. 2011. An integrated hybrid approach to the examination timetabling problem. *Omega* 39 (6):598-607.
- Welsh, D. J. A. and Powell, M.B.. 1967. An upper bound to the chromatic number of a graph and its application to timetabling problem. *the computer journal* 10:85 - 86.
- White, George M.; Xie, Bill S. and Zonjic, Stevan. 2004. Using tabu search with longer-term memory and relaxation to create examination timetables. *European Journal of Operational Research* 153 (1):80-91.

## Anexo A – Solução atual

SIGLA	Dep.	Restrição	Proposto	Total	Diferença	Realizadas
AAL	DEM		6	4	1	1
ADS	DEM		12	9	3	
AFG	DEQ		3	2	1	
AFL	DEFI		6	3	3	
AFS	DEM		12	9	3	
AGC	DEMA		12	9	3	
AGM	DEM		6	3	3	
AGP	DEM		4	4	0	
AGR	DEM		12	9	3	
AGS	DEM		6	6	0	
AHA	DEM		6	5	0	1
AJC	DEM		6	3	3	
ALG	DEMA		6	3	3	
ALS	DEM		6	4	2	
ALT	DEM		6	3	3	
ANT	DEM		12	9	3	
APA	DEFI		4	2	2	
APD	DEE	MES	2	2	0	
APM	DEM		6	3	3	
ASA	DEE		12	9	3	
ASR	DEQ		6	3	3	
ATA	DEE	MES	3	2	1	
ATM	DEMA		8	4	4	
CAS	DEQ		9	5	4	
CFA	DEQ		8	4	4	
CFC	DEM		3	2	1	
CMI	DEM		6	3	3	
DAB	DEM		6	3	3	
EAM	DEM		6	3	3	
FAC	DEM		10	7	3	
FCB	DEM		6	4	2	
FFM	DEMA	MES	3	3	0	
FGS	DEM		12	8	4	
FJF	DEM		6	3	3	
FLC	DEFI		6	2	4	
FTA	DEE	MES	4	3	1	
GCA	DEE	LEMA	6	4	2	
GEM	DEM		6	2	2	2
HDA	DEM		6	3	1	2
HJM	DEM		6	4	2	
HMC	DEM		6	4	1	1
HOC	DEM		8	4	4	
IPF	DEMA		9	5	4	
ISP	DEM		12	8	4	

Figura A. 1 - Distribuição das vigilâncias (solução atual) 1/2

JAB	DEM		2	2	0	
JDM	DEMA		10	6	4	
JFJ	DEM		12	8	4	
JFS	DEM		3	3	0	
JJV	DEM		8	5	3	
JLL	DEFI		8	4	4	
JMS	DEMA	LEMA	2	2	0	
JOM	DEG	MES	2	2	0	
JPP	DEM		6	4	0	2
JPR	DEM		12	8	4	
JSB	DEM		6	3	2	1
JSD	DEM		12	8	4	
JSO	DEC	MES	2	2	0	
JTM	DEE	MES	2	2	0	
JTV	DEE	MEM	2	2	0	
LAM	DEM		6	3	3	
LLM	DEM		12	8	4	
LMD	DEM		6	2	4	
LMO	DEFI		8	4	4	
LMT	DEM		10	6	4	
LNH	DEMA		6	4	2	
LOC	DEM		6	3	3	
LSR	DEM		10	6	4	
MDC	DEM		10	6	4	
MDJ	DEM		12	8	4	
MGM	DEMA		10	6	4	
MHF	DEMA	LEMA	4	3	1	
MIC	DEM		4	4	0	
MIL	DEM		12	8	4	
MJP	DEM		6	3	3	
MMA	DEM		12	8	4	
MMS	DEM		12	9	3	
MPL	DEM		6	2	4	
NAL	DEM		6	2	3	1
NBG	DEE	MES	2	2	0	
NFO	DEE		12	8	4	
OCF	DEM		6	3	2	1
OMP	DEM		2	2	0	
ORC	DEM		12	8	4	
PAA	DEM		10	7	3	
PAC	DEM		6	4	2	
PRF	DEMA		6	2	4	
PSA	DEM		2	0	2	
RAR	DEE		6	2	4	
RDS	DEM		6	2	4	
RFR	DEM		6	3	3	
RFS	DEM		12	8	4	
RHR	DEMA		12	8	4	
RMP	DEQ		6	4	2	
SEC	DEFI		6	2	4	
SFR	DEMA		3	2	1	
SMS	DEQ	MES	2	2	0	
TIS	DEC	MES	2	2	0	
VHN	DEM		6	2	3	1
VML	DEM		6	1	4	1
VMM	DEM		6	3	2	1

Figura A. 2 - Distribuição das vigilâncias (solução atual) 2/2

## Anexo B – Solução do modelo

Sigla	Dep.	Restrição	Proposto	Total	Diferença	Realizadas
AAL	DEM		6	4	1	1
ADS	DEM		12	6	6	
AFG	DEQ		3	3	0	
AFL	DEFI		6	3	3	
AFS	DEM		12	6	6	
AGC	DEMA		12	6	6	
AGM	DEM		6	3	3	
AGP	DEM		4	4	0	
AGR	DEM		12	6	6	
AGS	DEM		6	4	2	
AHA	DEM		6	3	2	1
AJC	DEM		6	3	3	
ALG	DEMA		6	3	3	
ALS	DEM		6	3	3	
ALT	DEM		6	3	3	
ANT	DEM		12	6	6	
APA	DEFI		4	3	1	
APD	DEE	MES	2	2	0	
APM	DEM		6	3	3	
ASA	DEE		12	6	6	
ASR	DEQ		6	3	3	
ATA	DEE	MES	3	3	0	
ATM	DEMA		8	4	4	
CAS	DEQ		9	6	3	
CFA	DEQ		8	4	4	
CFC	DEM		3	2	1	
CMI	DEM		6	5	1	
DAB	DEM		6	5	1	
EAM	DEM		6	5	1	
FAC	DEM		10	8	2	
FCB	DEM		6	6	0	
FFM	DEMA	MES	3	3	0	
FGS	DEM		12	8	4	
FJF	DEM		6	6	0	
FLC	DEFI		6	5	1	
FTA	DEE	MES	4	4	0	
GCA	DEE	LEMA	6	6	0	
GEM	DEM		6	4	0	2
HDA	DEM		6	4	0	2
HJM	DEM		6	3	3	
HMC	DEM		6	3	2	1
HOC	DEM		8	4	4	
IPF	DEMA		9	5	4	
ISP	DEM		12	6	6	

Figura B. 1 - Distribuição das vigilâncias (solução do modelo) 1/2

JAB	DEM		2	2	0	
JDM	DEMA		10	5	5	
JFJ	DEM		12	6	6	
JFS	DEM		3	3	0	
JJV	DEM		8	4	4	
JLL	DEFI		8	4	4	
JMS	DEMA	LEMA	2	2	0	
JOM	DEG	MES	2	2	0	
JPP	DEM		6	2	2	2
JPR	DEM		12	6	6	
JSB	DEM		6	4	1	1
JSD	DEM		12	6	6	
JSO	DEC	MES	2	2	0	
JTM	DEE	MES	2	2	0	
JTV	DEE	MEM	2	2	0	
LAM	DEM		6	3	3	
LLM	DEM		12	6	6	
LMD	DEM		6	3	3	
LMO	DEFI		8	8	0	
LMT	DEM		10	5	5	
LNH	DEMA		6	3	3	
LOC	DEM		6	6	0	
LSR	DEM		10	10	0	
MDC	DEM		10	5	5	
MDJ	DEM		12	6	6	
MGM	DEMA		10	5	5	
MHF	DEMA	LEMA	4	4	0	
MIC	DEM		4	4	0	
MIL	DEM		12	6	6	
MJP	DEM		6	5	1	
MMA	DEM		12	6	6	
MMS	DEM		12	6	6	
MPL	DEM		6	3	3	
NAL	DEM		6	3	2	1
NBG	DEE	MES	2	2	0	
NFO	DEE		12	6	6	
OCF	DEM		6	3	2	1
OMP	DEM		2	2	0	
ORC	DEM		12	6	6	
PAA	DEM		10	6	4	
PAC	DEM		6	3	3	
PRF	DEMA		6	3	3	
PSA	DEM		2	1	1	
RAR	DEE		6	3	3	
RDS	DEM		6	3	3	
RFR	DEM		6	3	3	
RFS	DEM		12	6	6	
RHR	DEMA		12	6	6	
RMP	DEQ		6	6	0	
SEC	DEFI		6	3	3	
SFR	DEMA		3	2	1	
SMS	DEQ	MES	2	2	0	
TIS	DEC	MES	2	2	0	
VHN	DEM		6	3	2	1
VML	DEM		6	3	2	1
VMM	DEM		6	3	2	1

Figura B. 2 - Distribuição das vigilâncias (solução do modelo) 2/2