Optimizing Staff Scheduling in Emergency Medical Services: a case at INEM

Joana Maria Martins Namorado Rosa

Thesis to obtain the Master of Science Degree in

Biomedical Engineering

Supervisor: Prof. Inês Marques Proença

Prof. Ana Paula Ferreira Dias Barbosa Póvoa

Examination Committee

Chairperson: Prof. Mónica Duarte Correia de Oliveira

Supervisor: Prof. Inês Marques Proença

Members of the Committee: Prof. Maria Isabel Azevedo Rodrigues Gomes

November 2017
Acknowledgments

TO my supervisors, Professor Inês Marques and Professor Ana Póvoa, for giving me the opportunity to develop a study in my field of interest and for the valuable guidance, encouragement and sympathy.

TO Hendrik Vermuyten for the availability, persistence and wisdom. I express to him my deepest thanks and gratitude. It was a privilege for me to work and learn from Hendrik.

TO the people from INEM, namely Dr. Luis Meira, Jaime Naia and Filipe Louro for the kind and patient endless help and generous advices. I am thankful to their constant comprehension and immense engagement to this study.

TO my colleagues at IST who were present from the very beginning: Carmo, Irina, Chico and Mendonça. I thank them all for their immense joy and constant perseverance. To my other friends who listened to me and who helped me to achieve this milestone.

TO my family, in particular my parents, my four siblings and Miguel, for their unconditional kindness and extreme support throughout my life.
Abstract

The staff scheduling problem involves assigning people to tasks, integrated in working shifts. It is a complex and time-consuming activity common to several real-world companies. Quite often, staff scheduling problems comprise conflicting objectives. These problems are usually conditioned by legal and working rules, and by personal’s preferences. Thus, the challenge is not simply attaining feasible schedules but further finding schedules that most accurately fit staff demands, cost savings, individuals’ satisfaction, equity aspects, etc.

In this thesis, it is developed a model for the scheduling of workers. For that purpose, a standard integer programming formulation is proposed, and a general Column Generation-based diving heuristic approach is developed to solve the built model. The model is generic and easily adjusted to several realities and companies.

In this context, the model is applied to solve a real-life problem at Instituto Nacional de Emergência Médica (INEM). The quality and the computational time of the solutions obtained are analyzed, demonstrating that the heuristic developed finds good quality solutions for real instances in relatively short running times. The best-found solution is compared with a planned schedule of INEM, strengthening the practical value of the model.

Since the schedules at INEM are still manually performed, an automatic generator scheduling application to produce solutions with an intuitive graphical user interface is implemented. The ultimate goal is to use the tool to support INEM in their staff scheduling activities while enhancing finance viability, by detecting current scheduling-related issues and by constantly improving the schedules’ quality.

Keywords: Staff Scheduling, Emergency Medical Services, Optimization, Column Generation, Diving Heuristic


**Resumo**

O escalonamento de pessoal envolve a alocação de trabalhadores a tarefas, inseridas em turnos de trabalho específicos. É uma atividade complexa e morosa, comum a muitas empresas reais. Frequentemente, os problemas de alocação apresentam objetivos inconciliáveis. Estes problemas são geralmente condicionados por regras de natureza legal e laboral, e por preferências individuais. É necessário obter escalas viáveis que, simultaneamente, cumpram exigências de cariz pessoal, permitam poupança de custos, satisfação dos trabalhadores, considerem aspectos de equidade, etc.

Nesta dissertação, é desenvolvido um modelo para escalonamento de pessoal. É proposto um modelo baseado em programação inteira e o método de geração de colunas baseado numa abordagem heurística (*diving heuristic*) é desenvolvido como método de resolução do problema. O modelo apresenta caráter genérico podendo ser adaptado a diversas realidades/empresas.

Neste contexto, o modelo desenvolvido é aplicado ao problema real do Instituto Nacional de Emergência Médica (INEM). A qualidade da solução e o tempo de computação das soluções são analisados, demonstrando-se que a abordagem heurística permite obter soluções de qualidade para instâncias reais em tempos de computação relativamente curtos. A solução da melhor configuração encontrada é comparada com uma escala real prevista pelo INEM, reforçando a viabilidade do modelo.

Uma vez que os horários no INEM ainda são elaborados manualmente, é desenvolvida uma aplicação que permite elaborar escalas automaticamente, através de uma interface gráfica de utilização intuitiva. O objetivo final é utilizar esta ferramenta para auxiliar o INEM, gerando impacto financeiro, através da detecção de atuais problemas e constante melhoramento das escalas elaboradas.

**Palavras-chave:** Escalonamento de Pessoal, Serviços de Emergência Médica, Otimização, Geração de Colunas, Heurística
# Table of Contents

Acknowledgments .................................................................................................................................................. ii  
Abstract ................................................................................................................................................................. iv  
Resumo ...................................................................................................................................................................... vi  
Table of Contents .................................................................................................................................................. viii  
List of Figures ........................................................................................................................................................ xii  
List of Tables .......................................................................................................................................................... xiv  
List of Acronyms .................................................................................................................................................. xvi  
1. Introduction ......................................................................................................................................................... 1  
   1.1. Motivation and objectives .............................................................................................................................. 1  
   1.2. Methodology .................................................................................................................................................. 2  
   1.3. Thesis outline ............................................................................................................................................... 3  
2. Problem Contextualization ................................................................................................................................. 5  
   2.1. Sistema Integrado de Emergência Médica ................................................................................................. 5  
   2.2. Instituto Nacional de Emergência Médica ................................................................................................. 5  
      2.2.1. Emergency request ................................................................................................................................. 6  
      2.2.2. Emergency vehicles ............................................................................................................................... 7  
      2.2.3. Dispatching and routing ......................................................................................................................... 8  
   2.3. Staff scheduling problem at INEM ........................................................................................................... 9  
      2.3.1. Current status ......................................................................................................................................... 9  
   2.4. Chapter considerations ............................................................................................................................... 11  
3. Literature Review ................................................................................................................................................. 13  
   3.1. Staff scheduling problems .......................................................................................................................... 13  
      3.1.1. Staff scheduling in EMS ....................................................................................................................... 14  
   3.2. Methodologies ............................................................................................................................................ 14  
   3.3. Solution approach .................................................................................................................................... 17  
      3.3.1. Exact algorithms ................................................................................................................................. 17  
      3.3.2. Heuristic algorithms ......................................................................................................................... 19  
      3.3.3. Hybrid algorithms ............................................................................................................................. 19
4. Model Formulation .........................................................................................................................23
   4.1. Problem statement ..................................................................................................................23
   4.2. Standard IP formulation .........................................................................................................24
       4.2.1. Notation ..........................................................................................................................24
       4.2.1.1. Sets ..............................................................................................................................24
       4.2.1.2. Subsets .........................................................................................................................24
       4.2.1.3. Parameters ..................................................................................................................25
       4.2.2. Decision variables ..........................................................................................................25
       4.2.3. Objective function ..........................................................................................................26
       4.2.4. Constraints ......................................................................................................................26
   4.3. Solution approach .................................................................................................................31
       4.3.1. Alternative hybrid formulation .......................................................................................32
       4.3.1.1. Master problem ...........................................................................................................32
       4.3.1.2. Subproblems ...............................................................................................................33
       4.3.2. Strategies to solve the hybrid formulation .......................................................................35
       4.3.2.1. CG initialization and loop .........................................................................................35
       4.3.2.2. Stopping criteria of CG scheme .................................................................................37
       4.3.2.3. Diving heuristics ........................................................................................................38
   4.4. Chapter considerations ..........................................................................................................40
5. INEM Case-study .........................................................................................................................41
   5.1. INEM setting ..........................................................................................................................41
       5.1.1. The workforce – TEPHs .................................................................................................42
       5.1.2. Tasks and working teams ..............................................................................................42
       5.1.3. Shift sequences ..............................................................................................................44
       5.1.3.1. Stint basis at CODU .................................................................................................44
       5.1.3.2. EVs operation mode .................................................................................................44
   5.2. Input analysis ........................................................................................................................45
       5.2.1. General information .........................................................................................................46
5.2.1.1. More data sets ........................................................................................................50
5.2.2. Weights .....................................................................................................................51
5.3. Computational results ..................................................................................................52
  5.3.1. CPLEX .....................................................................................................................52
  5.3.2. CG and diving heuristic methods ............................................................................54
  5.3.3. Test sets ..................................................................................................................58
  5.3.4. Statistical tests .........................................................................................................60
  5.3.5. Analysis of the Best-Found Solution ......................................................................61
    5.3.5.1. Validation with a planned schedule at INEM ..................................................62
5.4. Graphical user interface ...............................................................................................65
5.5. Chapter considerations ...............................................................................................67
6. Conclusions and Future Remarks ..................................................................................69
  6.1. Recommendations to INEM .....................................................................................70
  6.2. Limitations and future remarks ................................................................................70
References ..........................................................................................................................73
Appendix I – Time and Quality Statistical Tests ..................................................................77
Appendix II - More Visualizations of the GUI Developed ..................................................79
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Representative scheme of the methodology adopted throughout the dissertation.</td>
<td>2</td>
</tr>
<tr>
<td>2.1</td>
<td>Workflow of a medical emergency request in continental Portugal.</td>
<td>6</td>
</tr>
<tr>
<td>2.2</td>
<td>Medical emergency response at CODU in Portugal continental.</td>
<td>8</td>
</tr>
<tr>
<td>2.3</td>
<td>Authentication webpage of Gestão de Horários (GH) platform.</td>
<td>10</td>
</tr>
<tr>
<td>2.4</td>
<td>Current scheduling process workflow at INEM.</td>
<td>11</td>
</tr>
<tr>
<td>3.1</td>
<td>Stages proposed by Blöchliger (2004) to model staff scheduling problems.</td>
<td>15</td>
</tr>
<tr>
<td>3.2</td>
<td>Column generation scheme in a node of the branch-and-bound tree (adapted from: Hans, 2001).</td>
<td>18</td>
</tr>
<tr>
<td>4.1</td>
<td>Illustration of the set of days to be constrained in (4.4) and (4.5).</td>
<td>27</td>
</tr>
<tr>
<td>4.2</td>
<td>Overview of the solution approach workflow. The solution of the CG might not be integer thus the diving heuristic is then applied to ensure that the final solution is integer.</td>
<td>31</td>
</tr>
<tr>
<td>4.3</td>
<td>Example of an activity pattern in which tasks are assigned for every person of the problem over the entire planning period.</td>
<td>32</td>
</tr>
<tr>
<td>4.4</td>
<td>Column scheme where one subproblem is solved for every person in each iteration.</td>
<td>36</td>
</tr>
<tr>
<td>4.5</td>
<td>Column generation scheme where a subproblem is solved for a single person after which the master is reoptimized.</td>
<td>36</td>
</tr>
<tr>
<td>4.6</td>
<td>Tailing-off effect suffered by CG for the problem in analysis.</td>
<td>37</td>
</tr>
<tr>
<td>4.7</td>
<td>Overview of branching on the largest fractional variable that is set to 1.</td>
<td>38</td>
</tr>
<tr>
<td>4.8</td>
<td>Overview of branching on variables with a value above $\delta$.</td>
<td>39</td>
</tr>
<tr>
<td>5.1</td>
<td>INEM services geographic distribution. Red marker indicates the CODU service location. (Google maps, 2017).</td>
<td>41</td>
</tr>
<tr>
<td>5.2</td>
<td>Stint sequence for the working groups at the CODU service.</td>
<td>44</td>
</tr>
<tr>
<td>5.3</td>
<td>Sample of person x team matrix for the real-life dataset at INEM.</td>
<td>47</td>
</tr>
<tr>
<td>5.4</td>
<td>Sample of team x task matrix for the real-life dataset at INEM.</td>
<td>47</td>
</tr>
<tr>
<td>5.5</td>
<td>Sample of task x demands matrix for the real-life dataset at INEM.</td>
<td>48</td>
</tr>
<tr>
<td>5.6</td>
<td>Sample of person x task matrix for the real-life dataset at INEM.</td>
<td>49</td>
</tr>
<tr>
<td>5.7</td>
<td>Full task x length matrix, for the real-life dataset at INEM.</td>
<td>49</td>
</tr>
<tr>
<td>5.8</td>
<td>Part of the results window using CPLEX tool on CODU dataset.</td>
<td>53</td>
</tr>
</tbody>
</table>
Figure 5.9 - Fragment of the BFS: tasks assigned to each person in the first week of the planning period...

Figure 5.10 - Fragment of the BFS: supply and demand for each task in the first week of the planning period..

Figure 5.11 - GUI initial screen after introducing the input data.

Figure 5.12 - GUI settings dialog. This tab is used to set the objective function weights and algorithm settings.

Figure 5.13 - GUI output for people tasks solution. This view shows the tasks assigned to each person over the entire planning period.

Figure 5.14 - GUI output for people shift requirements solution. This view shows the supply and demand for each task over the entire planning period.

Figure AII.1 - GUI visualization when the algorithm is running.

Figure AII.2 - GUI visualization once the algorithm finds a solution.

Figure AII.3 - GUI Help action. A tutorial explaining how to use the INEM shift scheduling problem is available for consultation.
List of Tables

Table 3.1 - Set of possible constraints found in a staff scheduling problem (Cheang et al., 2003). .......15
Table 5.1 - Set of tasks at the CODU service (CODU shift responsible and CODU task) and the EVs service (AEM driver, AEM team responsible, SIV task, TIP task, UMIPE task, MEM task). ..................42
Table 5.2 - Distance (km) between every pair of EV locations for the problem in analysis. ...............43
Table 5.3 - Distribution of the number of workers, task number and task type per working team for the present case-study at INEM. .............................................................................................................46
Table 5.4 - Summary of the instances used to perform the computational experiments. The INEM instance results from combining CODU and EVs. TestMD, Test MP, TestLS and Test HS are variants of the INEM instance, used to validate the algorithm. .................................................................51
Table 5.5 - Objective function weights used in the computational tests. ........................................51
Table 5.6 - Solution for the standard IP model using CPLEX as IP solver on three instances: CODU, EVs and INEM. ...............................................................................................................................................54
Table 5.7 - Impact of the different algorithm configurations tested on the CODU dataset..................55
Table 5.8 - Results for different algorithm configurations on EVs dataset. ....................................57
Table 5.9 - Results for different algorithm configuration on INEM dataset. ..................................57
Table 5.10 - Results for different algorithm configuration on four test sets: TestMD, TestMP, TestLS, TestHS........................................................................................................................................59
Table 5.11 - P-values obtained for the Wilcoxon signed rank test. .....................................................61
Table 5.12 - Coverage indicators: INEM real case and best solution case on CODU, EVs and INEM. 62
Table 5.13 - Analysis of the average number of Sundays and weekends off per person in INEM real case and best solution case on CODU, EVs and INEM. .................................................................64
Table 5.14 - Team assignment indicators: in INEM real case and best solution case on CODU, EVs and INEM..................................................................................................................................................64
Table AI.1 - Summary table with the computational results (in seconds) for the instances statistically tested in the different algorithm configurations..........................................................77
Table AI.2 - Summary of the ANOVA analysis results conducted to validate the computational times..77
Table AI.3 - Summary table with the BFS results for the instances statistically tested in the different algorithm configurations.. ........................................................................................................77
## List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Full Form</th>
</tr>
</thead>
<tbody>
<tr>
<td>AEM</td>
<td>Medical Emergency Motorcycle</td>
</tr>
<tr>
<td>BFS</td>
<td>Best Found Solution</td>
</tr>
<tr>
<td>CG</td>
<td>Column Generation</td>
</tr>
<tr>
<td>CODU</td>
<td>Centro de Orientação de Doentes Urgentes</td>
</tr>
<tr>
<td>EMS</td>
<td>Emergency Medical Services</td>
</tr>
<tr>
<td>EV</td>
<td>Emergency Vehicle</td>
</tr>
<tr>
<td>GH</td>
<td>Gestão de Horários</td>
</tr>
<tr>
<td>GUI</td>
<td>Graphical User Interface</td>
</tr>
<tr>
<td>HS</td>
<td>Homogeneous Skills</td>
</tr>
<tr>
<td>INEM</td>
<td>Instituto Nacional de Emergência Médica</td>
</tr>
<tr>
<td>IP</td>
<td>Integer Programming</td>
</tr>
<tr>
<td>LP</td>
<td>Linear Programming</td>
</tr>
<tr>
<td>LS</td>
<td>Less Symmetry</td>
</tr>
<tr>
<td>MEM</td>
<td>Medical Emergency Motorcycle</td>
</tr>
<tr>
<td>MD</td>
<td>More Days</td>
</tr>
<tr>
<td>MIP</td>
<td>Mixed Integer Programming</td>
</tr>
<tr>
<td>MP</td>
<td>More People</td>
</tr>
<tr>
<td>NSP</td>
<td>Nurse Scheduling Problem</td>
</tr>
<tr>
<td>RC</td>
<td>Reduced Cost</td>
</tr>
<tr>
<td>SIEM</td>
<td>Sistema Integrado de Emergência Médica</td>
</tr>
<tr>
<td>SIV</td>
<td>Immediate Life Support Ambulance</td>
</tr>
<tr>
<td>TEPH</td>
<td>Técnico de Emergência Pré-Hospitalar</td>
</tr>
<tr>
<td>TIP</td>
<td>Inter-hospital Pediatric Transport</td>
</tr>
<tr>
<td>UMIPE</td>
<td>Mobile Unit of Psychological Emergency Intervention</td>
</tr>
<tr>
<td>VNS</td>
<td>Variable Neighborhood Search</td>
</tr>
</tbody>
</table>
1. Introduction

1.1. Motivation and objectives

Healthcare has become a considerable portion of a country’s economy. In Portugal, the expenses were €2.30 per capita in 2016. Thirty years earlier, in 1986, these expenses were equal to €0.41 per capita, that is more than five times lower (OECD, 2017). This rise is also significant when comparing the total health care expenses as a percentage of the Portuguese GDP over the same period. The healthcare expenditure represented, in 2016, 8.9% of the national GDP, and 5.9% in 1986 (OECD, 2017). The main cause lies in the rapidly raise on the life expectancy, leading to an ageing society. Along with technological progression and development of new procedures, high pressures and demands emerge to ensure healthcare goods and services to the population. An urge to deliver the most effective care to individuals using the restricted resources available is imposed. Accordingly, clever planning and wise decisions are gainful to solve a wide range of healthcare related issues and prevent unnecessary expenses (Rais & Viana, 2011).

In this context, Operations Research techniques have been successfully applied to manage a variety of problems in healthcare environments (Brailsford & Vissers, 2011). These include staff scheduling problems, which seek to specify the human resources required to serve certain purposes, acknowledging applicable requirements such as organizational rules, personal's preferences and skills, demand requirements, among other requirements. It represents a difficult problem with specific objections that differ between industries and companies. Personnel scheduling highly influences the companies’ performances and therefore is a top preoccupation of c-level managers. Nowadays, in several organizations, schedules are still built manually. Indeed, time and resources could be saved if using an automatic scheduling generator instead. Automatic schedules also contribute to enhance transparency perception and also to an easier detection of failures. Proactive and informed staffed decisions can then be performed (e.g. optimize staff coverage or track workers skills). Staff scheduling problems have gathered a lot of attention from the academic community. Models are usually developed to solve case-specific problems, being sometimes difficult to adapt to other contexts as they require significant reformulation. Usually the problems are complex and with many dimensions, forcing to pursue a heuristic solution rather than an optimization one (Hulshof et al., 2012).

The present dissertation addresses a staff scheduling problem in the Emergency Medical Services (EMS) field. It is motivated by the real-life setting at Instituto Nacional de Emergência Médica (INEM). The demand at this organization has been increasing, accompanying the growth on the healthcare sector previously mentioned. The quality of a schedule at this emergency medical institution, still performed by hand, directly impacts on the emergency care provision and on the on-job performance. In this study, a mathematical model is built and solved, through exact and heuristic algorithms. Ultimately, it is generated an automatic scheduling application to produce practical solutions used to study the specific case-study at INEM.
This dissertation has a twofold motivation. From an academic perspective, it intends to be innovative and enrich the literature. The model developed is easily adjusted to several real-life problems in distinct application areas, and employs a state-of-the-art solution approach. From INEM view, this application aims to enhance productivity and profitability standards at this institution. By addressing the current scheduling conflicts at INEM, the schedules’ quality is expected to improve.

1.2. Methodology

This Section explains the adopted framework to explore the staff scheduling problem under analysis. It comprises several stages, represented in Figure 1.1.

1. **Characterization and contextualization of the problem**: consists on the comprehension of the medical emergency characteristics, focusing on a main player, that is INEM. The current scheduling situation at this Portuguese emergency institution is presented and described to better comprehend this real-life scheduling problem.

2. **Literature review**: provides a theoretical basis on staff scheduling problems, useful to understand how the current problem should be addressed. Some relevant models’ issues are studied, and the related literature is reviewed, mainly on emergency medical services. Given the problem under study, the application of a mathematical model arises as the more adequate choice.

3. **Data collection and analysis**: based on the former stages, this third stage is done to identify the data that is necessary to incorporate in the model. It involves the treatment and organization of the essential information so that an accurate modelling is accomplished.

4. **Mathematical model**: a mathematical model is developed to a general staff scheduling problem.

5. **Test and validation of the model**: the model is applied to the personnel staff scheduling problem at INEM. The model is tested based on the previously collected data. To ensure the validity of the solution provided by the model, the main results are analyzed and compared to a real-life schedule case at INEM.

![Figure 1.1 - Representative scheme of the methodology adopted throughout the dissertation.](image-url)
6. **Analysis and discussion of the results**: Final considerations are taken regarding the preceding stages of the methodology. The accomplished work is summed up and suggestions for future developments are provided.

1.3. **Thesis outline**

The present dissertation is organized in six main parts. Through this planned structure, the present work seeks to respond to the specific purposes and objectives previously raised up.

An introductory Chapter (Introduction), introduces the motivation and objectives of the work developed as well as the methodology proposed.

The second Chapter (Problem Contextualization) introduces the **Sistema Integrado de Emergência Médica** (SIEM), emphasizing the main stakeholders, namely INEM, that is also presented. A description of the staff scheduling problem at INEM provides an insight on the current problem status, highlighting several opportunities and challenges within it. The main goal is to provide a basic knowledge on the INEM staff scheduling context.

The third Chapter (Literature Review) presents a comprehensive overview on staff scheduling problems. Special attention is given to problems in the medical emergency area. Frameworks, solution approaches and graphical user interfaces from the literature are reviewed. It intends to inform a reader about the state-of-the-art on staff scheduling problems.

The fourth Chapter (Model Formulation) introduces the general Integer Programming (IP) formulation, with its main characteristics. An alternative hybrid formulation to heuristically solve the problem is also presented.

The fifth Chapter (INEM Case-study) comprises the description of the case-study at INEM. The model developed is applied to this real-life problem. All fundamental details and required inputs are introduced. The results of the approaches are compared. Validation of the model is completed by using test sets and a current schedule. At the end, the graphical user interface is shown.

A conclusive Chapter (Conclusions and Future Remarks) draws the main conclusions and suggestions for future work in the sequence of this study.
2. Problem Contextualization

Staff scheduling is identified as the process of deploying timetables for a set of workers within an organization so as to satisfy demand for various services, while simultaneously ensuring a distinctive level of employee satisfaction (Ernst et al., 2004b). Moreover, staff scheduling also needs to consider legal, organizational and contractual constraints (Van Den Bergh et al., 2013).

Staff scheduling is of paramount importance in emergency medical contexts. High concern is deposited to exploit scheduling tools with positive impact in the financial assets of a company without compromising any legal or medical aspect. In an EMS case, scheduling holds both the emergency vehicles systems and the dispatch centers, whose demands are forecast spatially and temporarily (Trudeau et al., 1989).

Chapter 2 describes the staff scheduling problem Instituto Nacional de Emergência Médica, representing a real-life case in an EMS. Section 2.1. summarizes the Sistema Integrado de Emergência Médica (SIEM) main activities and players. Several activities of the SIEM are developed by INEM, which are described in Section 2.2. Then, Section 2.3. discusses the actual scheduling practice at INEM, which presents relevant opportunities for further improvement. Finally, Section 2.4. presents some final considerations.

2.1. Sistema Integrado de Emergência Médica

The SIEM encompasses the coordinated activities from different stakeholders within the emergency medical care delivery structure. An efficient response of the EMS systems is essential to attest the high-quality of the SIEM process.

The activities of the SIEM range from the initial recognition of the emergency until the dispatch of a vehicle to rescue a patient. The first step of the SIEM is the detection of the emergency usually by civilians at the scene. Secondly, the situation is reported by calling the authorities for on-scene aid. Next, there is a basic first-aid, performed before the arrival of the specialized staff and equipment. Once the emergency vehicles arrive at the scene, aid and rescue is provided. Health professionals aid on-scene as well as in transit. Finally, the SIEM tasks also include the transfer and definitive treatment of the patient to the convenient health care unit, which may include medical intervention.

The participants of the SIEM chain include several players such as citizens, dispatchers, national polices, fire services, ambulance crews, doctors and nurses, technical personnel, inter alia. INEM supervises the activities of several of these actors (SIEM, 2017).

2.2. Instituto Nacional de Emergência Médica

Established in 1981, under the Ministry of Health, INEM is a key player of the SIEM, responsible for overseeing and controlling most of the SIEM processes in Portugal (Decreto de Lei 176/81). Aiming to achieve both INEM vision and mission, this public institution guarantees, on time and efficiently, highly specialized health care services in an urgent or immediate situation. Their actions include reception of the emergency request, prompt and accurate assistance at the scene when necessary, and aided
transportation to the convenient health facility. To deliver pre-hospital care at the scene and during transportation in life-threatening situations, INEM requires highly qualified professionals and specialized material resources (INEM, 2017a).

Initially, INEM operated exclusively in the Lisbon area. However, since its foundation, the institution has been experiencing tremendous growth and is nowadays spread throughout Portugal. During 2016, INEM answered 1,370,349 emergency calls, resulting in the activation of 1,280,322 emergency vehicles, corresponding to an increase of 67,391 incoming calls when compared with the previous year (INEM, 2017b).

Accompanying INEM expansion over the last years there are fundamental aspects to be considered. These involve a close management and monitorization of the processes which may have potential consequences in the performance of the emergency stream. Within these, the organization of human resources over the different tasks is crucial for successful outcomes (Bandara et al., 2014). This thesis explores this activity and addresses the scheduling of a specific type of emergency workforce at INEM, known as Técnicos de Emergência Pré-Hospitalar (TEPHs), i.e. technical personnel, who work on the dispatch center and on the emergency vehicles.

2.2.1. Emergency request

In INEM, the emergency medical processes require, from an early stage, a profound cooperation between the emergency dispatch center known as Centro de Orientação de Doentes Urgentes (CODU) and the EMS resources delivery system.

In Portugal, when someone’s life, safety or health is threatened the emergency medical assistance is reached by dialing the European emergency number: “112” (Figure 2.1). The dispatcher answering the call belongs either to the consortium of north or south of Portugal, known as Consórcio Norte and Consórcio Sul, respectively. The former answers calls from regions located to the north of Leiria and Castelo Branco whereas the latter covers the southern part of the country from these two districts. Their responsibility is to recognize the nature of the call such as a fire, robbery or medical situation. In case of medical situation, the call is immediately transferred to a CODU.

![Figure 2.1 - Workflow of a medical emergency request in Continental Portugal, triggered by dialing “112”](image)

---

6
Three CODUs – North, Center and South - operating 24/7 cover Portugal mainland. The CODU acts in direct dependency with the INEM pre-hospital medical vehicles, activating the incoming calls through appropriate triage procedures. In fact, CODU is responsible for establishing a link between the caller requesting emergency medical assistance and the EMS delivery system by acquiring the calls, delivering voice assistance and instructions, and dispatching the convenient transportation to the scene.

At CODU there are distinct specialists, resulting in tasks that are performed by people with different skills. Medical physicians are responsible for clinical counselling during calls, validation of protocols, and occasionally support triages. Psychologists are the ones that acquire the calls from the Centro de Apoio Psicológico e Intervenção em Crise known as CAPIC and deliver psychological support. Lastly, there is a group composed of TEPH, representing the greatest proportion of workers at CODU, among which several responsibilities are considered. When an income call is received, it is necessary a primary recognition of the emergency location, followed by an understanding of the real nature of the situation through a triage procedure. Next in order, it is required activation and dispatch instructions to select and assign the most appropriate vehicle to each scene. The get-together and communication of data from evaluation at the scene and the recovery of non-reached calls are also essential jobs performed at CODU by TEPHs.

2.2.2. Emergency vehicles

Efficient and high-quality medical services are delivered to the target population in an articulated way. For that purpose, INEM requires a specialized mobile resources unit for patients’ care provision and transportation, from the emergency scene to a health facility for treatment. Clinical medical care can be supplied at the scene and during the carriage.

Emergency vehicles (EVs) are physically disseminated over the country at INEM designations and in emergency medical points located at fire stations or at Red Cross centers. To land in the scene rapidly, EVs possess priority while on the road in relation to any other vehicle. In addition, they are easily identified throughout visible markings and specific design characteristics such as catchy colors and highlighted descriptions intend to help their purpose.

In Portugal, the EVs, mostly operated by INEM, enclose the ground and air based interventions. The nature of the situation determines the category of vehicle to dispatch and each sort of vehicle has pre-defined staff requirements to operate. As mentioned, this thesis focusses on the scheduling of TEPHs that crew vehicles operated by INEM coordinaded by the Lisbon center. Among INEM vehicles that request TEPHs, the following staff requirements are necessary to operate:

- Medical Emergency Ambulance (AEM) – staffed by two TEPHs
- Immediate Life Support Ambulance (SIV) – staffed by a TEPH and a nurse
- Inter-hospital Pediatric Transport (TIP) – staffed by a TEPH, a nurse and a physician
- Mobile Unit of Psychological Emergency Intervention (UMIPE) – staffed by a TEPH and a psychologist
- Medical Emergency Motorcycle (MEM) – staffed by a TEPH
Note that there are other vehicles such as the Vehicle of Medical Emergency and Reanimation (VMER), crewed by a nurse and by a physician or the medical emergency helicopters, staffed by two pilots, a physician and a nurse. None of these are staffed by TEPHs, henceforth they are not considered in the present staff scheduling study.

2.2.3. Dispatching and routing

Once a call arrives in CODU and considering the clinical situation of the victims, the proximity to the scene location as well as its accessibility, the medical dispatcher has two options: order a vehicle to the emergency scene or not. The medical response flow is shown in Figure 2.2:

![Figure 2.2 - Medical emergency response at CODU in Portugal continental. Once the call reaches CODU, the dispatcher decides whether the emergency requires medical assistance.](image-url)

For each medical situation, it is compulsory to assign priorities. High priority situations demand, at least, one vehicle prepared with advance life-support equipment along with one common ambulance, i.e. one SIV and one AEM, respectively. Lower priority situations claim merely one AEM. Simultaneously, a different triage process designates the commitment of a pediatric or a psychologist to the episode, therefore, indicative of the need of a TIP and UMIPE, at the local scene, respectively. The use of motorcycle ambulances, MEM, may equally occur for all priorities reckon the distance of the motorcycle to the accident and the urgency to have an emergency vehicle there. MEMs are useful when is necessary to rapidly reach the site of the emergency as these can travel faster than a car or a van ambulance through heavy traffic.

When reaching the scene, the health professionals check and assess the occurrence, leaving the correspondent information available through the Integrated Clinical Ambulance Record system (ICARE) for the CODU, the hospital and for the patient health record.

If there is no need of medical assistance, the call is shifted to the 24-health line – Saúde 24 – responsible for the provision of health insights and guidance by voice. In parallel, calls arriving to the 24-health line may be forward to CODU if an emergency requires emergency medical vehicles. The introduction in Portugal of this line by the former Ministry of Health in 2007 was extremely successful to widespread the access to clinical and non-clinical health information. With this solution, the number of non-urgent
calls at CODU decreased, leaving the CODU personnel more available for upcoming urgent calls (Saúde 24, 2017).

2.3. Staff scheduling problem at INEM

Services at INEM operate on a continuous basis, 24 hours a day, every day of a year. Staff scheduling of emergency personnel such as physicians, nurses, psychologists, TEPHs is, in general, a complex and time-consuming task. The difficulty is not simply attaining an optimal solution but further finding a schedule that most accurately fits demands, cost savings, work and legal rules, individuals' satisfaction, equity issues, among other requirements.

In this study, it is only addressed the scheduling of TEPHs, as they correspond to the greatest slice of emergency personnel coordinated at INEM, therefore the one raising more complications. Other emergency staff such as physicians, nurses and psychologists are fewer in number and not all are scheduled by INEM. Many times, these professionals do not belong to INEM but to other organizations, e.g. hospitals, other health care units, that are in charge of their allocation. The ones actually scheduled by INEM, lead to much simpler schedules when compared to the TEPHs’ schedules.

Each TEPH is allocated to one of the two services, CODU or EVs. Irrespective of the attributed service, TEPHs may sporadically perform tasks at the other service. Each service has a set of tasks to be performed. In the CODU, the location of these tasks is not an issue since all CODU tasks are executed in a room at the INEM headquarters. In EVs, TEPHs may work in the identified emergency medical carriages: AEM, SIV, TIP, UMIPE or/and MEM. EVs are identified by an ID and positioned in a specific location. Here, spatial considerations are well thought-out, that is, each job is characterized not merely by a specific time slot, but also by its location.

To favor management simplicity, each TEPH is allocated to a team of one of the two services, and each team is then committed to a specific set of tasks. This implies that one can expect a certain number of teams in CODU and the same for the EVs. Ideally, TEPHs perform jobs from their own group nevertheless, due to requirements or even subjective preferences, they may be assigned to other teams’ tasks as long as they have the skill to perform the tasks. Regardless of the service every TEPH at INEM maintains a full-time contract with fixed workweek hours, which is, currently, transversal to the public sector in Portugal. Working beyond the contractual hours is possible at an extra cost, along these, overtime is allowed.

2.3.1. Current status

Until not long ago, the communication of schedules at INEM was handmade, consequentially time and resources expenditures were extremely high besides the lack of quality, transparency and failures that were, quite often, evident. Technological adventure through digital solutions enabled the use of enhanced supporting tools and the manual procedure was recently replaced by a more convenient one. The existing platform is an interface to communicate the information yet, until now, cannot be used to generate schedules automatically.
Currently, scheduling is done through an on-line platform, the so-called Gestão de Horários (GH). The authentication page appearance is presented in Figure 2.3 and is accessed by every TEPH after introducing the respective username and password. Once logged in, the access to a personal homepage is authorized.

Figure 2.3 - Authentication webpage of GH platform, which is currently used at INEM to visually display the personnel schedules.

In a first step, every TEPH proposes a schedule for themselves. In doing this, they are expected to adhere to certain constraints, to make sure feasible schedules can be obtained. However, this is something employees must be aware of but they are not alerted or notified by the platform. In a first instance in this step, every TEPH manages the own agenda, that GH visually displays. Flexibility of schedules is possible but somehow limited. For instance, in the CODU service the teams are assigned to a specific sequence of shifts and, consequently a person is expected to perform a task in the day and shift of the correspondent team. Whereas, in EVs there is no sequence of shifts assigned to the working teams and, therefore, the schedules are more volatile from one planning period to another than in CODU.

In a second step, the coordinator of each team manually collects the data from all personnel and checks for infeasibilities. Reassignments are made to meet the pre-defined demand as much as possible. This process can take several iterations until assembling the information from the overall agendas to the desired planning horizon.

Finally, in a third step, the coordinators of the different teams synthesize their respective schedules and again reassignments are made between teams to satisfy the overall demand. This phase obliges readjustments to attain all demand and also entails several attempts until reaching the final schedule. This is a complex and currently hand-made and time-consuming task.

Even with the appealing appearance of the schedules, GH fails in a very relevant and notorious aspect. GH doesn’t contemplate legal, work or personnel warning signs. These considerations need to be checked and audited by the user, whom in a first approach is the regular TEPH and in a next phase the coordinator of the service. Besides, it may easily present biases regarding equity aspects due to the
subjective readjustments done by the coordinators e.g. in case of conflict choices, making it clear that GH system is not exploiting the full range of capability it could achieve.

Nowadays the scheduling of technical personnel for CODU and EV through GH programme is performed independently, in distinct timetables, and the coordinator of the respective service is the one that finishes the respective schedule (Figure 2.4). As stated, TEPHs are allocated to a main service but they can work in the other service if they have the skills and if it is needed to meet specific demands. Hence, the coordinators need to define the schedule for their service and, later, adjust the schedules together for both services. Consequently, it is necessary a permanent communication between the two coordinators to conclude the schedules. The adjustments correspond to direct changes of the workers or, alternatively, attempts to cover empty spots of a certain task in a certain day and shift.

![Diagram of current scheduling process workflow at INEM. The schedules from CODU and EVs are prepared disjointedly by the respective coordinators.](image)

Despite both CODU and EVs agenda start to be planned in the previous month, there is no established deadline to launch them.

Considering the staff scheduling issues at INEM, the main objective of this thesis is to design, based upon mathematical models and algorithms, a programme to support an automated staff tool to provide solutions in a significantly less time and, simultaneously, improving solutions’ performance.

### 2.4. Chapter considerations

EMS is a critical part of the health care delivery system which provides efficient assistance within fast response time in emergency situations (Aringhieri et al., 2017). EMS systems’ performance is defined by the response times, the care provided, and the material used by the technicians. In this context, the scheduling of personnel plays an essential role to guarantee maximum coverage and efficacy of the EMS (Erdoğan et al., 2010). The optimization of staff scheduling should be pursued as for this wisely programmed decision support systems are needed.

The present Chapter has presented an overview of the SIEM stages in Portugal to better understand the context and circumstances in which the study is developed. INEM, the target institution of this thesis, is briefly characterized. The emergency flow from request to dispatching and routing at INEM is shown.
As INEM is committed to accomplish high standards of quality and excellence the goal of this thesis is to create an application to automatically produce schedules to scope a specific type of employees at INEM. This specific range is the TEPHs workers, i.e. emergency technical personnel. The staff scheduling problems with focus on the current situation at INEM are amply scrutinized throughout Chapter 2.

The next Chapter develops a literature review of papers related to the staff scheduling problem, more specifically on the EMS. Main variants and possible problem frameworks on how to solve the scheduling problem are also explored.
3. Literature Review

Staff scheduling is a common problem for most organizations and for that reason has been extensively studied in the literature. Aiming to design efficient and fair schedules, Section 3.1. explores what has been done so far in this field and its main applications, especially in the emergency medical area. Recognizing that there is not a unique methodology to approach a problem, Section 3.2. highlights some relevant methodologies of the literature, which may be helpful to tackle the problem at INEM. In Section 3.3. the main solution approaches are outlined, focusing on recent hybrid approaches that combine exact and heuristic methods. Section 3.4. discusses the importance of constructing interfaces upon the developed algorithms which is, quite often, neglected in the literature. To finalize, the last Section presents some conclusions on the state-of-the-art, and relates the information displayed with the problem of this thesis.

3.1. Staff scheduling problems

Personnel scheduling problems arise in a wide variety of settings, such as transportation systems, call centers, health care systems, protection and emergency services, tourism, retail, manufacturing, among other settings. Scheduling tools are being explored, looking for valuable impacts in companies’ assets while not compromising any labor or performance aspect (Ernst et al., 2004a).

Over the last decades, different literature reviews have been published on staff scheduling problems. In part due to its complexity, it is not possible to attain a unique and consensual problem classification, but rather several classifications are proposed in the literature. Causmaecker et al. (2005) classify staff scheduling problems into four categories: (1) permanence centered planning, consisting of problems where the volume of personnel required is known in advance; (2) in fluctuation centered planning, demand changes throughout the day, such as in call centers or fast food restaurants; (3) mobility centered planning, the case of e.g. railway and airline companies, where tasks involve transportation of workers from one place to another; (4) finally, in areas such as consultancy, work is typically divided into projects to which different groups of employees are assigned, and this belongs to the project centered planning category. The staff scheduling problem at INEM is a mix of the first three categories, as the required number of personnel for each task is acknowledged a priori, but differs between morning, afternoon, or night shifts, and different teams operate in different locations.

In a more recent work, Brucker et al. (2011) present generic mathematical programming formulations for permanence and fluctuation centered planning and for the rostering part of the mobility centered planning. The authors identify general scheduling problems as either NP-hard or polynomial solvable. A common NP-hard problem described is the Nurse Scheduling Problem (NSP). In these, nurses, with diverse skills, are assigned to working shifts over a specific time frame. The allocation of nurses needs to accomplish specific requirements such as demand of the requested shifts, legal and contractual aspects and also personal preferences. Indeed, the NSP is a widely studied problem in personnel scheduling and corresponds closely to this thesis problem as the goal is to allocate the technical personnel of INEM to certain tasks given various requisites and restrictions. Reviews of models and solution methods focusing the NSP are provided by Cheang et al. (2003) and Burke et al. (2004).
Finally, it is worth mentioning an annotated scheduling bibliography of ca. 700 articles with a short summary of each paper presented by Ernst et al. (2004b). The papers are classified according to the type of problem addressed, the application areas covered, and the methods used.

3.1.1 Staff scheduling in EMS

In emergency medical institutes, staff scheduling is of paramount importance, since shortages in the number of required personnel directly impact the quality of care that patients and communities receive. Additionally, the shift sequences assigned to personnel also impact on their competencies. If someone is e.g. assigned to a morning shift after doing a night shift the previous day, this can lead to so-called jet fatigue (Kreeft, 2012). Furthermore, employee satisfaction must not be neglected, as undesirable schedules can lead to increased staff turnover (Cline et al., 2003).

The EMS planning consists of two fundamental categories of staff: the staff from the dispatch center and the staff working in the emergency vehicles. Depending on a country legislation and context, the schedules of these two staff categories may be done altogether or separately (Van Den Berg, 2016). In the present case at INEM, the schedules are built separately even though the workforce is partially shared, meaning each TEPH is assigned to one of the two mentioned categories, but some TEPHs can sporadically perform tasks at the other service if they have the proper skills to do so.

In general, plenty of papers have been published to address the scheduling of personnel, but only some of them focus on the EMS. Li and Kozan (2009) define two hierarchical phases for solving a rostering problem using a nonlinear integer programming technique. A first model defines the shifts as well as the necessary emergency vehicles’ staff, in number and location, so that the volatility between supply and demand is minimized. The second phase uses the outcome of the first one to allocate ambulance crews to the pre-defined shifts and demands. The model aims to minimize the working units through a complete roster as it includes not only organizational rules but also individual preferences. The integration of these preferences turns the problem NP-hard. The results exhibited that this second phase, when applied to 50 ambulance crews over a monthly schedule, takes only few minutes to run using Lingo.

All in all, the staff scheduling in emergency services do share characteristics with other related problems such as the NSP, so deeply explored in the literature. However, based on the visited literature and to the best of knowledge, there is no problem that can be straightforward applied to the INEM situation without major transformations and, consequently, the work developed in this thesis intends to enrich the literature in this field.

3.2. Methodologies

In staff scheduling problems, specific features are translated into a mathematical formulation through an accurate modelling. For this propose, Blöchliger (2004) proposes a four-stage tutorial concerning the modelling aspects (Figure 3.1), struggling to be as wide-ranging as possible so it can be modified for comparable problems. Primarily, the significant data is gathered and defined. Within this the correspondent notation is introduced: sets, subsets, parameters and decision variables. Secondly, different constraints are studied: these can be either hard if they must be satisfied at all costs, or soft in
the case they should be met but their fulfillment is not strictly mandatory. This stage is followed by the definition of the objective through the objective function subject to the problem constraints. Depending on the problem, the objective may be maximized or minimized. Finally, the fourth stage assembles the information from the three previous stages. It consists of the final formulation of the model, accurately depicting the scheduling problem, so that a good or an optimal solution is able to be searched throughout a convenient solution approach.

Regarding the stage of collecting relevant data, Van Den Berghe (2013) states the importance of specifying, in an initial phase, the main features of a problem. These include the time interval of the schedules, likely to range from few days or weeks up to months or even years; the operating hours of the organization which in the EMS specific case is a 24-hours continuous basis; or the shift type, such as the starting-times and lengths, possibility of overlapping, cyclicity, breaks. The workforce is also a matter, particularly in terms of skills, contract type (e.g. full time, part time, daily regime) and personal preferences.

The constraints rely upon the problems’ specific characteristics and settings. The list available in Table 3.1 considers common constraints found in the literature (Cheang et al., 2003).

**Table 3.1 - Set of possible constraints found in a staff scheduling problem (Cheang et al., 2003).**

<table>
<thead>
<tr>
<th>Coverage Requirements</th>
<th>Time Related Constraints</th>
<th>Work Regulation Constraints</th>
<th>Internal Ward Constraints</th>
</tr>
</thead>
<tbody>
<tr>
<td>Skill levels and categories</td>
<td>Workload (minimum/maximum)</td>
<td>Preferences or requirements</td>
<td>Shift-patterns</td>
</tr>
<tr>
<td>Shift type(s) assignment (max shift type, requirements for each shift types)</td>
<td>Consecutive same working shift (minimum/maximum/exact number)</td>
<td>Number of free days (minimum/maximum/consecutive free days)</td>
<td>Historical record (e.g. previous assignments)</td>
</tr>
<tr>
<td>Constraints among groups/type of workers (workers must work together or must work separately)</td>
<td>Consecutive working shift/days (minimum/maximum/exact number)</td>
<td>Free time between working shifts (minimum)</td>
<td></td>
</tr>
<tr>
<td>Other requirements in a shorter or longer time period other than the planning period, e.g. every day in a shift must be assigned</td>
<td></td>
<td>Holidays and vacations (predictable)</td>
<td></td>
</tr>
<tr>
<td>Constraints among shifts e.g. shifts cannot be assigned to a person at the same time</td>
<td></td>
<td>Working weekend e.g. complete weekend</td>
<td></td>
</tr>
<tr>
<td>Requirements of workers or staff demand for any shift (minimum/maximum/exact number)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Accordingly, coverage requirements comprise the personnel demands for every day and shift over the planning horizon. The contractual workload, the number of consecutive same working shift or the number of consecutive working shift/days are included in the time related constraints. Restrictions related to personal schedules such as individual requests, preferences or number of free days are considered in the work regulation category. Finally, some specific practices applied by wards are considered in the set of internal ward constraints. Moreover, Cheang et al. (2003) also provides a list referencing works for each defined constraint.

From one problem to another, the objectives are mutable and highly dependent on the context. A review by Burke et al. (2004) presents some plausible objectives for staff scheduling problems such as to minimize: personnel cost; “under” coverage, employees’ number; deviation between scheduled employees and demands, etc.

The objective considered in a model gives rise to a mathematical formulation. As previously mentioned, while the hard constraints must be satisfied to obtain a feasible solution, the soft ones should only be satisfied, as much as possible. Hence, it is quite common to find models in which operators assign penalties associated to the violation of the soft constraints which are normally weighted in the objective function. In these cases, all the soft constraints ($C$) are penalized in the objective function, typically corresponding to a linear combination of the number of violations ($\text{#violations}_{n,c}$) and the weight of each soft constraint ($\text{weight}_c$) as shown in Equation 3.1. The weighted sum is then maximized or minimized based on the nature of the problem (Smet et al., 2013).

\[
\text{Weighted Sum} = \sum_{n \in N} \sum_{c \in C} \#\text{violations}_{n,c} \cdot \text{weight}_c
\]  

For solving the NSP, several works in the literature have used analogous strategies. For example, in Kreeft (2012), it is illustrated a model with a cost associated to the nurses’ overwork. In this situation, the hospital would need to pay an additional wage for every overtime hour and the hospital would incur in legal fines beyond a certain number of extra-hours. From the hospital point of view, overwork should be avoided, as possible to all nurses. Hence, extra-hours are weighted in the objective function which is to be minimized. For this case, slack variables are used to obtain the penalty cost function. Penalty coefficients are selected to weight every single slack term.

Other alternatives to explore the staff scheduling problems are proposed in the literature. Van Den Bergh et al. (2013) look at four different sets of problem characteristics, namely (i) personnel characteristics, decision delineation, and shifts definition, (ii) constraints, performance measures, and flexibility, (iii) solution method and uncertainty incorporation, and (iv) application and applicability of research. The review aims to easily compare distinct problems, allowing benchmark of the problems by pointing out their similarities and variants among them. At INEM the problem can be categorized in the different sets
as follows: (i) every TEPH has full time contracts with tasks assignment. The shifts are non-overlapping and have fixed length and starting times; (ii) coverage constraints are soft as under and overstaffing are acceptable while the possession of skills is modeled as a hard constraint. There is no flexibility to schedule breaks apart from what is predicted by the law. Several time related constraints are hard such as the maximum number of consecutive days; (iii) the solution technique is based on mathematical programming and the problem is considered deterministic since workload is based on past records; (iv) the application area is the medical emergency area, which can be approximated to other types of problems.

To conclude, no single framework can be defined as the only approach to tackle the INEM problem. Consequently, in this thesis, it is used a combination of approaches to ensure that all the relevant characteristics are included in the mathematical model and that the solution approach provides good quality schedules in useful time.

3.3. Solution approach

Once the formulation is settled, several solution techniques can be applied to solve the personnel scheduling problems. A board classification divides these techniques into exact and heuristic methods. While the former guarantees optimality of the solutions, the latter does not, but is quite convenient if an optimal solution is not mandatory, speeding up significantly the processes. Recently, researchers have tried to incorporate both methods using the so-called hybrid methods (Cheang et al., 2003).

The exact solution methods are the firsts that should be considered. Within these, a problem is organized, and the fundamental properties are deeply analyzed. This step is helpful to understand the complexity of the problem so that all essential insights are well-thought-out. However, according to Beliën (2005), optimal methods tend to be too demanding in computational aspects or very difficult to apply in real-life scenarios so alternatively, to overcome these hurdles, the problems are addressed by heuristic methods. Heuristics are usually a good choice, especially when optimal solutions are extremely difficult to obtain, but these methods involve exhaustive testing to be validated (Burke et al., 2004). In most contexts, a heuristic algorithm may be attained by adapting an original exact algorithm, and by accepting an optimal gap that relates the resulted solution with a theoretical optimum (Beliën, 2005).

3.3.1. Exact algorithms

Mathematically exact algorithms are applied to find an optimal solution of real world-problems. Within exact methods one may consider Linear Programming (LP) and Integer Programming (IP) techniques. LP techniques are used to solve problems with continuous decision variables, whereas IP techniques provide integer solutions too (Ernst et al., 2004b). Mixed Integer Programming (MIP) appears as a third option combining continuous and discrete decision variables at the optimal solution (Beliën, 2005). In general, staff scheduling problems entail formulations with many variables, which is advantageous to obtain a tighter bound relaxation and to discard symmetry (see Barnhart et al., 1998), however it averts to efficiently solve a problem with a common commercial MIP solver. Consequently, to accommodate large-scale problems other exact approaches have been explored to solve the LP relaxations of these problems.
A straightforward approach is to formulate the problem as an integer programming model and solve it using a general IP solver (Bard, 2003; Isken, 2004). Despite this, researchers have employed other solution approaches such as branch-and-price, which combines Column Generation (CG) with branch-and-bound (Bard, 2005a; Bard, 2005b; Burke, 2010b). This approach has gained considerable attention over the last years and is characterized by using CG in each node in the branch-and-bound tree to solve the LP relaxation.

![Diagram of Column Generation scheme in a node of the branch-and-bound tree (RC = Reduced Cost)](adapted from: Hans, 2001).

The CG scheme in a single node is schematized in Figure 3.2. An optimal solution to the master problem is also optimal to the LP relaxation. Henceforth, after solving the master it is necessary to check the optimality of the LP solution. For this propose, it is solved a subproblem known as pricing problem that identifies the columns that should be considered (i.e. enter the basis). If the solution of the pricing problem is not feasible, there is at least one violated constraint detected by the pricing algorithm, which is to say it exists at least one column with negative Reduced Cost (RC). The columns under these circumstances are added to the master, and the master problem is reoptimized. The scheme terminates when the LP relaxation is solved, which happens when there are no more columns with negative RC (Hans, 2001). The book by Desrosiers (2005) comprises a full chapter on the use of Column Generation for solving problems.

Generally, after solving the CG scheme, the solution arising is fractional and therefore not feasible. Branching is then performed to find a feasible solution to the IP problem. Vanderbeck (2000) addresses some branching schemes and their respective implications on the CG scheme. The author shows the possibility to keep the subproblem conformity and, simultaneously, eliminate the fractional solutions that have been found. A compromise between branching efficiency and subproblem tractability needs to be considered. Barnhart et al. (1998) present a general methodology for branch-and-price which unifies the existing literature. Finally, Hans (2001) clarifies ways to speed up a branch-and-price algorithm, e.g. by reducing the number of analyzed nodes of the tree. However, when decreasing the size of the branch-
tree, the integrality gap would increase, reducing the accuracy of a solution. Some other techniques addressed in the literature have shown to be relevant (see e.g. Desrosiers, 2005).

### 3.3.2. Heuristic algorithms

Despite the vast improvements in computer hardware and commercial solvers over the last decades, staff scheduling problems remain difficult to solve to optimality. Furthermore, optimal solutions that require many hours to be calculated are often less valuable than quick suboptimal solutions, which allow easier user feedback or sensitivity analysis (Cheang et al., 2003). Moreover, heuristics are relatively simple to implement and able to handle complex constraints and objectives (e.g. non-linear cost functions). This has led researchers to a widespread application of heuristics in personnel scheduling problems (Ernst et al., 2004a).

According to Cheang et al. (2003), classical approaches include shuffling, where a schedule is readjusted and enhanced after solving it to the worst-case scenario, and greedy shuffling, where all possible shuffles for every person are generated and listed from the highest to the lowest cost benefit schedule.

Other non-classical approaches such as metaheuristics have been developed and proven high-efficiency in finding suboptimal solutions for a vast set of problems. Genetic algorithms are a popular type of metaheuristics (Cai and Li, 2000; Aickelin, 2000; Aickelin, 2004; Pato and Moz, 2008; Moz and Pato, 2007; Puente et al., 2009). Cai and Li (2000) address a staff scheduling problem with heterogeneous skilled workforce using a genetic algorithm. A multi-criteria model comprising three main objectives is used to formulate the model. Firstly, it seeks to minimize the total costs for assigning the necessary workers to satisfy the pre-defined staff requirements. The second criterion aims to achieve maximum staff surplus. Thirdly, there is an intention to minimize the variation of staff excess over the schedules. Computational tests applied to a local labor-intensive organization have demonstrated effectiveness of the proposed approach to find good solutions. Tabu search is also explored in several papers (Bellanti, 2004). Other metaheuristic approaches include particle swarm optimization (Altamirano, 2012), memetic algorithms combining genetic algorithms with local improvement procedures (Burke et al., 2001), and hyperheuristics (Smet, 2014).

### 3.3.3. Hybrid algorithms

In order to explore the best features of the previous algorithms, the combination of different procedures is becoming rather popular and promising (Van Den Bergh et al., 2013).

Recently, MIP-based heuristics have been successfully applied to the NSP (Burke et al., 2010a; Valouxis et al., 2012; Santos et al., 2016). These strategies aim to combine the strengths of both MIP and metaheuristic approaches. Burke et al. (2010a) first solve the NSP using an IP model that includes all hard constraints and only a subset of soft constraints. This relaxed IP problem is solved to optimality using CPLEX, at significant fast runtimes. Then, in a second step, the solution obtained by the IP model is improved using a variable neighborhood search (VNS) which focuses on the satisfaction of the other soft constraints that were not treated in the preceding IP model. This framework combining the benefits
of IP and VNS, shows satisfactory computational results when applied to a real instance at a Dutch hospital. Valouxis et al. (2012) also uses a two-phase approach, resulting from the partition of a larger problem. In the first phase, nurses are assigned to days without considering shifts, thus, defining the basis workload of each nurse. The planning period is divided into groups of seven days, and for each group an IP model is solved. The resulting local solution is then improved using local search by recombining partial schedules. In the second phase, an IP model is solved for each day assigning nurses to shifts. This phase only takes into account costs related to shift assignments. This sequence is repeated until the available computation time is exhausted. The possibility of reproducing the model for other scenarios is a strength of this research, since soft constraints may be easily adapted regardless of the IP model. Santos et al. (2016) also use a two-pronged method, consisting of cut generation to improve bounds and prove optimality, followed by the use of primal heuristics to quickly find good solutions. Starting from a feasible initial solution, the authors use a variable neighborhood descent consisting of two different neighborhood moves to improve the solution. The first neighborhood fixes all assignments except for a number of days. The subproblem is then solved to optimality using IP tools. Subsequently, the sequence is repeated for a different subset of days. In a second step, the second neighborhood fixes all allocations of shifts, except one. Again, for every shift a subproblem is solved using an IP model. From this research, good primal and dual bounds have been achieved. Moreover, the authors also exhibit that the best-known solutions are improved by up to 15% when compared to previous instances.

Instead of using standard MIP approaches inside a heuristic framework, Column Generation can be used as well (Gamache et al., 1999; Joncour et al., 2010; Gomes et al., 2017). Gomes et al. (2017) explore the NSP by applying CG through an IP formulation in a problem with a high number of variables. Then, CG is combined with a VNS heuristic to efficiently find columns with negative RC. This search speeds up the convergence rate of the CG method. In terms of results, the developed algorithm shows high performance on all the 29 instances tested for a planning horizon of 4 weeks. Moreover, it improves the best-known solutions, obtaining a gap in the order of 8% for all the instances tested.

On the other hand, CG may also be combined with other heuristics such as a diving heuristic, proposed by Joncour et al. (2010). In an optimization problem, diving heuristics can be used to obtain integer feasible solutions. This method heuristically selects a branch in the branch-and-price tree driven by a greedy, random or rounding principle. Within rounding, one may use strategies such as rounding down, up, to the closest integer, or based on a threshold, until the first integer solution is found. In contrast to branch-and-price algorithms where branching is usually done on the original variables to preserve the structure of the pricing problem, in diving heuristics often one or more columns are fixed to one so that the remaining solution space is reduced much more and therefore speeding up the process (Joncour et al., 2010). After each branching decision, new columns are generated for the people for which no column has been fixed yet, as it is unlikely that the columns of the root node can be combined into a good or even feasible integer solution (Barnhart et al., 1998).
3.4. Graphical user-interface

An important aspect in the acceptance and implementation of a new scheduling method is the availability of an easy-to-use application with an attractive graphical user-interface (GUI). Despite many different models having been proposed in the literature to solve staff scheduling problems, only around 30% are implemented and applied in practice (Kellogg and Walczak, 2007).

According to Van Den Bergh (2013), the lack of implementation arises from the complex nature of the problems’ settings. Additionally, models usually fail in some real arguments namely taking personal preferences into account. Difficulties also appear in integrating algorithms in the software system of a company. Here, a clear description and effortlessly interpretation of the developed algorithm makes it easier for a company to integrate an algorithm. In this review, the author also emphasizes the tendency, with very little exceptions, to compare the quality of the solution approaches with specific real-world data or to theoretical data, rather than using benchmark instances or variations of the real-world data.

The ways to develop user interfaces out of a model range from simple to very complex ones. Manual processes such as spreadsheets tend to be simpler to put into practice but limited for problems with larger dimensions. Unlike this, mathematical models can be implemented in computer software packages within friendly and ease-to-use interfaces (Ernst et al., 2004b). Ho and Johnson (1998) developed a software tool for airline crew scheduling and fleet assignment problems. A software relying upon the Microsoft Access database product was used to implement a user interface, to accumulate relevant information and to connect to CPLEX solver to produce solutions.

The present thesis fills this implementation gap found in the literature, by developing a graphical interface that allows the implementation of the proposed algorithm at INEM. The model needs thus to be validated and constantly readjusted in close collaboration with the practitioners at INEM.

3.5. Chapter considerations

Reckon the importance of health systems, there is an extensive number of works related to the scheduling topic accessible in the literature. Applicable studies published in this area were analyzed. The objective is to obtain background on the topic and present studies that share aspects with the problem under analysis in this thesis.

An overview of staff scheduling problems is initially developed, with some proposed classifications. Scheduling can directly impact human lives, so there is a strong component of medical considerations influencing the formulation of the problem (Trudeau et al., 1989).

Some frameworks with great applicability in real-life situations were introduced. The computational burden arising from problems with large dimensions or hardly constrained has been described as well. Nowadays sophisticated heuristic algorithms are able to deal with demanding problems, and became dominant over former mathematical programming approaches. A trend verified is the development of hybrid approaches for specific applications. Undoubtedly, there is room for these recent approaches, which represent a novelty to solve staff scheduling problems. A crucial aspect to consider after
developing a model is its implementation in practice. Conceiving a graphical user interface is challenging, but without it a model remains theoretical, therefore is kept with meaningless utility.

Based on the literature review presented in this Chapter and on the problem characteristics previously described the problem in study will be tackled by formulating the problem as an IP model that will be solved through a hybrid approach. In the following Chapter, the problem formulation to solve the case study at INEM is presented.
4. Model Formulation

Chapter 4 introduces the model developed to formulate the problem analyzed in this thesis. As this research is motivated by the specific personnel scheduling problem at INEM, Section 4.1. explains the main features of the staff scheduling setting for a deeper understanding of the problem. Section 4.2. proposes a standard integer programming (IP) formulation. For this purpose, the notation is introduced, followed by the formulation of the model, which include the linear objective function subject to a set of hard and soft constraints. Section 4.3. suggests a hybrid solution approach based on CG which uses a diving heuristic to find an integer solution. The strategies considered include two CG schemes as well as two branching methods. Finally, Section 4.4. draws some conclusions.

4.1. Problem statement

The proposed staff scheduling model intends to be flexible and general enough so that it can be easily adapted to solve diverse real-world problems as there are resemblances and common features among different scheduling problems.

The problem corresponds to a 24/7 shift-scheduling problem concerning personnel from different services. The workforce is individually assigned to one service. Each service is divided in a number of teams. To these teams belong both a set of tasks and a set of people. The idea is to assign people to tasks, preferably in their own team but it is also possible to assign to tasks from other teams to meet the required demand. This can only occur if a worker has the required skills for a given task.

In their nature the shifts do not overlap and have fixed lengths and starting times. Each day of the planning horizon is divided into three working shifts in which the tasks are performed:

- Night shift (N) – from 0:00 a.m. to 8:00 a.m.
- Morning shift (M) – from 8:00 a.m. to 4:00 p.m.
- Afternoon shift (A) – from 4:00 p.m. to 0:00 a.m.

The model allows that some tasks have duration that differ from the shift length. The task then starts at the same time as the shift it is assigned to, but can finish either before or after the end of the shift.

Some working patterns are undesirable and therefore tried somehow to be avoided. If e.g. someone does an afternoon shift it is not convenient to work a morning shift on the following day. There are also unpopular shifts such as weekend shifts or holidays' shifts, which must be covered, but employees prefer to avert them.

The main objective is to make feasible schedules, assigning people to tasks in a certain day and shift. When preparing the schedules, the primary goal is to ensure functionality of the services. Each task has a certain workforce demand. This is known in advance and vary accordingly to the forecasted supply. In addition, legal rules include a minimum resting time between each pair of shifts worked. Working time regulations limit the maximum number of consecutive working days and of consecutive days off for each person. Workers must have a minimum number of Sundays off over the planning period. Furthermore, it is necessary to guarantee that employees are only assigned to tasks which they are able to perform.
The staff must feel motivated and engaged so a schedule should aspire to be as equitable as possible for every worker: (1) to balance the schedules, every person needs to work at least a foretell number of night, morning and afternoon shifts; (2) to ensure workforce well-fare, there is an effort so that employees have the entire weekend off instead of a lone day of the weekend; (3) there is a number of contract hours that should be met, meaning both overtime and undertime are undesirable; (4) finally, even though a person may perform tasks outside the own set of tasks in case it contributes to satisfy coverage requirements, this should be minimized.

Apart from contractual, legal and fairness aspects, there are supplementary characteristics to consider when building a schedule, such as the definition of the planning frame i.e. the number of days of a schedule. Other relevant aspects consist on the identification of the first weekday of the schedule and the number of weekends within it. Furthermore, if there are public holidays or other publicly recognized non-working days, there will be a decrease on the overall number of working hours for each employee in that specific planning period.

Finally, in terms of working agreements, the model may consider a certain number of working hours for each person, individually. The workload is distributed in a full-day basis, meaning that every person is expected, apart from extraordinary exceptions, to work in every sort of working shift. Privileges due to seniority, such as the choice of a more favorable shift work or the preference for a deemed relaxed work, are not considered. Pairing preferences among workers and breaks are not treated too.

4.2. Standard IP formulation

4.2.1. Notation

Notation is first introduced, including sets, subsets, parameters and decision variables.

4.2.1.1. Sets

The following sets are defined:

- \( i \in I \): the set of people
- \( t \in T \): the set of tasks
- \( d \in D \): the set of days in the planning horizon
- \( w \in W \): the set of full weekends in the planning horizon
- \( s \in S \): the set of shifts, i.e. \( S = \{ \text{night, morning, afternoon} \} \)
- \( g \in G \): the set of working teams
- \( j \in J \): the set of working services

4.2.1.2. Subsets

The following subsets are defined:

- \( I^t_i \): the subset of people that can perform task \( t \)
- \( I^g_i \): the subset of people that belong to team \( g \)
- \( T^g_s \): the subset of tasks that belong to team \( g \)
4.2.1.3. Parameters

The parameters of the model are now defined. These represent the input data necessary to run the model.

- $k$: starting weekday of the planning horizon (0 = Monday, 1 = Tuesday, ..., 6 = Sunday)
- $\xi$: number of holidays of the planning horizon
- $\theta_i$: number of contract hours of person $i$
- $\eta$: number of hours to discount from the contract hours $\theta_i$ per holiday $\xi$
- $L_t$: duration of task $t$
- $R_{tds}$: required number of people to be assigned to task $t$ on shift $s$ and day $d$
- $\theta^1$: maximum number of consecutive working days
- $\theta^2$: maximum number of consecutive days off
- $\theta^3$: minimum number of Sundays off
- $\theta^4$: minimum number of shifts of type $s$ necessary to be worked
- $w^R_j, w^L_j$: weight of penalty variables for excess and shortage workforce supply in service $j$, respectively
- $w^W_i$: weight of penalty variable for full weekend off for person $i$
- $w^H_i$: weight of penalty variable for excess and shortage hours worked, respectively
- $w^O_g$: weight of penalty variables for assigning tasks of a team to members of another team in service $j$

4.2.2. Decision variables

The decision variables and respective domains are as follow:

- $x_{itsd} \in \{0, 1\}$: equals 1 if person $i$ is assigned to task $t$ on shift $s$ and day $d$, 0 otherwise
- $Y^R_{tds}, Y^L_{tds} \in \mathbb{N}_0$: penalty variables for excess and shortage of workforce supply for task $t$ on shift $s$ and day $d$, respectively
- $Y^W_{iw}, Y^W_{iw} \in \mathbb{N}_0$: penalty variables for full weekend off for person $i$ in weekend $w$
- $Y^H_i, Y^H_i \in \mathbb{N}_0$: penalty variable for excess and shortage hours worked for person $i$, respectively
- $Y^O_g \in \mathbb{N}_0$: penalty variables for assigning tasks of a team $g$ to members of another team
4.2.3. Objective function

The objective function (4.1) represents the objective of the problem and is a function of some of the parameters and decision variables.

\[
\text{minimize:} \quad \sum_{d \in D} \sum_{s \in S} \sum_{j \in J} \left( w_{j}^{RE+} \gamma_{tds}^{RE+} + w_{j}^{RE-} \gamma_{tds}^{RE-} \right) + \sum_{i \in I} \sum_{w \in W} w_{i}^{WO} \left( \gamma_{iw}^{WO+} + \gamma_{iw}^{WO-} \right) \\
+ \sum_{i \in I} \left( w_{i}^{H+} \gamma_{i}^{H+} + w_{i}^{H-} \gamma_{i}^{H-} \right) + \sum_{j \in J} \sum_{g \in G_{j}} (w_{j}^{g} \gamma_{g}^{g})
\]

Soft constraints are weighted in the objective function, and therefore if these constraints are violated, there is a cost associate to it. Each term in (4.1) corresponds to a penalty, weighted correspondingly. The first term comprises the penalization for under and overstaffing supply. The second term is the penalty for full weekend off. The third is related to under and over hours worked by a person. Finally, the fourth term is the penalty assigned to perform tasks outside the respective team. The overall objective is to minimize the weighted sum of all these penalty variables, thus minimizing the real impact of violating the soft constraints.

4.2.4. Constraints

The sets of constraints that define the problem are presented in this Section. These constraints contemplate the insights gathered from the problem statement phase.

Coverage requirements

Equation (4.2) translates the coverage requirement for each shift. Every task in each day and shift has specific workforce demand: \( R_{tds} \).

\[
\sum_{i \in I_{t}} x_{tds} - \gamma_{tds}^{RE+} + \gamma_{tds}^{RE-} = R_{tds}, \quad \forall \ t \in T, d \in D, s \in S
\]

The total number of staff assigned to task \( t \), for a given day and shift of the planning period, should be equal to the required workforce demand, \( R_{tds} \). In case it does not match the supply, there is either an under (\( \gamma_{tds}^{RE-} > 0 \)) or overstaffing situation (\( \gamma_{tds}^{RE+} > 0 \)). This variable is penalized in the first term of the objective function (4.1). Therefore, under and overstaffing are both allowed, but at a cost.

Resting time between two assignments

Given that not every shift sequence is acceptable, common hard constraints in IP problems are the so-known sequence constraints. Usually it is legally compulsory to have a minimum period of resting hours between consecutive working shifts.
In the present model, the shift restriction forces each worker to have a gap of at least two resting shifts until working again. As previously mentioned, the sequence of shifts considered in a day is night, morning and afternoon. Three possibilities must be avoided for every person in each day and shift. Constraints (4.3) prevents a person from working more than one shift a day. Expression (4.4) only allows a person to work a single shift amidst a morning and an afternoon shift of a certain day \( d \) and a night of the immediate next day, \( d+1 \). Finally, (4.5) states that the sum of the assigned tasks for every person in an afternoon of a certain day \( d \) and in a night and in a morning of the next day, \( d+1 \), cannot exceed 1.

Note that for constraints (4.4) and (4.5) the last day of the planning period – \( |D| \) – must be excluded from the analysis since these constraints are checked for every pair of days: a given day and the following day – \( d \) and \( d+1 \), respectively. Consequently, the following day of the last day in the planning horizon (\( |D| \)), \( |D|+1 \), is out of the planning period. Figure 4.1. illustrates this idea for a generic planning period with \( |D|=28 \) days. For every pair of days, the respective shifts are evaluated according to (4.4) and (4.5), ensuring that these hard constraints are not violated. For example, if \( d=1 \), the equations are valid for days 1 and 2. Therefore, the number of pairs of days to be constrained are \( |D|-1=27 \).

$$\sum_{t\in T} (x_{it,\text{night}} + x_{it,\text{morning}} + x_{it,\text{afternoon}}) \leq 1, \quad \forall i \in I, d \in D \quad (4.3)$$

$$\sum_{t\in T} (x_{it,\text{morning}} + x_{it,\text{afternoon}} + x_{it,d+1,\text{night}}) \leq 1, \quad \forall i \in I, d \in D \setminus \{|D|\} \quad (4.4)$$

$$\sum_{t\in T} (x_{it,\text{afternoon}} + x_{it,d+1,\text{night}} + x_{it,d+1,\text{morning}}) \leq 1, \quad \forall i \in I, d \in D \setminus \{|D|\} \quad (4.5)$$

![Figure 4.1](image-url) - Illustration of the set of days to be constrained in (4.4) and (4.5), when considering a sample with 28 days.

**Tasks assignment**

An important aspect to consider when designing schedules is the individual skills of the workers. Perhaps some employees do have the skills to perform every task. But, there may be others who are not able to perform every task.
Constraints (4.6) avoid each worker to be assigned to tasks they have no skills to perform. These constraints must be applied for every person, in every slot of the planning period.

**Consecutive working days**

Usually organizations impose that people cannot work more than a certain amount of days uninterrupted. Hence, expression (4.7) considers an upper limit for the number of consecutive working days for each employee:

\[
\sum_{t \in T} \sum_{i \in I} \sum_{d, s \in S} x_{its} \leq \theta^1, \\
\forall i \in I, d \in D \setminus \{\lfloor |D| \rfloor - 1, \ldots, |D| - \theta^1 + 1\}
\]

For every person and for every set of \(\theta^1\) consecutive days of the planning period, the sum of all working tasks cannot exceed \(\theta^1\).

For the same reason pointed for constraints (4.4) and (4.5), the last \(\theta^1\) days of the planning period must be excluded. For example, if \(|D| = 28\), and it is not possible to work more than 6 consecutive days \((\theta^1 = 6)\), it yields \(D \setminus \{28, 27, 26, 25, 24, 23\}\). The maximum value for \(r\) in the summation is 22, and \(d + \theta^1 = 22 + 6 = 28\) is the last day of the planning period to be screened.

**Consecutive days off**

There is usually an upper limit for the number of consecutive days off, formulated in the expression (4.8).

\[
\sum_{t \in T} \sum_{r \in \{d, d+1, \ldots, d+\theta^2\}} \sum_{s \in S} x_{its} \geq 1, \\
\forall i \in I, d \in D \setminus \{\lfloor |D| \rfloor - 1, \ldots, |D| - \theta^2 + 1\}
\]

It is stated that every person in every set of \(\theta^2\) consecutive days must have at least one working day. For example, if it is forbidden to enjoy more than 5 consecutive days-off \((i.e. \theta^2 = 5)\), for a planning of 28 days \((|D| = 28)\), it yields \(D \setminus \{28, 27, 26, 25, 24\}\). The last day to be check is \(d + \theta^2 = 23 + 5 = 28\) (the last day of the planning period).

**Number of Sundays off**

Regardless of the number of Sundays included in a timetable, employees cannot have less than \(\theta^3\) Sunday(s) as a day-off. This hard constraint is specified through inequations (4.9).
Note that $k$ in (4.9) refers to the starting week day of the planning period, ranging from 0 to 6, which directly influences the number of Sundays within a schedule. So, if the planning starts on a Monday (i.e. $k = 0$) the set of days to screen are $d = \{7, 14, \ldots\}$. As the days are counted as Monday ($d=1$), ..., Sunday ($d=7$), index $d$ belongs to the subset of Sundays of the planning period.

Let’s suppose that it was necessary to have at least one Sunday off ($\theta^3 = 1$) in the entire planning and assuming 4 weeks, $|W| = 4$. Under these circumstances, every person could work 3 Sundays at maximum, which is to say at least one Sunday is off. Or in case $|W| = 5$, the maximum number of Sundays a person could work would be 4.

**Levelled shift types**

A challenge when building rosters is to balance the type of shifts each person works. Workers do share the same rights and responsibilities so fairness and flatness among the type of each shifts worked is indispensable. Indeed, the present model considers that each person must work a minimum number of shifts of each shift type ($\theta^4_a$) – i.e. night, morning and afternoon – during a planning period. These is formulated in constraints (4.10).

$$\sum_{d \in \{7-k, 7-k+7, \ldots\}} \sum_{s \in S} x_{itsd} \leq |W| - \theta^3, \quad \forall i \in I$$  \hspace{1cm} (4.9)

Constraints (4.10) comprise the abovementioned hard requisite: the model must ensure that for every person and for every shift type, the assignments are equal or greater than a certain number, $\theta^4_a$. These aims to acquire unbiased and even schedules.

It is common that the compliance of these conditions allows employees to receive a bonus. It corresponds to a payment above the standard wage to motivate employees as they work in a shift-regime basis. Undeniably, these workers are predisposed to more flexible schedules and very often work on less desirable shifts, comparing to other jobs. According to the actual Portuguese Labor Law, the shift allowance in the public sector is an increase around 20% on top of the standard wage.

**Preferably full weekend off**

Soft constraints (4.11) try to give workers a full weekend off instead of a single weekend day.

$$\sum_{d \in \{7-k, 7-k+7, \ldots\}} \sum_{s \in S} x_{itsd} \geq \theta^4_a, \quad \forall i \in I, \forall s \in S$$  \hspace{1cm} (4.10)

$$\sum_{d \in \{7-k, 7-k+7, \ldots\}} \sum_{s \in S} (X_{itsd} - X_{it, 6-k+7(w-1), s}) - Y^{W^O+}_{W^O} + Y^{W^O-}_{W^O} = 0,$$

$$\forall i \in I, w \in W$$  \hspace{1cm} (4.11)
For every person in every weekend, the difference between the first and second term in equation (4.11), respectively Sunday and Saturday decision variables, for all tasks and shifts, should be 0. Otherwise it is penalized in the objective function through $W^{\text{WO}+}$ and $W^{\text{WO}-}$, meaning working on a Sunday but not on a Saturday and vice versa, respectively.

Indeed, there are only two desirable hypotheses for these constraints to be satisfied. The first one is to work both days of the weekend and in this case both decision variables are equal to 1 and, therefore their difference equals 0. Or, on the contrary, it is preferable not to work in any day of the weekend and in that case the decision variables are both 0, thus not leading to any penalty in equation (4.11).

**Contractual working hours**

Within a planning period, every worker is expected to work a certain number of hours. Constraints (4.12) state that ideally, people work their specified number of contract hours, $\theta_i$. These hours are adjusted for the number of holidays ($\xi$) of the planning period, conveniently discounted ($\eta\xi$).

$$\sum_{t} \sum_{d} \sum_{s} L_t x_{ids} - Y_i^{H^+} + Y_i^{H^-} = \theta_i - \eta\xi, \quad \forall i \in I$$

(4.12)

Since tasks may have distinct lengths, it needs to be considered in these constraints. $L_t$ refers to the duration of each task $t$. The duration of a task directly influences the number of accumulated working hours of a person which should be approximated as much as possible to the contractual hours. For a certain week, the model allows a worker to work more or less hours but later compensates it, somehow, on a following week to complete the contractual hours. Also, a person may work more or less hours in a planning period and balance it on a next one. These compensations from one planning period to another may be treated in the model by adjusting the contractual hours of each worker individually. If the number of hours a person works differs from the contractual hours it is weighted as over or undertime, respectively, $Y_i^{H^+}$ and $Y_i^{H^-}$, in the third term of the objective function (4.1).

The Portuguese Labor Law states that every worker claims 22 vacation days per year. Unlike days off, vacation refers to a deliberate period away from the job (paid days). The number of consecutive days for the vacation days do not follow the rule stated in Equation (4.8). Vacation days are usually agreed between the administration team and workers. The present model does not consider directly the possibility of vacations. However, it can be taken into consideration by pondering it on the number of working hours of a person or by completely removing a person from a planning period.

**Performing tasks in the respective team**

Every person is allocated to one or more working groups. These comprise specific tasks with specific staff requirements, intended to facilitate the organization of the operations and avoiding unnecessary expenses.
Soft constraints (4.13) require that tasks belonging to a certain team should be assigned to members of that team as much as possible. Consequently, for each group, it is necessary to identify if people are assigned to tasks of their respective team(s) or not. In case they are not, the objective function is penalized through variables $Y^G_{O}$. Note that previously enforced by hard constraints (4.6) a person must be allocated to errands he/she is able to perform.

4.3. Solution approach

When dealing with complex problems, optimal methods are often intractable as they take long periods to find solutions and sometimes it is not even possible to find a feasible solution. Heuristic approaches are often chosen. These provide solutions in an acceptable running time for complex personnel scheduling problem such as the one that will be study at INEM.

Hence, after formulating the problem as an integer programming (IP) model, it is proposed a solution approach based on Column Generation (CG) which uses a diving heuristic procedure to find an integer feasible solution. An illustrative overview of the procedure and techniques applied is shown in Figure 4.2. The scheme shows that in the CG scheme, the problem is decomposed in a master and in a subproblem. At each iteration of the CG, a subproblem is solved to generate new columns to the master problem. Iterations last until the problem is solved to optimality. After optimality, the master is no longer repeated. In this Section, two distinct schemes to generate columns are presented. The main difference between them relies on the way the subproblem is solved in each iteration step.
Since the solution from CG might not be integer, branching is then applied (Figure 4.2). The diving heuristic approach dives on a branch-and-price tree to find good integer solutions. Once the LP relaxation through the master is solved, one or more variables are selected at each node for branching. After each branching step, the master LP is reoptimized and new columns are generated to the next level of the branch-and-bound tree. If the optimal solution of the new LP relaxation is again fractional, branching continues. There are many ways to perform the branching. In this Section, two branching methods are employed and compared. One method employs the branching on the largest fractional variable while the other employs it on all variables with a value above a certain threshold.

4.3.1. Alternative hybrid formulation

The integer model consisting of (4.1) - (4.13) can be formulated in a different way. For every person, a work pattern can be defined as the tasks assigned to that person over the planning horizon (Figure 4.3). This activity pattern, which corresponds to certain decision variables of the problem, is referred to as a column.

![Figure 4.3 - Example of an activity pattern in which tasks are assigned for every person of the problem over the entire planning period. It is emphasized in red a person's activity pattern.](image)

Although this approach can significantly speed up the optimization procedure, especially in complex problems with several variables, it requires a revision of the notation and readjustments on the original IP formulation.

4.3.1.1. Master problem

A new notation is defined:

- \( k_i \): set of columns for person \( i \)
- \( a_{iktsd} \): equals 1 if column \( k \) for person \( i \) assigns task \( t \) on shift \( s \) of day \( d \), 0 otherwise
- \( c_{ik} \): the cost of column \( k \) for person \( i \)

Additionally, binary decision variables can be defined as:

- \( z_{ik} \in \{0, 1\} \): equals 1 if column \( k \) is chosen for person \( i \), 0 otherwise

The objective function of the adapted master problem is formulated as follows (4.14):
minimize:
\[
\sum_{i \in I} \sum_{k \in K_i} (c_{ik} z_{ik}) + \sum_{d \in D} \sum_{s \in S} \sum_{j \in J} \sum_{t \in T} (w_j^{RE+} Y_{tds}^{RE+} + w_j^{RE-} Y_{tds}^{RE-})
\]  \hspace{1cm} \text{(4.14)}

The master is still a minimization problem. The main goal is to minimize two terms: the costs of the chosen activity patterns for every person in terms of \(z_{ik}\), and the under and overstaffing costs within each service \(j\).

The master problem has constraints (4.15) - (4.17).

\[
\sum_{i \in I} \sum_{k \in K_i} a_{iktds} z_{ik} - Y_{tds}^{RE+} + Y_{tds}^{RE-} = R_{tds}, \quad \forall t \in T, d \in D, s \in S
\]  \hspace{1cm} \text{(4.15)}

\[
\sum_{k \in K_i} z_{ik} = 1, \quad \forall i \in I
\]  \hspace{1cm} \text{(4.16)}

Constraints (4.15) are the coverage requirements. If coverage is not met, variables \(Y_{tds}^{RE+}\) and \(Y_{tds}^{RE-}\) are penalized in the second term of the objective function (4.17). Constraints (4.16) enforce that exactly one work pattern is chosen for each person. The feasible solution space of this new formulation is much shorter than the one of the IP formulation (4.1) - (4.13). Therefore, it is predictable a significantly decrease in the computational efforts when compared to the former formulation, which will be further analyzed.

Let \(\lambda_{tds}\) represent the dual costs associated with constraints (4.15) and \(\mu_i\), the dual costs associated with constraints (4.16). The Reduced Cost (RC) of a new column \(k\) for person \(i\) is then given by equation (4.17), where \(c_{ik}\) is the cost of the column based on the soft constraint violations.

\[
c_{ik} - \sum_{t \in T} \sum_{d \in D} \sum_{s \in S} a_{iktds} \lambda_{tds} - \mu_i
\]  \hspace{1cm} \text{(4.17)}

The duals \(\lambda_{tds}\) and \(\mu_i\) decrease the RC, whereas \(c_{ik}\) increases it. CG approach generates variables with negative RC that can potentially improve the objective function.

4.3.1.2. Subproblems

The LP relaxation of the adapted master problem must be feasible so that the information is used to solve the subproblem, named the pricing problem. The pricing problem checks if the LP solution is optimal and finds the column with the lowest RC for each person.

Hence, the decision variables for the pricing problem are \(a_{iktds} \in \{0,1\}\) : equal 1 if task \(t\) is assigned on shift \(s\) of day \(d\), otherwise it is 0.
The pricing problem is solved to generate new columns at each iteration that are added to the master (4.18).

minimize:

\[
\sum_{t \in T} \sum_{d \in D} \sum_{s \in S} (-a_{tds} \Lambda_{tds}) \\
+ \sum_{w \in W} (w^W y^W_w^+ + w^W y^W_w^-) + w^H y^H + w^H^2 y^H^2 \\
+ \sum_{j \in J} \sum_{g \in G_j} w^G_j y^G_g
\]

The set of constraints of the pricing problem are presented in expressions (4.19) - (4.29). These constraints are a reformulation of constraints (4.3) - (4.13), for column \( k \), using the hybrid formulation.

\[
\sum_{t \in T} (a_{td,night} + a_{td,morning} + a_{td,afternoon}) \leq 1, \quad \forall d \in D
\] (4.19)

\[
\sum_{t \in T} (a_{td,morning} + a_{td,afternoon} + a_{td+1,night}) \leq 1, \quad \forall d \in D \setminus \{|D|\}
\] (4.20)

\[
\sum_{t \in T} (a_{td,afternoon} + a_{td+1,night} + a_{td+1,morning}) \leq 1, \quad \forall d \in D \setminus \{|D|\}
\] (4.21)

\[
a_{tds} = 0, \quad \forall t \in T \setminus T^I, d \in D, s \in S
\] (4.22)

\[
\sum_{t \in T} \sum_{r \in (d,d+1, \ldots, d + \theta^1)} \sum_{s \in S} a_{trs} \leq \theta^1, \quad \forall d = \{1, \ldots, |D| - \theta^1 + 1\}
\] (4.23)

\[
\sum_{t \in T} \sum_{r \in (d,d+1, \ldots, d + \theta^2)} \sum_{s \in S} a_{trs} \geq 1, \quad \forall d = \{1, \ldots, |D| - \theta^2 + 1\}
\] (4.24)

\[
\sum_{t \in T} \sum_{d \in \{7-k, 7-k+1, \ldots\}} \sum_{s \in S} a_{drs} \leq |W| - \theta^3
\] (4.25)

\[
\sum_{t \in T} \sum_{d \in S \setminus \{7-k, 7-k+1, \ldots\}} \sum_{s \in S} a_{tds} \geq \theta^4_s, \quad \forall s \in S
\] (4.26)
Accordingly, staff must have a certain number of resting hours between consecutive shifts, which is enforced by constraints (4.19), (4.20), (4.21) for night, morning, and afternoon shifts, respectively. Staff cannot be assigned to tasks which they cannot perform (4.22). Staff are not allowed to work more than $\theta$ days consecutively (4.23) and they cannot have $\theta_2$ or more consecutive days off (4.24). In every planning period, each person must have at least $\theta$ Sundays off (4.25). Each person needs to work at least $\theta_t^s$ shifts of each type s (4.26). Preferably, people get the entire weekend off instead of a single day (4.27). Ideally, people work their specified number of contract hours ($\theta_W$), adjusted for the number of holidays ($\xi$) in the planning period, properly pondered ($\eta$) (4.28). Finally, tasks belonging to a certain group should be assigned to members of that group (4.29).

### 4.3.2 Strategies to solve the hybrid formulation

#### 4.3.2.1. CG initialization and loop

The master problem is initialized with a limited set of columns. The entries of these starting columns (activity patterns) must be defined before running the algorithm. In the present case, the master is initialized with 'supercolumns' with very high cost in which all entries are set to zero. This means that every entry of every column $k$ for every person $i$ is 0, irrespective of the task $t$, shift $s$ and day $d$, thus no one is assigned to any task within the planning period.

Another critical aspect is the number of columns added to the master in each LP optimization. The CG loop can be implemented in different ways (see e.g. Beliën and Demeulemeester, 2006). In this study, two possibilities are considered for the loop implementation – Figures 4.4 and 4.5.

A first option, schematized in Figure 4.4, is to solve the master problem, and obtain the dual variables for each constraint in the master problem. This information is then used to solve one subproblem for every person $i$ with objective function (4.18). After solving all the subproblems, columns that are found with a negative RC are added to the master problem, and then a new iteration begins. In the new iteration, the master generates a new set of dual variables and the subproblems are solved again.
Since this is a minimization problem it is repeated until no more columns with negative RC can be found (Hans, 2001). Once there are no negative RC variables, the master is no longer repeated, and one can say the problem is solved to optimality.

Figure 4.4 - Column scheme where one subproblem is solved for every person in each iteration (RC = Reduced Cost).

A second alternative, schematized in Figure 4.5, is to solve the master problem and then solve a subproblem for a single person $i$. If a column with negative RC is found, it is added to the master and the master is reoptimized. Then another subproblem is solved for person $i + 1$ and so on. The main difference comparing to the first alternative, is that in each iteration a subproblem is solved for a single person $i$ instead of solving a subproblem for every person $i$. Once all subproblems are solved for every person and no columns with negative RC are found, the process ends, and an optimal solution has been found.

Figure 4.5 - Column Generation scheme where a subproblem is solved for a single person after which the master is reoptimized (RC = Reduced Cost).
4.3.2.2. Stopping criteria of CG scheme

The stopping criteria for the CG phase directly influences the performance of the algorithm. A trade-off involves obtaining feasible convergence for the method and, at the same time, controlling the size of the branch-and-bound tree. The longer a search tree is, the more computational efforts are required. The tailing-off effect is extremely convenient to timely stop the CG scheme. The tailing-off effect is based on the idea that a large number of iterations are required to find an optimal solution. It results from the poor convergence of CG per iteration in the neighborhood of optimality. The root cause of the effect is the instability of the dual variables and the effects are more significant in large and degenerate problems (Hans, 2001).

![Figure 4.6 - Tailing-off effect suffered by CG for the problem in analysis.](image)

This effect is presented in Figure 4.6, where at first new columns improve the LP solution significantly, while in the later iterations each additional column only improves the objective function value very slightly. However, in an integer solution, only one column can be chosen for each person. Therefore, when the aim is to quickly find integer solutions, one can end the CG phase prematurely and start branching before LP optimality has been reached.

Hence, the challenge is to stop generating columns as soon as this effect becomes perceptible, without compromising the LP solution. There are three options considered to end the CG phase. For each one, once the stopping criteria are met the CG phase finishes.

A first way is to use the Lagrange lower bound, which gives a bound on the optimal LP value after each iteration. However, this option can only be used for the CG method in which the subproblem is solved for each person individually, after solving the master (Beliën and Demeulemeester, 2006). Accordingly, it could only be solved for the case presented in Figure 4.5 but not to the one in Figure 4.4.

A second way is to stop the optimization once the improvement in the objective value between two iterations drops below a certain threshold. However, it can happen that for a single iteration this improvement is temporarily zero, while in following iterations still significant improvements can be achieved. This instability and unpredictability can lead to unbiased results, very difficultly detected.
In the third way, the CG scheme is stopped after a predetermined number of iterations. For this option two positive integer parameters are set: $\beta_{\text{root}}$ for the root node and $\beta_{\text{diving}}$ for each node during diving. This is to say, $\beta_{\text{root}}$ states the number of iterations allowed in the root node, whereas $\beta_{\text{diving}}$ the number of iterations allowed in all other nodes (after the first branching decision). The higher the value of these parameters, the more accurate the solution is expected to be but the more it takes until finishing the CG phase. In the limit, if the number of iterations are set to infinite, the CG scheme is solved to optimality. This option is the one used in the computational results, in Chapter 5.

### 4.3.2.3. Diving heuristics

The solution found through CG might not be integral. Therefore, branching is needed to find an integer solution. However, as an exact branch-and-price approach requires a prohibitively large amount of computation time for large problem instances, a diving heuristic is used to quickly find good integer solutions. In a diving heuristic, the branch-and-price tree is traversed in a depth-first manner until finding a feasible solution. After solving the LP relaxation, one or more variables are selected for branching. Unlike an exact branch-and-price approach, the tree originated in diving heuristic is imbalanced. Indeed, in each branching level, only one selected variable is set to one and consequently the number of nodes that must be examined is reduced significantly. This can be done in multiple ways. Figures 4.7 and 4.8 show the two ways considered in this thesis.

One way, described in Figure 4.7, is to fix the largest fractional variable to one. In this case, the solutions from CG phase are split in integer and fractional. In case a solution for person $i$ is already integer the column for that person $i$ is fixed and that person is removed from the problem. Otherwise, if the solution is not integer yet, the problem reads all the fractional columns and sets to one the fractional column with the highest solution i.e. the column with the highest variable. If different columns have a same highest column solution, then the algorithm picks only one, specifically the first one to appear. Next, the person whose column has been fixed is eliminated from the next iteration step. On the contrary, for the ones whose partial schedule has not yet been found, a new CG phase is undertaken. The iterations proceed until one solution has been found for every person $i$ considered in the problem. The number of iterations corresponds to the number of tree levels to search in the branch-and-bound tree in this diving heuristic phase.

![Figure 4.7 - Overview of branching on the largest fractional variable that is set to 1.](image-url)
Another way, displayed below in Figure 4.8, is to set to one all fractional variables with a value above a certain threshold ($\delta$). In case a person has at least one column variable above that threshold, the remaining columns for that person ($z_{ik'}$) are set to zero and the variable has been found for person $i$. Next, CG is used to iterate new solutions for the people whom solution has not been fixed yet. While in the method illustrated in the first method (Figure 4.7) only one person is removed per iteration, in this second method (Figure 4.8), more than one person can be removed per cycle. Indeed, the number of fixed columns match the number of solutions found above the threshold. Consequently, the number of iterations and the correspondent levels of the branching tree are expected to be lower and the process is expected to be faster when compared to the former method, schematized in Figure 4.7.

![Diving heuristic: method 2](image)

**Figure 4.8 - Overview of branching on variables with a value above $\delta$.**

In diving heuristic and similar to a typical branch-and-price approach, after each branching decision, new columns are generated as it is not guaranteed that the columns of the root node can be combined into good integer solutions (Gomes et al., 2017; Beliën and Demeulemeester, 2006). Subproblems only need to be solved for those people $i$ for which no column variable has been fixed yet. Also, all columns for people for which a column has been fixed are explicitly removed from the master problem (Gamache et al., 1999). The problem is completed when one column variable has been fixed for every person $i$ considered in the problem.

To better understand these two strategies employed, an example for branching on the column variables is presented.

Consider a sample composed by three people ($1, 2, 3$) each with three columns. The arbitrary values for their variables are as follows:

$$
\begin{bmatrix}
    z_{11} & z_{12} & z_{13} \\
    z_{21} & z_{22} & z_{23} \\
    z_{31} & z_{32} & z_{33}
\end{bmatrix} = 
\begin{bmatrix}
    0.70 & 0.10 & 0.20 \\
    0.65 & 0.10 & 0.25 \\
    0.50 & 0.30 & 0.20
\end{bmatrix}
$$
If the first strategy is chosen, the largest fractional solution is set to 1. In this sample, $z_{11}$ is 1 and the solution for person 1 has been found. For the next iterations, the CG generates columns only for people 2 and 3. The diving process is repeated until an integer solution is obtained for every person.

Setting a threshold $\delta = 0.6$, the second strategy implies that $z_{ik}$ is set to 1 for every case $z_{ik} > \delta$. Accordingly, $z_{11}$ and $z_{21}$ are above the threshold and therefore are set to 1. All the other columns for persons 1 and 2 are eliminated because their feasible solution has already been found. In the following re-optimization, only person 3 is considered.

Branching on the column variables in a diving heuristic is well-suited as it reduces the remaining solution space much more and it does not require significant changes to the pricing problems because the column variables are always fixed to one and never to zero (Joncour et al., 2010).

### 4.4. Chapter considerations

At first, the main characteristics of the model have been presented. It considers staffing of 24/7 services. Schedules can be used for any time horizon, depending on the context it is applied. Each day of the planning horizon is divided into three shifts: a night shift from 0:00 a.m. to 8:00 a.m., a morning shift from 8:00 a.m. to 4:00 p.m., and an afternoon shift from 4:00 p.m. to 0:00 a.m. The workforce is organized in working teams. Each team has a set of tasks to be performed and a set of people to perform those tasks.

On top of the problem statement, an IP model was formulated. There is an attempt to generalize the model so that it may be easily adapted to several contexts. IP models tend to be unmanageable for many real-life problems as they often fail in finding a reasonably good solution within an acceptable time frame (specially in large dimensions problems). Henceforth, it was considered an alternative exact-heuristic formulation, including a CG formulation that decomposes the problem on its members. The purpose is to find the activity pattern, as a column setting, for every person, in compliance with every restriction of the model. A diving heuristic is used to find integer solutions. Two CG schemes as well as two branching methods have been implemented. Additionally, as the CG phase requires a prohibitively large amount of time to be solved to optimality, branching is started prematurely after a fixed number of CG iterations.

In the next Chapter, the model is applied to a real-life dataset at INEM. The relative performance of the different algorithm configurations are compared in computational aspects and solutions' quality.
5. INEM Case-study

This Chapter presents the results from the application of the model to the real-life case-study at INEM. Section 5.1. introduces background on the INEM problem. The context of the problem at INEM is explored, detailing the workforce, the tasks and the working teams. Section 5.2. introduces the parameters and dimensions of the instances under study. The input tables for the INEM instance are explained as well as the weights set for the tests. Section 5.3. presents the computational results, which allows to compare the performances on both approaches, exact and heuristic. The consistency of the heuristic is analyzed by testing it on different configurations and on four test instances generated by varying parameters on the original INEM instance (e.g. number of days, number of people). Tests are compared with respect to solution quality and computation time. The solution for the best configuration found is provided and compared to a current planned schedule at INEM. Section 5.4. shows the graphical user interface implemented, which can easily be used by everyone. Finally, Section 5.5. sums up the content of the Chapter, emphasizing some reflections for future extensions of this thesis work.

5.1. INEM setting

In this thesis, a model is formulated to represent the characteristics of the problem at INEM. This emergency institution operates on a continuous basis, 24 hours every day. The case is applied to the scheduling of the technical personnel from two distinct services, the dispatch center (CODU) and the emergency vehicles (EVs). The geographic distribution of both services is shown on Figure 5.1. The scope of this study is the Lisbon region and some neighboring areas where INEM vehicles are located and are coordinated by the Lisbon center: Almada, Amadora, Cascais, Elvas, Estremoz, Ponte de Sor, Sacavém, Seixal, Setúbal, Tomar and Torres Novas.

Figure 5.1 - INEM services geographic distribution. Red marker indicates the CODU service location. Green markers indicate EVs services location, depicted by the regions where INEM has vehicles and these are under the Lisbon Center coordination (Google maps, 2017).
5.1.1. The workforce – TEPHs

As previously mentioned in Problem Contextualization, the workforce considered in this thesis is composed by Técnico de Emergência Pré-Hospitalar (TEPHs) corresponding to workers from CODU and EVs services. INEM is responsible for scheduling the set of TEPHs which is, in fact, the larger-size category of emergency staff at INEM. This emergency institute also builds the schedules of some physicians, nurses and psychologists, but due to the significant smaller number of these professionals, their schedules are easier to construct, thus leading to less conflicts. Besides being less in number, while every TEPH has a fixed number of working hours, there are exceptions with physicians, nurses and psychologists who also perform activities in health care facilities. In these cases, professionals work both in the respective health unit and in the emergency services. Accordingly, their shifts need to be shared, and therefore their working hours at INEM tasks are substantially lower when compared to a regular contract held by a TEPH. Moreover, some physicians, nurses and psychologists are not legally connected to INEM but to a healthcare organization, such as hospitals or other medical units. Under these circumstances, their official administrations are the ones responsible for building the rosters, which are not under the INEM control.

Every TEPH at INEM holds a similar contract. It entails a full-time regime contract of 35 hours workweek. It is calculated by assuming 7 hours of work per day, and 5 working days during a week, from Monday to Friday. Overtime is a source of extra cost for INEM and it is allowed but not recommended. Indeed, working beyond the contractual hours is, most of the times, driven by the need to meet the operational demand due to e.g. unpredictable absences, sickness.

5.1.2. Tasks and working teams

Each service at INEM has specific tasks requirements that need to be satisfied. The tasks are summarized in Table 5.1:

<table>
<thead>
<tr>
<th>Tasks</th>
<th>CODU</th>
<th>EVs</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CODU shift responsible</td>
<td>AEM driver</td>
</tr>
<tr>
<td></td>
<td>CODU task</td>
<td>AEM team responsible</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SIV task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TIP task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>UMIPE task</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MEM task</td>
</tr>
</tbody>
</table>

At CODU, TEPHs are assigned to two tasks: the CODU shift responsible and the CODU task. The shift-responsible person supervises and coordinates the activities of a shift at CODU, by controlling the operations and taking important decisions whenever necessary. The second task is related to responsibilities over receiving the calls, routing and dispatching the vehicles, collecting emergencies data and call-back services.

Within EVs, AEM type is the only that requires two TEPHs. These two TEPHs play distinct roles, one acts as a driver whereas the other is the team-responsible. Their tasks are identified by AEM driver and
AEM team responsible, respectively. The remaining EVs tasks are recognized according to the category of the medical vehicle in which they are performed: SIV task, TIP task, UMIPE task and MEM task.

Each TEPH belongs to one of the two main services. Services at CODU and EVs are not mutually exclusive. Despite every TEPH is allocated into a core service they may perform tasks at the other service. However, unless it corresponds to direct desired changes between TEPHs, this should be avoided if possible. Within the assigned service, TEPHs are integrated in working teams with a set of tasks. The distribution of workers over the existent teams and tasks is crucial. The assignment of TEPHs to working teams and respective tasks depends on two main factors: the qualified skills of each person and the location of the tasks.

Firstly, a TEPH only performs a task if he/she has the required skills for it. For instance, every TEPH in the CODU can do the CODU task but not everyone can be CODU shift responsible as it requires a more specialized training that only some TEPHs have. In the EVs case, a specific prerequisite is needed to perform MEM task (license for motorcycles). Thus, it is convenient to assign people to teams in which they can perform most of that team’s tasks.

Secondly, both CODU tasks are confined to a room at INEM however more challenges arise to distribute TEPHs of EVs service over the existent groups and jobs as it is necessary to take spatial considerations into account. Even if TEPHs own the skills for some tasks, the distance between the location of those tasks must be weighted before assigning a task. For each TEPH is given a main EVs location. In case a TEPHs is assigned to a task located more than 50 km away from the own core location, INEM must, by law, provide a transportation budget. The distances between every two possible EVs locations considered in the case-study at INEM, in kilometers (km), are presented in Table 5.2.

Table 5.2 - Distance (km) between every pair of EV locations for the problem in analysis. The distances above and below 50 km are colored in red and green, respectively.

<table>
<thead>
<tr>
<th></th>
<th>Almada</th>
<th>Amadora</th>
<th>Cascais</th>
<th>Elvas</th>
<th>Estremoz</th>
<th>Lisbon</th>
<th>Ponte de Sor</th>
<th>Sacavém</th>
<th>Seixal</th>
<th>Setúbal</th>
<th>Tomar</th>
<th>Torres Novas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Almada</td>
<td>17</td>
<td>36</td>
<td>200</td>
<td>163</td>
<td>13</td>
<td>155</td>
<td>26</td>
<td>16</td>
<td>40</td>
<td>149</td>
<td>123</td>
<td></td>
</tr>
<tr>
<td>Amadora</td>
<td>29</td>
<td>40</td>
<td>216</td>
<td>171</td>
<td>13</td>
<td>148</td>
<td>18</td>
<td>29</td>
<td>54</td>
<td>143</td>
<td>117</td>
<td></td>
</tr>
<tr>
<td>Cascais</td>
<td>234</td>
<td>96</td>
<td>194</td>
<td>34</td>
<td>174</td>
<td>46</td>
<td>72</td>
<td>165</td>
<td>171</td>
<td>163</td>
<td>142</td>
<td></td>
</tr>
<tr>
<td>Elvas</td>
<td>40</td>
<td>100</td>
<td>208</td>
<td>216</td>
<td>194</td>
<td>26</td>
<td>16</td>
<td>216</td>
<td>171</td>
<td>163</td>
<td>142</td>
<td></td>
</tr>
<tr>
<td>Estremoz</td>
<td>171</td>
<td>70</td>
<td>169</td>
<td>157</td>
<td>137</td>
<td>143</td>
<td>134</td>
<td>140</td>
<td>116</td>
<td>140</td>
<td>134</td>
<td></td>
</tr>
<tr>
<td>Lisbon</td>
<td>144</td>
<td>138</td>
<td>143</td>
<td>143</td>
<td>83</td>
<td>83</td>
<td>83</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>Ponte de Sor</td>
<td>138</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>83</td>
<td>83</td>
<td>83</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td>143</td>
<td></td>
</tr>
<tr>
<td>Sacavém</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td>34</td>
<td></td>
</tr>
<tr>
<td>Seixal</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td>35</td>
<td></td>
</tr>
<tr>
<td>Tomar</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td></td>
</tr>
<tr>
<td>Torres Novas</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td>9</td>
<td></td>
</tr>
</tbody>
</table>

It is not desirable to assign people to tasks with distance locations in red. For instance, it is inconvenient to assign a TEPH to a task in Estremoz if the core location is in Lisbon as the distance between Lisbon and Estremoz is 171 km. In this case not only INEM incurs in transportation additional expenses but also it would be very inconvenient to the staff. On the contrary, the distance between Lisbon and Sacavém is only 12 km, so TEPHs performing tasks in these two places do not require any transportation.
budget. Still, upon the information displayed on Table 5.2, a correct and efficient definition of working teams represent potential saving costs for INEM, and can directly impact on the personnel satisfaction.

5.1.3. Shift sequences

A sequence of shifts is regular when it is possible to assign a repetitive roster for a worker or for a team. In this situation, the constant pattern of shifts recurring over time is known as a stint (Ernst et al., 2004b) and is often the option for the CODU service. Unlike this, it is not possible to assign a regular roster to the TEPHs working in EVs.

5.1.3.1. Stint basis at CODU

CODU is a stint based service, and for this reason only some specific shift sequences are recommended to the teams within this service. Rather than being assigned to a set of tasks or vehicles, in CODU there is an overall pattern-shift intended to be sequentially followed by the different teams. A determinant factor contributing to the use of a stint basis is the fact that every task in CODU is confined to the same physical space and has constant workforce demands. In Figure 5.2 is shown the current sequence of shifts followed by the 5 CODU teams, where a standard sequence is shifted for the different teams.

![Stint sequence for the working groups at the CODU service](image)

Figure 5.2 - Stint sequence for the working groups at the CODU service (M – Morning shift, A – Afternoon shift, N – Night shift, O – Day-off).

It is shown that team 1 starts with a morning shift, followed by another morning shift on the next day, and so forth. The roster line, as a sequence of stints, for group 1 is: MMAAONNOOO (O represents a day-off). Group 2 starts with an afternoon shift followed by another afternoon shift on the next day, a day off in the day after, etc. The sequence for team 2 is thus: AONNOOMM. The schedules at INEM are usually built for an entire month. Given that a month has more than 10 days (the number of days of a standard stint), the pattern is sequentially repeated more than once during a month. By using a stint basis scheme, there are benefits presumed, as demand is expected to be more easily met and timetables tend to be more unbiased and organized.

5.1.3.2. EVs operation mode

There is no pre-defined or sequential shift pattern for the EVs teams. Each task is performed at a specific site, with distinct demands. These characteristics disallow the attribution of a regular pattern to the overall service as it happens in CODU. In fact, the team within this service have a different set of tasks,
which rely on the set of vehicles assigned to it. It is possible to depict each emergency vehicle at INEM by its operating mode. It is based on the patients’ needs during the day, obtained through historical data and statistical procedures. Henceforth each vehicle has a specific arrangement of shift-daily patterns, resulting from the three shifts-combinations:

- 3 shifts per day – night, morning and afternoon (N+M+A).
- 2 shifts per day – night and afternoon (N+A), night and morning (N+M), morning and afternoon (M+A).
- 1 shift per day – night (N), morning (M) or afternoon (A).

Due to e.g. seasonality, EVs’ operating modes may be adapted or adjusted, such as the case with the introduction of one special MEM in Portimão during the summer period (although outside the region of Lisbon, which is the scope of this analysis).

5.2. Input analysis

Both the IP and the Column Generation based models proposed previously require specific parameters and inputs. All the data used for the experiments carried in this analysis have been provided by INEM. Besides, the model has been closely discussed with the schedule makers, so that the real-life context at INEM is accurately expressed.

In the context of the case study at INEM, working time regulations state that people cannot work more than 6 days consecutively \( \theta^1 = 6 \), cannot have 5 or more consecutive days \( \theta^2 = 5 \), and must have at least 1 Sunday \( \theta^3 = 1 \). Furthermore, out of equity considerations with regards to shift distributions, it is required that each person works at least 2 shifts of each type, i.e. \( \theta^4_{s} = 2, \forall s \in S \). Finally, a standard contract specifies a working time of 140 hours per month and the number of holidays in the instances tested is zero, implying that \( \theta_i = 140 \) and \( \xi = 0, \forall i \in I \). This means that everyone has zero accumulated hours from previous schedules and no one is on vacation.

Moreover, the overall problem has 289 TEHPs who are divided into 22 teams (5 in CODU and 17 in the EVs), and need to be assigned to 61 different tasks (10 in CODU and 51 in the EVs). The planning horizon considered is four weeks (28 days). Over the entire planning horizon, a total of 4527 task-demands need to be filled in. The starting day is set to 0, meaning the scheduling starts on a Monday. This parameter is not a determining factor for the quality of the solutions or for the computational times, so it is not modified throughout the tests. However, it is relevant to incorporate it in the model as it enhances the authenticity of the model, making it simpler to become applicable in the future to the INEM or to another context. In Table 5.3 is summarized relevant information for this case-study. For each working team, it is stated the number of workers assigned as well as the tasks number and the correspondent type of task. Tasks in CODU range from team one to team number five (i.e. task number 1 until 10), and in EVs, start from the sixth group (i.e. from task number 11).
Every team in CODU performs the same two tasks: CODU shift responsible and CODU task. As Table 5.3 details, the number of people and vehicles per team in the EVs service is quite irregular. The distribution of tasks within a team is dependent on the respective set of vehicles. Each vehicle is assigned to a single working team. For the sake of simplicity, each task is identified by the respective category (AEM, SIV, TIP, UMIPE or MEM) and by an ID that can be either a number, e.g. AEMs in Lisbon or a city, e.g. corresponding to the city where SIVs operate. Note that every category of vehicle comprises a single task, except the AEM category that comprises two tasks, namely AEM 1 comprise tasks number 11 and 12, AEM 1 driver and AEM 1 team responsible, respectively.

5.2.1. General information

The instances of the INEM staff scheduling problem are introduced in matrices in a .txt file format that is called before running the model. Five different tables are built. People are assigned to teams and tasks in table person x team and person x task, respectively. Each team comprises a specific set of tasks (team x task), and each task has specific demands (task x demands) and lengths (length x task). These tables are processed when running the algorithm.
**Person x Team**

This table is used to allocate each person in one or more working teams, in which they are expected to perform tasks. This assignment is based not only on a person's skills, mostly allocated to groups in which they can perform the tasks, but also on the tasks' locations.

Figure 5.3 shows a fragment of this table with 9 people. If it is given 1 to \((p,t)\) entry, it means that person \(p\) belongs to group \(t\). Otherwise, 0 is assigned and that person does not belong to that team.

![Person x Team Matrix](image)

The workforce of TEPHs in CODU belong exclusively to a single working team whereas in EVs they may fit in more than a team. TEPHs number 67 and 68 are assigned to the fifth group (the last one from CODU). The set of people assigned to EVs starts from person number 69. Some of these TEPHs are assigned to more than a working team, e.g. person 69 belongs to team 6 and to team 18. From Table 5.3, group 6 has the following set of tasks: AEM1, AEM10, AEM 15 and SIV Lisbon, while group 18 comprises only the TIP Lisbon task while there are others assigned to a single team, e.g. person number 75 only belongs to group 6.

**Team x Task**

This table is used to assign each working team to one or more tasks. Figure 5.4 shows a fragment of this table. The information corroborates the information previously displayed in Table 5.3. For instance, team number 5 (CODU) is assigned to task number 9 and 10, whereas team number 6 (EVs) is assigned to tasks ranging from 11 to 17, and so forth.

![Team x Task Matrix](image)

**Task x Demands**

This table states, for every single task, the personnel demand in every day and shift of the planning. The demands differ among the services and tasks.
The TEPHs’ daily demands per shift in CODU are established a priori:

- Night shift – requires 1 TEPH for CODU shift responsible and 7 TEPHs for CODU task (8 TEPHs for CODU task on Fridays and Saturdays).
- Morning shift – requires 1 TEPH for CODU shift responsible and 19 TEPHs for CODU task.
- Afternoon shift – requires 1 TEPH for CODU shift responsible and 17 TEPHs for CODU task.

As already mentioned in Section 5.1.3.1. it is attributed a sequence of shifts to the CODU teams and, consequently to the set of TEPHs that are integrated in those teams. The idea is to assign the predefined demand to one or more teams that are expected to work in a specific day and shift.

On the other hand, in the EVs service, the TEPHs’ demand to operate a single shift, regardless of the shift and day of the planning, are:

- AEM vehicle – 2 TEPHs, i.e. 1 TEPH for AEM driver and 1 TEPH for AEM team responsible.
- SIV vehicle – 1 TEPH for SIV task.
- TIP vehicle – 1 TEPH for TIP task.
- UMPE vehicle – 1 TEPH for UMPE task.
- MEM vehicle – 1 TEPH for MEM task.

As stated in Section 5.1.3.2. Subs, unlike CODU, for the EVs service it is not possible to assign an overall shift-pattern to its teams. For the circumstances of the problem described at INEM, distinct operation modes are predicted for the vehicles upon which the specified staff requirements arise. The majority of AEMs operate all shifts in a continuous 24/7 (N+M+A), some run during morning and afternoon only (M+A), or only morning (M), or only afternoon (A) and there is one functioning only afternoons during the week and with full capacity on the weekends. Every SIV, TIP and UMPE operate in an interruptedly basis of three shifts every day (N+M+A). Finally, MEM vehicle operates only the morning shift (M).

As stated in Section 5.1.3.2. Subs, unlike CODU, for the EVs service it is not possible to assign an overall shift-pattern to its teams. For the circumstances of the problem described at INEM, distinct operation modes are predicted for the vehicles upon which the specified staff requirements arise. The majority of AEMs operate all shifts in a continuous 24/7 (N+M+A), some run during morning and afternoon only (M+A), or only morning (M), or only afternoon (A) and there is one functioning only afternoons during the week and with full capacity on the weekends. Every SIV, TIP and UMPE operate in an interruptedly basis of three shifts every day (N+M+A). Finally, MEM vehicle operates only the morning shift (M).

As stated in Section 5.1.3.2. Subs, unlike CODU, for the EVs service it is not possible to assign an overall shift-pattern to its teams. For the circumstances of the problem described at INEM, distinct operation modes are predicted for the vehicles upon which the specified staff requirements arise. The majority of AEMs operate all shifts in a continuous 24/7 (N+M+A), some run during morning and afternoon only (M+A), or only morning (M), or only afternoon (A) and there is one functioning only afternoons during the week and with full capacity on the weekends. Every SIV, TIP and UMPE operate in an interruptedly basis of three shifts every day (N+M+A). Finally, MEM vehicle operates only the morning shift (M).

A small piece of the task x demands matrix is displayed in Figure 5.5. For task number 9 (CODU shift responsible) and 10 (CODU task), the sequence of stints to this team (team number 5) is: OOMMAAONNO. Accordingly, task number 9 and 10 are assigned to a day-off on the first and second
days of the planning and, therefore, they have no demand on those days. Then, on the third and fourth day, these teams are assigned to two morning shifts and demands (for tasks 9 and 10) are 1 and 19, respectively. The sample also shows that AEM 1 operates 24/7, since tasks 11 and 12, respectively AEM 1 driver and AEM 1 team responsible require one TEPH to operate every day and shift of the planning horizon. These demands should be satisfied as much as possible.

**Person x Task**

This table defines the skills of each person to each task. Accordingly, if a person can do a given task, the table displays 1, otherwise 0. By internal regulation, every TEPH allocated in CODU must possess competencies to perform CODU task. Only some can do CODU shift responsible as it requires a higher job experience. Besides this, people from EVs that sporadically work in CODU, have skills to do CODU task but not CODU shift responsible. A TEPH that has EVs as the core service must have the proper skills to do AEM driver and AEM team responsible. Some of the remaining tasks cannot be performed by everyone since they require extra hours of training such as the case of SIV, TIP and UMIPE task, or supplementary abilities as the license for driving motorcycles in the MEM task case. Naturally, each person is expected to have most of the skills to perform the tasks belonging to that person’s group.

![Person x Task Matrix](image)

*Figure 5.6 - Sample of person x task matrix for the real-life dataset at INEM.*

The available piece of data, in Figure 5.6, corroborates the aforementioned information: e.g. person 67 and 68 can do CODU task but not CODU shift responsible; person 69 does all AEM and SIV task related, and also TIP task, which is in agreement with the nature of the tasks included in the teams where this person is integrated; person 75 does every AEM and SIV task related and additionally CODU task too.

**Length x Task**

This table informs, in hours, about the length of each task:

![Length x Task Matrix](image)

*Figure 5.7 - Full task x length matrix, for the real-life dataset at INEM. The time unit of the tasks length is the hour.*

Figure 5.7 shows that for the circumstances at INEM, shifts can only last 8 or 12 hours. There is a 30 minutes break after 5 working hours, as long as the breaks do not compromise the care delivered
performance. The majority of the tasks takes 8 hours, except the MEM task that has a length of 12 hours instead (task number 59, 60 and 61). Indeed, for the MEM category, there is only one shift per day, a special one of 12 hours (M).

The proposed model allows MEM task to have a shift duration that differs from the regular shift length. The MEM task starts at the same time of the shift that is assigned to, but finishes after the end of it. Specifically, MEMs have a longer morning shift of 12 hours, from 8:00 a.m. to 8:00 p.m. Motorcycles do not operate neither on the night nor on the afternoon shift, which prevents the overlapping of working shifts.

5.2.1.1. More data sets

In this Section, a set of test set are generated based on the INEM data set so that the algorithm proposed is validated. The most important issues influencing the complexity of the problem at the INEM context are: the number of TEPHs, the distribution of their skills, i.e. the number of people that can perform each task (this impacts the symmetry of the problem), and the length of the planning horizon. The number of tasks is implicitly determined by the number of TEPHs and narrowly linked to the distribution of their skills. To create these sets, proportional metrics are applied to ensure their viability. It yields the following data sets:

1. Test more days (TestMD): the planning horizon is extended from 28 to 56 days. The same task demand is used for both months. Task x demands is the only matrix changed. All the other parameters are kept constant: number of people, teams and tasks.

2. Test more people (TestMP): the number of workers is increased from the original 289 to 417. This number is chosen so that each team size is increased by the same factor. The daily demand is also higher and at the same rate as the number of workers. The skills of each TEPH for each task are proportionally modified as well. Thus, the revised matrices are: person x team, task x demands and person x task. Only team x task and task x length are kept. All the other parameters, namely number of teams and tasks, are not modified.

3. Test less symmetry (TestLS): the number of tasks is increased from 61 to 103. This number is chosen so that the number of tasks of each team increases by the same factor. It is assumed that people that have skills to do the old task do not have skills to perform every new task, so that symmetry of the problem is reduced. Task demand is divided proportionally over the new tasks, so that the old demand is decreased, while demand for the new tasks are introduced. The number of TEPHs and teams remain constant, but every single matrix needs to be modified. The new task lengths are defined so a same proportion is guaranteed.

4. Test homogeneous skills (TestHS): created by assuming every TEPH has skills to do every task, so that symmetry is maximized. Only person x task matrix is changed: every entry is set to 1. All the other parameters and matrices are maintained.
The problem dimensions for the instances tested are summarized in Table 5.4, where the INEM instance considers the overall problem. From the INEM dataset two additional real-life datasets can be derived by considering only CODU or only the EVs, i.e. CODU and EVs instance, respectively. The other generated test sets are also displayed. For each instance, Table 5.4 shows the number of people, teams, tasks, days and also the number of decision variables. Recap from Model Formulation that the decision variable of the problem is $x_{itds}$. Thus, the number of $x$ variables for each instance throughout the respective number of people, tasks, days and shifts.

Table 5.4 - Summary of the instances used to perform the computational experiments. The INEM instance results from combining CODU and EVs. TestMD, Test MP, TestLS and Test HS are variants of the INEM instance, used to validate the algorithm.

<table>
<thead>
<tr>
<th>Instance</th>
<th># People</th>
<th># Teams</th>
<th># Tasks</th>
<th># Days</th>
<th># Variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODU</td>
<td>68</td>
<td>5</td>
<td>10</td>
<td>28</td>
<td>57,120</td>
</tr>
<tr>
<td>EVs</td>
<td>221</td>
<td>17</td>
<td>51</td>
<td>28</td>
<td>946,764</td>
</tr>
<tr>
<td>INEM</td>
<td>289</td>
<td>22</td>
<td>61</td>
<td>28</td>
<td>1,480,836</td>
</tr>
<tr>
<td>TestMD</td>
<td>289</td>
<td>22</td>
<td>61</td>
<td>56</td>
<td>2,961,672</td>
</tr>
<tr>
<td>TestMP</td>
<td>417</td>
<td>22</td>
<td>61</td>
<td>28</td>
<td>2,136,708</td>
</tr>
<tr>
<td>TestLS</td>
<td>289</td>
<td>22</td>
<td>103</td>
<td>28</td>
<td>2,500,428</td>
</tr>
<tr>
<td>Test HS</td>
<td>289</td>
<td>22</td>
<td>61</td>
<td>28</td>
<td>1,480,836</td>
</tr>
</tbody>
</table>

5.2.2 Weights

The objective function seeks to minimize the weighted sum of the penalties associated to the soft constraints. However, the criteria do not have all the same importance and therefore should be weighted accordingly. The relative and then quantitative importance of the criteria have been discussed and adjusted with the decision makers at INEM. The goal is to accomplish accurate and feasible weights to properly test the performance of the algorithms on the different problem dimensions. The weights used for the tests, which are maintained for every instance tested, are summarized in Table 5.5:

Table 5.5 - Objective function weights used in the computational tests. Each parameter in the objective function is weighted based on its relative importance.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>$w^{CODU}$</td>
<td>10</td>
</tr>
<tr>
<td>$w^{EV}$</td>
<td>10</td>
</tr>
<tr>
<td>$w^{CODO}$</td>
<td>100</td>
</tr>
<tr>
<td>$w^{C}$</td>
<td>1000</td>
</tr>
<tr>
<td>$w^{DD}$</td>
<td>10</td>
</tr>
<tr>
<td>$w^{D}$</td>
<td>1</td>
</tr>
<tr>
<td>$w^{C}$</td>
<td>10</td>
</tr>
<tr>
<td>$w^{EV}$</td>
<td>20</td>
</tr>
</tbody>
</table>
Generally, overstaffing is considered less undesirable than understaffing. Additionally, understaffing in EVs is much worse than understaffing in CODU, as CODU continues to operate without significant problems if there is a small shortage in the personnel assigned on a certain shift, while the understaffing of EVs may lead to the non-operationality of the vehicles, thus \( w_{EV}^{RE} > w_{CODU}^{RE} \).

Not having a full weekend off is worse than working under and overtime. Moreover, a TEPH working under or overtime is considered equivalent and therefore is equally weighted \( (w^{H+} = w^{H-}) \).

Finally, because EVs are located at different sites, assigning people to tasks outside their team is less wanted for the EVs than for CODU, i.e. \( w_{EV}^{G} > w_{CODU}^{G} \). In fact, even if people have the skills to perform a task of another team, the distance between tasks' locations is a crucial issue to forbid the assignment.

### 5.3. Computational results

Both the standard IP model and the diving heuristic approach outlined in Chapter 4 are used to solve the current problem at INEM. The algorithm is coded in C++14 and compiled with Microsoft Visual Studio 2015. C++ is used in combination with CPLEX. The callable library of ILOG CPLEX 12.6.2 is used as IP solver. All tests are executed on a PC with an Intel Core i5-5200U CPU of 2.20 GHz and 8 GB of RAM under the Windows 10 operating system.

The objective function weights used in the tests are the ones previously displayed in Table 5.5.

#### 5.3.1. CPLEX

The CPLEX software employs a branch-and-cut approach to solve the problem, which is initialized by decomposing the model into a single master and a series of LP subproblems. The first node to be explored is the root node for which the correspondent subproblem is the LP relaxation. At each node of the search tree there is a subproblem. The nodes are explored by solving the related subproblem. After each iteration, the value of the objective function is improved. This optimization software finds a set of integer values for the decision variables of the problem, which corresponds to feasible and optimal solutions of the model. To obtain a feasible solution, all hard constraints in the model need to be satisfied, while the optimal solution also minimizes the objective function value. Ideally, CPLEX finds an optimal solution but due to problems’ complexity and large dimensions it is not always achievable. In the present Section, computational experiments are performed on three instances (CODU, EVs and INEM; the characteristics of these instances are available in Table 5.4). The objective is to understand how CPLEX outperforms on the staff scheduling problem at INEM. The linear optimization problem has been first solved for CODU dataset as this is the simplest one. The problem is based on the standard IP model, described by expressions (4.1) - (4.13) and the results obtained are exhibited in Figure 5.8:
From Figure 5.8 some observations can be made. For this small instance, CPLEX is able to find an optimal solution very fast. The total computation time for it is just 3.472 seconds. The best integer solution has an objective value of 4196, while the lower bound is already, very close, at 3941. The LP solution is used as lower bound and it corresponds to the objective value of the LP relaxation of the root node. The optimality gap is 0%, while the gap between the optimal solution and the LP relaxation of equations (4.1) - (4.13) is only 6%. In the optimal solution all the 1277 task-demands are met (i.e. the unmet demand for CODU is zero).

Following, Table 5.6 summarizes the computational results for CODU, EVs and INEM instances with respect to the Best-Found Solution (BFS) and total time, LP relaxation value and correspondent time, and the percental gap between the BFS and the LP relaxation solved by CPLEX solver. The CODU dataset is the only one for which CPLEX succeeds in finding an optimal solution. For EVs and INEM, the software has been stopped after 5 hours of search (i.e. 18 000 seconds).
For EVs instance, after 5 hours CPLEX has not found an optimal solution, therefore it is not known an optimal solution of the IP problem in a reasonable time for this instance. Still, CPLEX finds a solution with an optimality gap, in percent based on the LP relaxation solved by CPLEX, which is less than 3 percent.

Similarly, for INEM real-life instance, CPLEX software does not succeed in finding a good feasible solution within an acceptable time frame for the IP problem (see Table 5.6). By the time CPLEX is stopped, the best integer solution has an objective value of 1,901,916. Furthermore, it takes approximately 159 seconds to solve the LP relaxation of the root node, obtaining a value of 37,021 for it. In this case, it is still quite small when compared to the objective value. The LP relaxation is not a feasible solution to the original problem, since the variables’ domain is relaxed, i.e. fractional variables are allowed. However, the value of the LP relaxation is used as a lower bound to check the quality of the best integer solution found. The maximal optimality gap as a percentage of the LP relaxation solved by CPLEX for INEM dataset is at 98.05%.

### 5.3.2. CG and diving heuristic methods

Motivated by the limitations of CPLEX, the diving heuristic described in Section 4.3. is used as an alternative to solve the personnel scheduling problem at INEM. For this purpose, different algorithm configurations on the CODU, EVs and INEM dataset are tested in order to understand their impact on the solutions obtained.

The following notation is used to refer to the different algorithm configurations: A/B/C-D, where A refers to the CG method, B to the diving heuristic method, and C-D to the stopping criteria of the CG method based on the values of $\beta_{\text{root}}$ and $\beta_{\text{diving}}$, respectively.

The CG method (A) has values:

- E if a pricing problem is solved for every person in each iteration.
- P if only a pricing problem for person $i$ is solved in each iteration.

<table>
<thead>
<tr>
<th></th>
<th>CODU</th>
<th>EVs</th>
<th>INEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>BFS</td>
<td>4196</td>
<td>14,040</td>
<td>1,901,916</td>
</tr>
<tr>
<td>Total time (s)</td>
<td>3</td>
<td>&gt;18,000</td>
<td>&gt;18,000</td>
</tr>
<tr>
<td>LP relaxation</td>
<td>3941</td>
<td>13,688</td>
<td>37,021</td>
</tr>
<tr>
<td>LP relaxation time (s)</td>
<td>1</td>
<td>75</td>
<td>159</td>
</tr>
<tr>
<td>Gap between BFS and LP relaxation solved by CPLEX</td>
<td>6.08%</td>
<td>2.51%</td>
<td>98.05%</td>
</tr>
</tbody>
</table>

Table 5.6 - Solution for the standard IP model using CPLEX as IP solver on three instances: CODU, EVs and INEM. The BFS denotes the objective value of the Best-Found Solution.
The branching method during diving (B) has values:

- **L** if the branch is on the highest fractional variable.
- **T** if the branching is on all variables with a value above a threshold, \( \delta \). In all tests where branching is done on variables with a value above a certain threshold \( \delta \), it is set \( \delta = 0.6 \).

The stopping criteria (C-D) is based on the fixed number of iterations allowed, respectively:

- \( \beta_{\text{root}} \) is the number of iterations allowed in the root node.
- \( \beta_{\text{diving}} \) is the number of iterations allowed after each branching decision.

\( \beta_{\text{root}} \) and \( \beta_{\text{diving}} \) are both positive integers. The number of columns generated increase with the number of iterations that are allowed. These are expected to improve the quality of the solutions, but impacting the running times of the algorithm. In the limit, when the CG method is solved to optimality, the iterations tolerable are infinite (\( \infty \)).

The solution approach developed is first assessed by testing different algorithm configurations on CODU dataset, since this is the only instance for which an optimal solution has been found (see Table 5.6). The results are shown in Table 5.7, where is possible to compare CPLEX results with the different algorithm configurations tested. For CODU instance, CPLEX finds an optimal solution much faster than the diving heuristic.

### Table 5.7 - Impact of the different algorithm configurations tested on the CODU dataset. BFS denotes the objective value of the Best-Found Solution, 'Objective root' the objective value of the LP relaxation of the root node, and 'Gap' the maximal optimality gap in percent compared to the CPLEX BFS.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>BFS</th>
<th>Total time (s)</th>
<th>Objective root</th>
<th>Root time (s)</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODU</td>
<td>E</td>
<td>L</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>4,196</td>
<td>3</td>
<td>3,941</td>
<td>1</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>E</td>
<td>T</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>4,196</td>
<td>431</td>
<td>4,196</td>
<td>293</td>
<td>0.00%</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>L</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>4,434</td>
<td>175</td>
<td>4,196</td>
<td>44</td>
<td>5.37%</td>
</tr>
<tr>
<td></td>
<td>P</td>
<td>T</td>
<td>( \infty )</td>
<td>( \infty )</td>
<td>4,646</td>
<td>144</td>
<td>4,196</td>
<td>38</td>
<td>9.98%</td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2</td>
<td>30,780</td>
<td>3</td>
<td>30,780</td>
<td>3</td>
<td>86.37%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>10,528</td>
<td>62</td>
<td>22,800</td>
<td>5</td>
<td>60.14%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>9,000</td>
<td>61</td>
<td>18,454</td>
<td>6</td>
<td>56.59%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>10,254</td>
<td>69</td>
<td>11,952</td>
<td>15</td>
<td>59.08%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2</td>
<td>30,780</td>
<td>5</td>
<td>30,780</td>
<td>5</td>
<td>86.37%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>10,936</td>
<td>21</td>
<td>970</td>
<td>5</td>
<td>61.63%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>11,554</td>
<td>27</td>
<td>1,070</td>
<td>7</td>
<td>63.68%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>11,854</td>
<td>23</td>
<td>881</td>
<td>18</td>
<td>64.80%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2</td>
<td>5,914</td>
<td>95</td>
<td>11,335</td>
<td>4</td>
<td>29.06%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>6,180</td>
<td>83</td>
<td>9,695</td>
<td>5</td>
<td>32.19%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>6,334</td>
<td>92</td>
<td>8,473</td>
<td>9</td>
<td>33.75%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>5,622</td>
<td>99</td>
<td>5,251</td>
<td>22</td>
<td>25.36%</td>
<td></td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>2</td>
<td>2</td>
<td>11,056</td>
<td>8</td>
<td>11,352</td>
<td>4</td>
<td>62.05%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>3</td>
<td>9,754</td>
<td>14</td>
<td>9,695</td>
<td>5</td>
<td>56.98%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4</td>
<td>8,692</td>
<td>20</td>
<td>8,473</td>
<td>8</td>
<td>51.73%</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>1</td>
<td>7,012</td>
<td>32</td>
<td>5,251</td>
<td>20</td>
<td>40.18%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
A first conclusion from the data in Table 5.7 is that the diving heuristic finds solutions very close to the optimal solution value when the CG phase is solved to optimality (i.e. the case when the values of column C-D, respectively $\beta_{\text{root}}$ and $\beta_{\text{diving}}$, are infinite). For configuration E/T/$\infty$-$\infty$ the BFS is even an optimal solution. Indeed, this configuration succeeds in finding an optimal solution with a value of 4196, which equals the correspondent value of the objective root. Additionally, for all configurations that are not limited on the CG phase, a lower bound of 4196 is achieved, which corresponds to the BFS obtained by CPLEX. On the other hand, if the CG is prematurely terminated (i.e. the values from columns C-D are positive integers and the number of iterations of the CG is limited) the optimality gaps of the solutions, compared to the optimal solution found by CPLEX, range between 25% and 86%, depending on the configuration. For these tests, since CG is stopped prematurely, the LP relaxation value does not correspond to an actual lower bound.

A second conclusion is that the CG scheme (column A) has an important impact on the computation time, for the root node as well as for the solution, on the different algorithm configurations tested. For the tests where the CG is prematurely terminated, the scheme where a subproblem is solved for person $i$ in each iteration (scheme $P$), requires more time to be solved than the other scheme (scheme $E$). The reason for this is due to the high symmetry of the inputted data, as most people have the skills to do most of the tasks. As such, given the same dual vector, for almost every person the same column is generated, for different people. However, in the master only one or a few of those columns are useful to meet the demand for a given set of tasks on the different days. Whereas in the scheme where a pricing problem is solved for every person in each iteration (scheme $E$), after the addition of a single column, the dual variables are always updated, so the subproblems for the remaining people generate columns with other activities patterns. For the tests where the CG is prematurely terminated, this leads to solutions found by scheme $E$ being poorer as the available columns cannot be combined into a good solution.

A third conclusion from Table 5.7 is related to the branching method (column B) selected. Indeed, when the branching is done on all variables with a value above $\delta$ (method $T$) it is faster when compared to the case where the branching is done on a single variable (method $L$). However, when the CG phase is stopped before the LP optimum has been found (i.e. values for columns C-D are positive integers), branching on a single variable is more likely to generate better solutions, as at each subsequent node more columns can be generated to improve the solution quality. Indeed, P/L/10-1 and P/T/10-1 have optimal gaps of 25.36% versus 40.16%, respectively. The results regarding the unmet demand are expected to corroborate these findings, as it is expectable a lower unmet demand for P/L/10-1 compared to P/T/10-1 unmet demand. However, the latter takes less than third of the time to obtain a solution.

After analyzing the results for CODU instance, an analogous analysis has been considered for the remaining instances.

Table 5.8 shows the results for EVs instance, where the gaps are calculated based on the value of LP relaxation solved by CPLEX solver (i.e. 13,688). The findings regarding the CG scheme (column A) are
comparable to the CODU results: CG method P outperforms method E, but taking longer running times. Configuration P/L/10-1 is unable to obtain a feasible solution in 10 hours running time. Indeed, when running the algorithm with EVs instance, the problem dimensions reach the limits of the capabilities of state-of-the-art commercial IP solvers. This is an effect of the symmetry property, which is extremely low for this dataset. Even though EVs instance comprises a smaller number of variables compared to the INEM dataset, it is in fact harder to solve as it is more constrained, yielding poorer results.

Table 5.8 - Results for different algorithm configurations on EVs dataset. BFS denotes the objective value of the Best-Found Solution, ‘Objective root’ the objective value of the LP relaxation of the root node, and ‘Gap’ the maximal optimality gap in percent compared to the LP relaxation solved by CPLEX.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>BFS</th>
<th>Total time</th>
<th>Objective root</th>
<th>Root time</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>EVs</td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
<td>2,379,710</td>
<td>25</td>
<td>2,379,710</td>
<td>22</td>
<td>99.42%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>65,666</td>
<td>2436</td>
<td>799,304</td>
<td>103</td>
<td>79.16%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>2</td>
<td>2</td>
<td>2,379,710</td>
<td>27</td>
<td>2,379,710</td>
<td>27</td>
<td>99.42%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>3</td>
<td>623,182</td>
<td>117</td>
<td>2,062,130</td>
<td>31</td>
<td>97.80%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>172,782</td>
<td>368</td>
<td>1,810,470</td>
<td>41</td>
<td>92.08%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>211,862</td>
<td>185</td>
<td>799,304</td>
<td>106</td>
<td>93.54%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T</td>
<td>2</td>
<td>2</td>
<td>49,292</td>
<td>76,939</td>
<td>94,662</td>
<td>27</td>
<td>72.21%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>-</td>
<td>&gt;36,000</td>
<td>23,934</td>
<td>112</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>2</td>
<td>2</td>
<td>97,268</td>
<td>242</td>
<td>94,662</td>
<td>43</td>
<td>85.93%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>3</td>
<td>88,976</td>
<td>1177</td>
<td>66,996,5</td>
<td>89</td>
<td>84.62%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>78,604</td>
<td>1641</td>
<td>62,294</td>
<td>117</td>
<td>82.59%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>53,098</td>
<td>9032</td>
<td>47,348</td>
<td>1152</td>
<td>74.22%</td>
</tr>
</tbody>
</table>

The results for INEM instance are shown below in Table 5.9, where it is possible to detect similar trends for the different algorithm configurations results. However, the higher dimensions of the INEM dataset seem to impact the performance of the algorithm. Some conclusions and analysis can be taken.

Table 5.9 - Results for different algorithm configuration on INEM dataset. BFS denotes the objective value of the Best-Found Solution, ‘Objective root’ the objective value of the LP relaxation of the root node, and ‘Gap’ the maximal optimality gap in percent compared to the LP relaxation solved by CPLEX.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>BFS</th>
<th>Total time</th>
<th>Objective root</th>
<th>Root time</th>
<th>Gap</th>
</tr>
</thead>
<tbody>
<tr>
<td>INEM</td>
<td></td>
<td></td>
<td>2</td>
<td>2</td>
<td>2,137,340</td>
<td>52</td>
<td>2,137,340</td>
<td>48</td>
<td>98.27%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>-</td>
<td>&gt;18000</td>
<td>515,331</td>
<td>224</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>2</td>
<td>2</td>
<td>2,137,340</td>
<td>35</td>
<td>2,137,340</td>
<td>32</td>
<td>98.27%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>3</td>
<td>189,732</td>
<td>488</td>
<td>1,727,850</td>
<td>58</td>
<td>80.49%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>164,796</td>
<td>587</td>
<td>1,437,690</td>
<td>72</td>
<td>77.54%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>204,992</td>
<td>434</td>
<td>515,331</td>
<td>224</td>
<td>81.94%</td>
</tr>
<tr>
<td></td>
<td></td>
<td>T</td>
<td>2</td>
<td>2</td>
<td>-</td>
<td>&gt;18000</td>
<td>103,131</td>
<td>58</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>-</td>
<td>&gt;18000</td>
<td>61,703</td>
<td>1765</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>L</td>
<td>2</td>
<td>2</td>
<td>92,640</td>
<td>287</td>
<td>103,131</td>
<td>58</td>
<td>60.04%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3</td>
<td>3</td>
<td>89,388</td>
<td>1592</td>
<td>86,628</td>
<td>130</td>
<td>58.57%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4</td>
<td>4</td>
<td>91,434</td>
<td>2432</td>
<td>80,398</td>
<td>262</td>
<td>59.51%</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10</td>
<td>1</td>
<td>71,112</td>
<td>7852</td>
<td>61,703</td>
<td>1765</td>
<td>47.94%</td>
</tr>
</tbody>
</table>
Regarding the CG scheme (column A), the same tendency as for CODU dataset has been verified. This is to say, the scheme where a subproblem is solved for a single person $i$ after which the master is reoptimized (scheme $P$), improves more the objective value at higher computational efforts.

Moreover, as it occurred on the EVs instance, if the branching is done on the largest fractional variable (method $L$) it takes much longer times. The former branching method has been the more promising as it held lower optimal gaps on the CODU dataset (see Table 5.7). However, on the INEM instance, this method is not able to find any solution within the 5-hour time set limit, irrespective of the CG scheme and the number of tolerable iterations, $\beta_{root}$ and $\beta_{diving}$. The only exception is configuration E/L/2-2, but this has a very poor solution (gap of 98.27%). The reasoning is directly related to the size and low-level of symmetry of the input data of the staff scheduling problem under analysis.

It is quite evident that the stopping criteria of the CG method determined by the number of iterations chosen in the root (column C) and diving (column D) also influences the solution quality. An increase on the number of iterations in the root and diving node improves the accuracy of the solution but extends the running time of the algorithm. Indeed, increasing $\beta_{root}$ seems to be more beneficial for the solution quality than increasing $\beta_{diving}$.

Finally, on INEM dataset, the optimality gaps are obtaining by comparing the BFS to the LP relaxation of the IP formulation solved with CPLEX, which in this case is 37,021. This is a consequence of not knowing an optimal solution and in fact this lower bound might be relatively weak. In the CODU instance this gap is only 6% (see Table 5.6), consequently it is predictable a tiny gap for INEM dataset too. Henceforth, compared to the standard IP approach, the diving heuristic approach with the lowest gap is the one from configuration P/T/10-1 (47.94%). This configuration has a BFS of 71,112 in 7852 seconds (ca. 130 minutes).

Summing up, the results above show that the solutions obtained by the developed solution approach achieve a reasonable trade-off between solution quality and computation time. Specifically, on INEM instance, the diving heuristics clearly outperforms the original IP formulation. Whilst through standard IP no good feasible solution is found in 5 hours, the heuristic algorithm found a solution in very reasonable time, for several configurations. Considering only INEM dataset, configuration P/T/10-1 is the one with better gap at 47.94%, followed by configuration P/T/4-4 with a gap of 59.51% and P/T/3-3 with 58.57%.

5.3.3 Test sets

In the present Section, the algorithm is validated by testing it on the different problem dimensions introduced in Section 5.2.1.1. Beyond this, it is investigated more thoroughly the configuration that achieves the best trade-off between solution quality versus computation time. The purpose is to explore whether the former observations on the relative performance of the different algorithm configurations hold for other problem instances as well.

As it occurred for EVs and INEM dataset, the CG phase is stopped prematurely and therefore the objective root is not an actual lower bound. Consequently, the gaps for the different configurations are calculated based on the LP relaxation solved by CPLEX for the respective dataset.
Despite the sets have been tested in more configurations, only the more relevant results are presented in Table 5.10. For each dataset, it is provided the solution value found by CPLEX. On each instance, the results show four algorithm configurations (E/T/3-3, P/T/2-2, P/T/3-3, and P/T/10-1). The solutions obtained for them are compared with the solution found by CPLEX within the five-hour time limit. Only for TestMD instance the limit has been extended to ten hours. The reasoning for increasing the time limit for instance TestMD is because for configuration P/T/2-2 it already requires a computation time of nine hours (32,373 seconds). Under these circumstances, for this dataset, configurations P/T/3-3 and P/T/10-1 are unable to solve the problem within ten hours, whereas the BFS achieved by configuration P/T/2-2 has a gap of 82.15%. It yields, that extending the planning horizon to two months instead of a single month increases the complexity of the problem.

Moreover, the information available in Table 5.10 shows than an increase on the number of people (instance TestMP) has small impact on the difficulty of the problem, e.g. the gap for the problem solved with CPLEX for TestMP is 98.94%, while the gap for the same test on INEM instance is close at 98.05% (see Table 5.6).
The gap for configuration P/T/3-3 on the instance TestLS is 68.16%. However, the algorithm fails to find a solution for configuration P/T/10-1 after 5 hours of search. So, the result for this test set has relatively poor performance. Indeed, it is suggested that a reduction on the problem’s symmetry, i.e. a reduction on the possibility of permutation of the solutions, increases the problem’s complexity.

Otherwise, the experiments with CPLEX exhibited in Table 5.10 demonstrate that an increase on the symmetry of the problem data (TestHS) simplifies the scheduling problem. It is intuitive as the chance of making bad decisions in the early stages of the heuristic is smaller for the case where every TEPH has the required skills to do every task. Later, at the end of the algorithm the people that haven’t been yet assigned, can be assigned to the free tasks still left, without major problems.

5.3.4. Statistical tests

Following, time and quality statistical tests are conducted to test the statistical significance of these findings.

First, to assess time performance, a one-way Analysis of Variance (ANOVA) is done to compare the computation times over the configurations between the four test sets and the INEM dataset. The test, with an acceptance level set at 0.05, yields a p-value of 0.004668. The results of this analysis are, in more detail, exhibited in Appendix I – Table AI.1 and Table AI.2. It is indicated that there is no significant difference within the required computation times for the five different sets. Despite this p-value being quite significant, when several hypotheses are statistically compared (and in this case, there are five), there are higher chances of non-detected rare events. Therefore, a post test on these data is done to assess each two sets individually, using a Bonferroni correction. This pairwise comparison demonstrate a significant difference for the computation time between the instance TestMD and the other datasets. For the remaining pairs, no significant differences are obtained.

Comparisons on quality performances are also analyzed. A Wilcoxon signed-rank test is chosen due to the small number of algorithm configurations and due to the large variance between the BFS of the datasets. The summarized information for this statistical test may be consulted in – Table AI.3. The acceptance level is set again at 0.05. The test is performed between configuration E/T/3-3 and P/T/3-3 for the same datasets to assess the performance of both CG schemes. The same test is then used to test whether there is a statistical difference in performance between P/T/2-2 and P/T/3-3 and finally between P/T/3-3 and P/T/10-1 configurations, respectively. If no solution has been found for a certain dataset using a given configuration, this dataset is discarded for the corresponding test. The p-values obtained are summarized in Table 5.11.

The results for the p-values demonstrate the p-value between configuration P/T/2-2 and P/T/3-3 is the only one that falls above the acceptance level. For the remaining comparisons, it is demonstrated that there is no significant difference between the objective values obtained.
5.3.5 Analysis of the Best-Found Solution

The best-solution, on INEM dataset, is the one obtained for configuration P/T/10-1. In this Section, the solution for this configuration is analyzed and compared to a planned schedule from INEM. Figure 5.9 presents the results for the tasks assigned to each person over a week of the planning horizon and Figure 5.10 the supply and demand for each task over the same time frame.

From Figure 5.9 it is perceptible a tendency to assign tasks from the respective service since person 67 and person 68 that belong to CODU are just assigned to CODU tasks, while the remaining people (from person 69 to 74) only do tasks from EVs service. Additionally, it seems that people from CODU are about to follow the sequence of stints assigned to their team. In this sample, person 67 and person 68, are from team 5, that as a stint pattern as follows: OOMMAAONNO. Moreover, in this solution some people

<table>
<thead>
<tr>
<th>Algorithm combination</th>
<th>E/T/3-3 vs P/T/3-3</th>
<th>P/T/2-2 vs P/T/3-3</th>
<th>P/T/10-1 vs P/T/3-3</th>
</tr>
</thead>
<tbody>
<tr>
<td>p-value</td>
<td>0.0313</td>
<td>0.1563</td>
<td>0.0625</td>
</tr>
</tbody>
</table>

Table 5.11 - P-values obtained for the Wilcoxon signed rank test.
perform tasks from the respective team(s) but also from other teams. Namely, person 67 performs tasks 4, 8 and 10. Tasks 4 and 8 are not in the set of tasks belonging to this person’s team, but task number 10 belongs to the set of tasks of this person’s team (i.e. team number 5). Another example is person 69 that performs tasks 14, 53 and 57. Tasks 14 and 57 correspond to tasks from his/her teams (respectively, team 6 and 18), but task 53 belongs to a different team (i.e. team 14). On the contrary, there are other people who just perform tasks from the team(s) they are assigned to, such as the case with person 75 that belongs to team 6 and only performs tasks within this team. But in any case, TEPHs only perform tasks if they have the required skills (see Figure 5.6).

Despite the small size of the sample provided in Figure 5.9, it is possible to verify the compliance of some of the hard constraints defined in the model. For every TEPH, the resting time of two shifts between every two assignments is respected, as well as the maximum number of consecutive working days and days-off (6 and 4, respectively).

The overall supply in the BFS solution is 5240, which is larger than the pre-defined 4527 task-demand. The results from configuration P/T/10-1 yield to 13 cases of understaffing versus 719 of overstaffing. There is only 1 understaffing situation at CODU, whereas the remaining 12 occur in EVs. The blue cell in Figure 5.10 shows one of these understaffing cases. In the night shift of the first day it is necessary one person to perform task 14, but in fact there are no TEPHs allocated to this duty, leading to 1 unmet demand. Overstaffing, in red in the figure, occurs when the planned supply exceeds the required demands. In this sample, several cases of overstaffing are present.

5.3.5.1. Validation with a planned schedule at INEM

To polish up the analysis started in the previous Section, the solution from the best configuration P/T/10-1 is compared with a planned schedule provided by INEM, so that the consistency of the model is appraised.

Both the real planned schedule and the best-solution schedule share the same number of days (28 days) and starting day (Monday). The number of teams and tasks is also identical. However, the number of people is 278 (68 in CODU, 210 in EVs), contrasting with the 289 TEPHs of the tests (68 in CODU, 221 in EVs). The overall demand of the INEM current schedule is 4770, versus 4527 of the INEM tested instance. In addition, the INEM current schedule has one public holiday and therefore contractual hours are 132 (instead of 140). Whenever required, indicators are normalized to ensure the veracity of the comparisons.

Table 5.12 - Coverage indicators: INEM real case and best solution case on CODU, EVs and INEM.

<table>
<thead>
<tr>
<th></th>
<th>INEM real case</th>
<th>Best solution case</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CODU</td>
<td>EVs</td>
</tr>
<tr>
<td>Average hours worked (% of contractual hours)</td>
<td>106.95%</td>
<td>82.77%</td>
</tr>
<tr>
<td>Understaffing (% of demand)</td>
<td>7.02%</td>
<td>18.22%</td>
</tr>
<tr>
<td>Overstaffing (% of demand)</td>
<td>1.66%</td>
<td>0.14%</td>
</tr>
</tbody>
</table>
Initially some general indicators are compared, for the two services in the two schedules (Table 5.12). The first indicator specifies the percentage of the real capacity used. It relates to the total number of working shifts and the overall coverage capacity, adjusted for the number of people and contractual hours. The second and third indicators correspond, respectively, to the shortage and surplus of the tasks as a percentage of the overall demand.

The figures obtained show that, in the INEM real situation, people in CODU work in average 6.95% beyond their contractual hours, while in EVs TEPHs work in average only 82.77% of what they are supposed to. In general, for INEM current schedule people work only 88.69% of their predetermined hours. Globally, TEPHs are working below their contractual hours, and therefore INEM is not taking advantage of its full capacity. It may be justified by a flaw organization of the schedule itself but also by other factors, such as formation and training activities, vacations, etc. On the other hand, in the best solution situation, in average a TEPH exceeds his/her working hours by only 3.61% (7.82% in CODU, 2.31% in EVs). To the algorithm solution the averages are extremely close to the ideally 100% and it seems that overtime is being explored so that demand is satisfied as much as possible.

From Table 5.12, the INEM schedule case has, all together, higher levels of understaffing and lower levels of overstaffing, compared to the solution schedule. From both services jointly, the percentage of total tasks that are not satisfied is 15.24% for the INEM case, versus only 0.29% in P/T/10-1 solution. In both scenarios, the percentage of understaffing is lower in CODU than in EVs. Indeed, the regular stint basis pattern followed by the workers in this service simplifies and is beneficial for the organization of the timetables. In terms of surplus, overstaffing corresponds only to 0.55% of the INEM demand and 16.04% of the best solution demand. For the best solution situation, as shortage was more weighted than surplus, these results showing the prevalence of the latter were foreseeable. However, even though understaffing in EVs has been considered much less desirable than in CODU, this percentage is still lower for CODU (0.08% for CODU versus 0.37% for EVs), which reflects the tremendous complexity of the EVs service.

These preliminary results suggest that INEM schedules allow a lot of understaffing due to a lack of specialized people to perform certain tasks (symmetry issues) or even to favour other conditions, possibly personal subjective preferences.

With respect to symmetry, measured as the percentage of workers that have the required skills to perform a certain task, it is 35.95% for the real-life schedule scenario and raised at 48.34% for P/T/10-1 scenario. In the latter case, more flexibility is expected to assign people to tasks and therefore there is more opportunity to explore TEPHs contractual hours, at the same time, avoiding under and overstaffing conflicts. A higher symmetry is beneficial to accomplish better outcomes. Moreover, the tasks with inferior demand exigencies tend to be the ones with less symmetry, such as CODU shift responsible task where only 3.81% and 3.11% of the personnel is able to do it in the INEM and in the solution case, respectively. Undeniably, these low ratios struggle to assign people to tasks when the people that can perform these tasks are not available for some reason (e.g. vacancies, sickness, maternity license). Inherently to this problem context, symmetry is also limited by the tasks’ location and consequently 100% of symmetry would be, in principal, impossible to achieve.
Regarding subjective factors, the average number of Sundays and weekends off per person are analyzed in Table 5.13.

**Table 5.13 - Analysis of the average number of Sundays and weekends off per person in INEM real case and best solution case on CODU, EVs and INEM.**

<table>
<thead>
<tr>
<th></th>
<th>INEM real case</th>
<th>Best solution case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of Sundays off per person</td>
<td>1.69</td>
<td>1.18</td>
</tr>
<tr>
<td>Average number of weekends off per person</td>
<td>0.96</td>
<td>0.67</td>
</tr>
</tbody>
</table>

The table shows that irrespective of the service and case considered, the average number of Sundays-off is always greater than 1. This matches the instruction of the context that imposes everyone to have at least one Sunday off over the planning. The chance of having an entire weekend off is not mandatory and for that reason falls below the former value. Globally, the data suggests that the INEM real case assigns greater importance to this personal preference as in average, a person in INEM has 0.99 weekends off while in the solution it is ensured more Sundays off in average, 1.33 per person opposed to 1.24 of the former case, but slightly fewer full weekends off (0.90).

Another soft constraint of the model is related with the appointments performed outside the respective team. In the context of the problem at INEM it is most likely to impact personnel satisfaction levels. Table 5.14 shows some ratios for this analysis, where three indicators have been considered. The first indicator demonstrates the percentage of people from one service working in the other service. The second and third indicators provide, respectively, the percentage of task-demand satisfied within the team and within the service.

**Table 5.14 - Team assignment indicators: in INEM real case and best solution case on CODU, EVs and INEM.**

<table>
<thead>
<tr>
<th></th>
<th>INEM real case</th>
<th>Best solution case</th>
</tr>
</thead>
<tbody>
<tr>
<td>% people that work in the other service</td>
<td>8.82%</td>
<td>30.88%</td>
</tr>
<tr>
<td>% demands of a task performed by someone within the team</td>
<td>72.52%</td>
<td>56.31%</td>
</tr>
<tr>
<td>% demands of a task performed by someone within the service</td>
<td>84.25%</td>
<td>91.93%</td>
</tr>
</tbody>
</table>

It is verified from Table 5.14 that both schedules and services have a certain percentage of TEPHs working in the other service. In the best solution case, 30.88% of CODU workers do jobs in EVs, while only 6.33% of the people from EVs fulfil tasks in CODU. The lack of full symmetry and the preference
given to work in the respective team keeps these ratios relatively low. Additionally, in the algorithm solution, 56.31% of the CODU demand is satisfied by someone from the team to which the CODU task is assigned to and 91.93% by someone allocated in the CODU service. Whereas, 72.16% of EVs demand is performed by someone from the team and 94.78% by someone from EVs but from another team. These results are related with the weights set for the tests. Within CODU, tasks’ location and skills (except for CODU shift responsible that has less symmetry) are not a matter and therefore low percentages are not alarming, if demand is met as much as possible. Unlike this, in EVs service different tasks’ location enhance the importance of working within the respective team. The ratios on the INEM real case show more commitment to the working teams in both services. This compromise is even higher for EVs for the reasons just mentioned: 83.40% of EVs task-demand is satisfied by someone within EVs task-team. However, it is most likely to be one source for the lower understaffing levels obtained for this case (see Table 5.12).

To conclude, this analysis allows to state that the heuristic method has practical value. INEM real schedule is used as a benchmark solution to confirm the evident application of the proposed model. Indeed, this automated scheduling tool provide schedules in significantly less time, relieving administrative personnel of the difficult and time-consuming task of building schedules manually. It improves schedules’ quality and transparency as the rules the algorithm uses to build a schedule are agreed upon beforehand, therefore increasing employee perception about fairness of the resulting rosters.

5.4. Graphical user interface

This study is enriched by implementing the solution in a graphical user interface (GUI). This easy-to-use tool arises as a necessity, so that the algorithms developed are effectively applied to a real-life case. It simplifies the acceptance of this decision support system and the reception of valuable feedback from experts. Initially, when using the developed GUI, the user needs to select the input data through the Edit Menu. The data is previously prepared in a .txt file format. The GUI allows an easy view of the imported data that are listed in a tree view so that it is simple to be checked and validated by a user (Figure 5.11). If desired, through Clear Data action within the Edit Menu the input data may be removed or edited. Both the objective function weights as well as the algorithm configurations (CG scheme, branching method, etc.) can simply be changed in the algorithm settings menu (see Figure 5.12). The stopping criteria (computation time in seconds or the optimality gap) may be set as well. If the settings are not modified, the model uses the default settings.
Once the input information is available and the settings panel is set, the user shall press the run command to solve the problem. The model runs until the programme finds a solution (see Appendix II – Figure AI.1 and AII.2) or the stopping criteria are achieved. The solution is displayed in two different tabs. A first tab shows the tasks assigned to each person over the planning horizon, while a second tab shows, for each task, the demand and the supply in each time slot (see Figure 5.13 and Figure 5.14, respectively). If there is understaffing, the corresponding cells are highlighted in red. Conversely, overstaffing is marked in blue. Changing assignments after the algorithm has found a solution is possible. The INEM scheduler application allows saving the solution to a .txt file by pressing the Save button. Finally, there is the Help action to guide the user and to resolve doubts regarding the use of the application (see Appendix II – Figure AII.3).
In Chapter 5, it has been analyzed the results of the application on real instances of an optimization and a heuristic approach. The instances are from a case-study at a Portuguese medical emergency institution. The former approach revealed not capable to solve the model for large instances when using CPLEX solver. The latter has been tested for different algorithm configurations and three conclusions can be drawn. First, the CG scheme where a subproblem is solved for every person during each iteration.
is outperformed by the scheme where the master is reoptimized each time a column is added. Second, branching on the largest fractional value is not feasible as the required computation time is too large. Third, it is more advantageous to increase the number of iterations in the root node than during diving. However, this also entails a significant increase in required computation time. The algorithm configurations have been validated through test sets with different dimensions. It was shown that the model can easily handle problems with more people and is robust for the level of symmetry. However, when the planning horizon is extended, the algorithm performs poorly. This and the slow convergence of the CG phase show that the pricing problems are the bottleneck of the algorithm.

From all the tests carried, the best solution found was for configuration P/T/10-1. For this, there is only 13 unmet demand. This solution has been compared with a real schedule case provided by INEM. The INEM real case seems to assign a greater importance to personnel preferences, while the solution case schedule seeks primarily to satisfy the pre-defined demand. The different indicators studied showed that the proposed model has practical meaning. Finally, the user-friendly Graphical User Interface (GUI) is shown. This makes possible the real application of the model to the context at INEM or to any other comparable context.

The findings of this thesis can be consolidated for future research so that the model is upgraded in performance and applicability aspects. The diving heuristic is able to deal with variations in the problem’s parameters and constraints as well as to find good quality solutions for real-life instances in relatively short computation times. However, the solution quality still has margin to be improved. Alternative heuristics can be explored to solve the model, namely a combination of a constructive and a variable neighborhood search (VNS) method.

In the next Chapter, the key conclusions relatively to the current real staff scheduling problem are discussed. Moreover, remarks and some advices considering staff scheduling problems are considered.
6. Conclusions and Future Remarks

This dissertation addresses a personnel scheduling problem and develops a mathematical model to cope with it. The developed model has been applied to a real-life situation at a medical emergency institution in Portugal. Several contributions from this study can be identified. First, the problem studied, due to its characteristics is innovative on the existent literature in the area. The developed model as well as the proposed algorithm to solve it add value to the existent literature. Second, the model is tested in a real case, the particular context at INEM. In fact, the study was proposed by INEM to address their current staff scheduling problems, and this action would possibly reduce several costs and enhance staff satisfaction. Third, a graphical user-friendly interface is created, so that the results of the model are easily communicated.

In an initial phase of this work, and to identify the most convenient way to respond to the INEM proposal, Chapter 3 addresses a literature review to understand if there are similar problems in the literature and how they are currently being treated. It has been verified that staff scheduling is a main concern to several administrations, in distinct activity sectors. Various gains arise from improving scheduling and management of the resources. In this phase, it has been identified that the development of a mathematical model would be the most appropriate methodology to tackle the INEM staff scheduling problem. Over the last decades, novel solution techniques (mostly heuristic) have been explored to solve mathematical models, but they often fail in the implementation phase. When dealing with large and multi-constrained problems, there is still high opportunities for improvement. Hence, the present study aims to enrich the existent literature in this field.

The model is primarily formulated as a standard integer program (Section 4.2.). This formulation fails when applied to large problem instances such as the one at INEM, so it was developed a CG formulation that decomposes the problem on its staff members. It assigns each person to a column that corresponds to an activity pattern. The solution from CG might not be integer thus the diving heuristic is then applied to find integer solutions. Two CG schemes as well as two branching methods have been implemented. Additionally, as the CG phase requires a prohibitively large amount of time to solve to optimality, branching is terminated prematurely after a fixed number of iterations (Section 4.3.).

This general framework is applied to the real-life context at INEM in Chapter 5 INEM Case-study. The optimization of the integer program only finds an optimal solution for the smallest dataset (i.e. CODU dataset). For the remaining instances, the heuristic approach shows better results than the exact method. The heuristic is validated by testing it different dimensions instances. The CG scheme where a subproblem is solved for a single person \(i\) after which the master is reoptimized improves more the objective value but requiring more computation time. For large dimensions, branching is only feasible when it is done above a threshold. Regarding the performance of the algorithm on the test sets, the model can easily handle problems with more people and is robust for the level of symmetry in the skills of workers. The test in which the size of the planning period is extended leads to a poor performance of the algorithm.
In general, the algorithm is able to find good quality solutions in reasonable computation time. For the INEM instance, the results demonstrated that the BFS is for configuration P/T/10-1, with a gap of 47.94%. When comparing this solution to a INEM planned schedule several conclusions can be draw (Section 5.3.4.). The solution case is exploring more the TEPHs contractual hours, meaning overtime is being used to satisfy the demand, as much as possible. INEM case allows more understaffing situations. Understaffing situations are related to the lack of symmetry and to the desire of meeting more personal preferences. It is concluded that a higher symmetry, enhances the flexibility to assign people to tasks, therefore leading to enhanced schedules. The INEM real case schedule provides on average more weekend-off but less Sundays off to its workers. There is a higher commitment to the working teams, especially in EVs service, within the schedule of the INEM real scenario than within the best solution scenario. All in all, the analysis shows that the solution from the best configuration has practical value. Indeed, the analysis carried in this dissertation provides valuable insights not only to upgrade the scheduling making process but also to optimize the utilization of human and material resources at INEM.

6.1. Recommendations to INEM

The model and the results of this dissertation have been closely discussed and readjusted with schedule makers at INEM, so that the real-life context at INEM is accurately expressed. In INEM the scheduling process is a periodic task, that is still manually performed. A first guidance is that the use of an aiding automatic scheduling application such as the one developed in this thesis, significantly decreases the time spent to build schedules, at the same time, boosting transparency and equity levels. Ultimately, it is meaningful to raise INEM’s productivity standards and resources exploitation.

Moreover, compared to what is currently being done at INEM, the integration of schedules of different services in case the services share partially or totally the same workforce holds several advantages. From the analysis developed, as this is a shift-based demand problem, it is essential the definition of the demand requirements beforehand and upon this, set the real surplus available. The prior characterization of the working teams, its tasks, location and people cannot be neglected too. Challenges arise e.g. in months with more public holidays or months in which a great proportion of workers take holidays. INEM can either motivate TEPHs to work in overtime (until a certain point) or consider hiring more temporary people to cover potential problematic periods. Another opportunity to overcome this is the specialization of TEPHs in as many tasks as possible (increase symmetry) which simplifies the scheduling process. In fact, if location was not an issue, under a homogeneous workforce, anyone could fill in an empty space in the schedule. As a conclusion, this study aims to beneficially impact some scheduling decisions, within INEM context.

6.2. Limitations and future remarks

The proposed model is flexible to deal with variations in parameters and constraints, and is possible to be applied to several real-life staff scheduling problems. However, to consolidate the findings, it could be interesting to exploit the application of the proposed framework to more real-world problems, even from sectors other than the healthcare sector.
A limitation of the solutions from the heuristic proposed in this dissertation is the solution gap obtained. This gap may be decreased by either refining the developed approach or by trying new ones. The current diving heuristic could be improved for instance by improving the algorithm that is used to solve the subproblems (pricing problems) as this is the bottleneck of the algorithm. Another interesting idea for a future research is to investigate other heuristics such as a fix-and-optimize heuristic embedded within a variable neighborhood decomposition search framework. This idea has already being study in a paper that has been developed along with this dissertation (Vermuyten et al, 2017). In the primary constructive method phase a physical solution is generated. Then it is used in the VNS phase, which starts by choosing a neighborhood, where little changes are done to improve the current solution arising from the former method. The size of the neighborhoods may be adjustable. Within this, the current solution is fixed, and the variables of the neighborhood are leased free. An IP solver finds a local optimum in this neighborhood. If the candidate solution improves the solution it is updated, otherwise the current solution is kept. Once all the neighborhoods have been visited, there is a shake phase where random changes in the solutions are done. The shaking phase uses a fix people neighborhood. The process is iterated until computation time runs out. This idea is being developed in a recent paper (Vermuyten et al, 2017).

Finally, concerning the model itself, and although it was intended to be as wide-ranging as possible, it may perhaps comprise more characteristics and constraints to become more realistic and closer to the INEM application context. It could be interesting the definition, as an input, of the holidays and trainings days for each worker. Besides and although the model handles a variety of constraints, it could include additional fairness aspects such as a criterion for the attribution of the days-off or holidays. To extend the model, the inclusion of the accumulated hours per employee in a straightforward way, from one planning period to another, is a prospect too. Develop a backup plan to deal with uncertainty, such as unpredictable absences. As future developments, the model can be used to study more services than the ones considered or more workforce types (i.e. physician, nurses and psychologists). Despite the approximation to the reality, these suggestions would significantly increase the complexity of the model and therefore would require more computational efforts. The satisfactory results encourage further research of the subjects of this thesis.
References


Appendix I – Time and Quality Statistical Tests

A one-way Analysis of Variance (ANOVA) is performed to demonstrate the statistical significance between the running times of the computational experiments. The computational times are available in Table Al.1. The acceptance level of the ANOVA is set at 0.05. The analysis is done in Excel. The results exhibited in Table Al.2 demonstrate a p-value of 0.004668 for the difference between groups, showing that the time results are statistically significant.

Table Al.1 - Summary table with the computational results (in seconds) for the instances statistically tested in the different algorithm configurations.

<table>
<thead>
<tr>
<th></th>
<th>INEM</th>
<th>TestMD</th>
<th>TestMP</th>
<th>TestLS</th>
<th>TestHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/T/3-3 time (s)</td>
<td>488</td>
<td>2,304</td>
<td>67</td>
<td>1,492</td>
<td>326</td>
</tr>
<tr>
<td>P/T/2-2 time (s)</td>
<td>287</td>
<td>32,373</td>
<td>432</td>
<td>481</td>
<td>135</td>
</tr>
<tr>
<td>P/T/3-3 time (s)</td>
<td>1,592</td>
<td>36,000</td>
<td>1,142</td>
<td>1,714</td>
<td>231</td>
</tr>
<tr>
<td>P/T/10-1 time (s)</td>
<td>7,852</td>
<td>36,000</td>
<td>10,952</td>
<td>18,000</td>
<td>4,933</td>
</tr>
</tbody>
</table>

Table Al.2 - Summary of the ANOVA analysis results conducted to validate the computational times.

<table>
<thead>
<tr>
<th>SUMMARY</th>
<th>Groups</th>
<th>Count</th>
<th>Sum</th>
<th>Average</th>
<th>Variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>INEM</td>
<td>4</td>
<td>10219</td>
<td>2554.75</td>
<td>12800630.25</td>
<td></td>
</tr>
<tr>
<td>Test MD</td>
<td>4</td>
<td>106677</td>
<td>26669.25</td>
<td>266774654.3</td>
<td></td>
</tr>
<tr>
<td>Test MP</td>
<td>4</td>
<td>12593</td>
<td>3148.25</td>
<td>27265222.92</td>
<td></td>
</tr>
<tr>
<td>Test LS</td>
<td>4</td>
<td>21687</td>
<td>5421.75</td>
<td>70604576.25</td>
<td></td>
</tr>
<tr>
<td>Test HS</td>
<td>4</td>
<td>5625</td>
<td>1406.25</td>
<td>5534084.917</td>
<td></td>
</tr>
</tbody>
</table>

ANOVA

<table>
<thead>
<tr>
<th>Source of Variation</th>
<th>SS</th>
<th>df</th>
<th>MS</th>
<th>F</th>
<th>P-value</th>
<th>F crit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Between Groups</td>
<td>1806912453</td>
<td>4</td>
<td>451728113.3</td>
<td>5.897554932</td>
<td>0.0046677660</td>
<td>3.055568</td>
</tr>
<tr>
<td>Within Groups</td>
<td>1148937446</td>
<td>15</td>
<td>76595829.72</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>2955848899</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To test the quality of the solution it is employed a Wilcoxon signed-rank test. The information regarding the BFS obtained in the computational tests is summarized in Table Al.3. The test is solved by using an online Wilcoxon signed-rank test calculator, and then corroborated by applying the singrank function available on MATLAB.

Table Al.3 - Summary table with the BFS results for the instances statistically tested in the different algorithm configurations. In case the solution has not been found for a certain dataset using a given configuration, that dataset is discarded for the corresponding test.

<table>
<thead>
<tr>
<th></th>
<th>INEM</th>
<th>TestMD</th>
<th>TestMP</th>
<th>TestLS</th>
<th>TestHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>E/T/3-3 BFS</td>
<td>189,732</td>
<td>636,206</td>
<td>2,367,270</td>
<td>307,336</td>
<td>524,058</td>
</tr>
<tr>
<td>P/T/2-2 BFS</td>
<td>92,640</td>
<td>282,926</td>
<td>133,084</td>
<td>154,374</td>
<td>111,498</td>
</tr>
<tr>
<td>P/T/3-3 BFS</td>
<td>89,368</td>
<td>133,838</td>
<td>129,990</td>
<td>112,578</td>
<td></td>
</tr>
<tr>
<td>P/T/10-1 BFS</td>
<td>71,112</td>
<td>95,132</td>
<td>83,608</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix II - More Visualizations of the GUI Developed

Figure AII.1 - GUI visualization when the algorithm is running. The programme provides an actual vision on the current parameters.

Figure AII.2 - GUI visualization once the algorithm finds a solution. A warning window pops up announcing the algorithm has found a solution.
Figure AII.3 - GUI Help action. A tutorial explaining how to use the INEM shift scheduling problem is available for consultation.