

**On establishing the additive value functions  
compatible with the public Portuguese hospitals'  
quality, efficiency, and access related criteria**

**Sergi Fortuny González**

Thesis to obtain the Master of Science Degree in

**Industrial Engineering and Management**

Supervisors: Prof. José Rui de Matos Figueira

Prof. Diogo Filipe da Cunha Ferreira

**Examination Committee**

Chairperson: Prof. Ana Isabel Cerqueira de Sousa Gouveia Carvalho

Supervisor: Prof. José Rui de Matos Figueira

Member of the Committee: Prof. Ana Sara Silva Rodrigues da Costa

**January 2021**



**Declaração**

Declaro que o presente documento é um trabalho original da minha autoria e que cumpre todos os requisitos do Código de Conduta e Boas Práticas da Universidade de Lisboa.

**Declaration**

I declare that this document is an original work of my own authorship and that it fulfils all the requirements of the Code of Conduct and Good Practices of the Universidade de Lisboa



# Acknowledgements

I would like to thank all those people who have made this thesis possible.

First of all, to my supervisors Professor José Rui Figueira and Professor Diogo Cunha Ferreira, for their help in all the steps that lead to this dissertation. I must sincerely thank Professor Figueira, for the support in the technical field and for monitoring the evolution of the work. These past months have been turbulent due to the global COVID-19 pandemic, and although it may not have been a close relationship, he has always stayed in regular contact and has always been available to help if any need. He has been understanding when I have been away for a while but has been demanding when needed, too.

During the fieldwork I had some very specific doubts, and he did not hesitate for a moment to offer the contact of some expert on the subject in order to help me. In this regard, I would like to thank Professor Salvatore Corrente for his attention and willingness to help me.

Also, I would like to thank the entire *hSNS*<sup>1</sup> team, for making me feel like one more member and for regularly reporting on the status of the overall project and its evolution. The meetings have been very educational and have helped me understand the importance of my work.

I must thank Instituto Superior Técnico for giving me this unique opportunity to do the Master's Thesis in Erasmus program, and in the same way I am grateful to my university in Barcelona Universitat Politècnica de Catalunya (UPC) and to my faculty ETSEIB for the education received for the last 6 years and for allowing me to start this personal and academic adventure.

Finally, thank you to my family and close friends, for having accompanied me on this journey. I really appreciate their unconditional support, especially during the hard times of the quarantine. Being in another country away from your loved ones in this difficult situation has not been easy, but it has been a personal learning that will accompany me all my life.

---

<sup>1</sup> This project is funded by the Portuguese Foundation for Science and Technology (FCT) with the reference PTDC/EGE-OGE/30546/2017.



# Abstract

The main objective of this thesis is to develop the  $UTA^{GMS}$  method to be implemented in the field of Portuguese public hospitals, in order to measure their performance in terms of its quality and access dimensions by establishing their additive value functions. The  $UTA^{GMS}$  is a Multiple Criteria Decision Analysis (MCDA) method that provides a ranking of alternatives using a set of additive value functions which result from an ordinal regression.  $UTA^{GMS}$  has been developed in MATLAB software in combination with CPLEX (an IBM ILOG CPLEX Optimization Studio extension). It is intended to be used interactively, so the Case Study presented in this dissertation is done in collaboration with an expert on the subject (Decision-Maker). Hospital data is collected from an official source and processed in order to be used in the method. The results of the Case Study are different rankings of the hospitals analyzed, depending on the information provided by the Decision-Maker. The result is a robust model with room for improvement. This  $UTA^{GMS}$  method can be used for different types of data and is not restricted to hospital data, making it a method to be further developed in future research given its clear potential.

**Key-words:** Hospitals, Quality, Multiple Criteria Decision Analysis, UTA method, UTA GMS method.





# Resumo

O principal objectivo desta tese é desenvolver o método UTA<sup>GMS</sup> a ser implementado na área dos hospitais públicos portugueses, de forma a medir o seu desempenho ao nível da sua qualidade e dimensões de acesso através do estabelecimento das suas funções de valor aditivo. O UTA<sup>GMS</sup> é um método de Análise de Decisão de Múltiplos Critérios (MCDA) que fornece uma classificação de alternativas usando um conjunto de funções de valor aditivo que resultam de uma regressão ordinal. UTA<sup>GMS</sup> foi desenvolvido em software MATLAB em combinação com CPLEX (uma extensão IBM ILOG CPLEX Optimization Studio). Pretende-se que seja utilizado de forma interactiva, pelo que o Estudo de Caso apresentado nesta dissertação é realizado em colaboração com um especialista na matéria (Tomador de Decisão). Os dados hospitalares são coletados de uma fonte oficial e processados para serem usados no método. Os resultados do Estudo de Caso são diferentes classificações dos hospitais analisados, dependendo das informações fornecidas pelo Tomador de Decisão. O resultado é um modelo robusto com espaço para melhorias. Este método UTA<sup>GMS</sup> pode ser utilizado para diversos tipos de dados e não se restringe aos dados hospitalares, sendo um método a ser desenvolvido em pesquisas futuras pelo seu evidente potencial.

**Palavras-chave:** Hospitais, Qualidade, Análise de Decisão de Múltiplos Critérios, Método UTA, Método UTA<sup>GMS</sup>.



# Table of contents

- 1 Introduction ..... 1**
  - 1.1 Motivation .....1
  - 1.2 Objectives .....2
  - 1.3 Methodology .....2
  - 1.4 hSNS project .....3
  - 1.5 Structure of the thesis .....4
- 2 Multiple-Criteria Decision Analysis (MCDA) ..... 7**
  - 2.1 Introduction to MCDA .....7
  - 2.2 Concepts .....8
  - 2.3 MCDA methods .....9
    - 2.3.1 Multi-Attribute utility theory ..... 9
    - 2.3.2 Outranking methods ..... 10
- 3 UTA method ..... 13**
  - 3.1 Overview .....13
  - 3.2 Development of the UTA method .....15
  - 3.3 UTA<sup>GMS</sup> .....17
    - 3.3.1 Introduction to UTA<sup>GMS</sup> ..... 17
    - 3.3.2 Properties of the relations  $\succeq N$  and  $\succeq P$  ..... 18
    - 3.3.3 Implementation of UTA<sup>GMS</sup> ..... 19
- 4 Implementation of UTA<sup>GMS</sup> in MATLAB ..... 23**
  - 4.1 Software description .....23
    - 4.1.1 IBM ILOG CPLEX Optimization Studio..... 23
    - 4.1.2 MATLAB - CPLEX ..... 25
  - 4.2 Implementation of the model .....27
- 5 Case study ..... 35**
  - 5.1 Data and sample .....35
    - 5.1.1 Alternatives ..... 35
    - 5.1.2 Criteria ..... 35
  - 5.2 Preference parameters .....38
  - 5.3 Results and discussion .....40

**6 Conclusions..... 49**

6.1 Conclusions.....49

6.2 Future research .....50

Appendix A: Some parts of the Matlab code .....53

# List of Tables

Table 1: MCDA problems and methods ..... 9

Table 2: Decision matrix ..... 10

Table 3: Properties of the binary relations  $\succeq_N$  and  $\succeq_P$  [22] ..... 18

Table 4: Necessary ranking computed in terms of  $d(a,b)$  ..... 21

Table 5: Possible ranking computed in terms of  $d(a,b)$  ..... 21

Table 6: LP problem ..... 26

Table 7: Inputs for an LP problem ..... 26

Table 8: Output for an LP problem ..... 26

Table 9: Meaning of *exitflag* values ..... 27

Table 10: Variables ..... 29

Table 11: Objective function coefficients ..... 29

Table 12: Ordered performances ..... 29

Table 13: Coefficients table ..... 30

Table 14: Matrix of equality constraints coefficients ..... 30

Table 15: Matrix that completes the equality constraints ..... 30

Table 16: Matrix of inequality constraints coefficients ..... 31

Table 17: Example of auxiliary matrix ..... 31

Table 18: Matrix that completes the inequality constraints ..... 31

Table 19: Lower bounds of  $x$  ..... 31

Table 20: Upper bounds of  $x$  ..... 32

Table 21: Partial preorder ..... 33

Table 22: Name of the alternatives ..... 36

Table 23: Name of the criteria ..... 36

Table 24: Performances of each alternative ..... 38

Table 25: Ranking for the 1st iteration ..... 40

Table 26: Ranking for the 2nd iteration ..... 42

Table 27: Ranking for the 3rd iteration ..... 43

Table 28: Ranking for the 4th iteration ..... 44

Table 29: Ranking for the 5th iteration ..... 45



# List of Figures

- Figure 1: Sequential stages followed ..... 3
- Figure 2: hSNS project logo ..... 3
- Figure 3: The disaggregation-aggregation approach ..... 13
- Figure 4: Example of data definition ..... 24
- Figure 5: Example of decision variables definition..... 24
- Figure 6: Example of a UTA method linear program..... 24
- Figure 7: Example of importing and exporting data..... 25
- Figure 8: Example of importing data ..... 28
- Figure 9: Building of Necessary Ranking matrix ..... 33
- Figure 10: Code for Necessary Ranking graph..... 33
- Figure 11: Ranking for the 1st iteration ..... 41
- Figure 12: Ranking for the 2nd iteration ..... 42
- Figure 13: Ranking for the 3rd iteration..... 43
- Figure 14: Ranking for the 4th iteration ..... 44
- Figure 15: Ranking for the 5th iteration ..... 45
- Figure 16: Ranking for the 6th iteration ..... 47
- Figure 17: Ranking for the 7th iteration ..... 47
- Figure 18: Ranking for the 8th iteration ..... 48





# List of Equations

Equation 1 ..... 10  
Equation 2 ..... 15  
Equation 3 ..... 15  
Equation 4 ..... 15  
Equation 5 ..... 15  
Equation 6 ..... 16  
Equation 7 ..... 16  
Equation 8 ..... 16  
Equation 9 ..... 18  
Equation 10 ..... 18  
Equation 11 ..... 19  
Equation 12 ..... 20  
Equation 13 ..... 20



# Glossary

<b>ACSS</b>	Administração Central do Sistema de Saúde
<b>AHP</b>	Analytic Hierarchy Process
<b>DM</b>	Decision-Maker
<b>ELECTRE</b>	Elimination and Choice Expressing Reality
<b>EU</b>	European Union
<b>FCT</b>	Fundação para a Ciência e a Tecnologia
<b>GAIA</b>	Geometrical Analysis for Interactive Aid
<b>IDE</b>	Integrated Development Environment
<b>LP</b>	Linear Programming
<b>MACBETH</b>	Measuring Attractiveness by a Categorical Based Evaluation Technique
<b>MAUT</b>	Multiple Attribute Utility Theory
<b>MCDA</b>	Multiple Criteria Decision Analysis
<b>MCDM</b>	Multiple Criteria Decision Making
<b>NHS</b>	National Health Service
<b>OPL</b>	Open Programming Language
<b>PPP</b>	Public-Private Partnerships
<b>PROMETHEE</b>	Preference Ranking Organization METHod for Enrichment of Evaluations
<b>SNS</b>	Serviço Nacional de Saúde
<b>UTA</b>	UTilités Additives
<b>VHI</b>	Voluntary Health Insurance



# 1 Introduction

This chapter presents an introduction to the thesis. It consists of an explanation of the reasons why this research is made (Section 1.1), a description of the objectives to be achieved (Section 1.2), the methodology used throughout the thesis (Section 1.3), a brief description of the project in which this study is being conducted (Section 1.4) how this document is organised (Section 1.5).

## 1.1 Motivation

Portugal has a total population of 10.3 million people, which is mainly concentrated in big cities as Lisbon or Oporto and along the coast, leaving an elderly population inland. It has the lowest fertility rate among the European Union (EU), with a 1.2 births per inhabitant rate. These facts, together with decreasing immigration and increasing emigration, is making the population of Portugal decrease since 2010. There are many factors that make the healthcare systems of all countries face substantial challenges of all kind, and Portugal is no exception [1].

Portuguese health system has three co-existing system: National Health Service (NHS), special health insurance for particular professions or sectors, and private Voluntary Health Insurance (VHI). NHS has a total of 113 hospitals out of 225 that Portugal has [2]. All residents in Portugal have universal access to health care provided by the NHS, which is mainly financed by taxes. It was established in 1979 and has progressively become more transparent. Since 2016, Portuguese citizens can choose to go to hospitals out of their resident area and there is a lot of information available in real time, so everyone can consult the waiting times for a specific procedure for each hospital [3]. The economic crisis in 2009 forced the Portuguese government to reduce the public spending in public health service, and decreased its quality and service, having repercussions during the following years and to this day [4]. There has been a lot of criticism of the government's management of the healthcare system, and as a result, public opinion about the system has declined in recent years. Portuguese public health offers a universal service to all Portuguese citizens, who pay it through their taxes. Therefore, his opinion on the system is of extreme importance [5].

This is the reason why it is essential to assess the quality of healthcare services of Portuguese public hospitals, by analysing the performances of the hospitals on representative criteria and searching for solutions that improve them in a cost-effective way [6].

This is the main objective of the *hSNS* Project, and this thesis is part of it. The characteristics of the project will be further described in Section 1.4. Once the representative criteria have been chosen and treated and the corresponding performances are given, this dissertation focuses on a more technical part to get a ranking of hospitals in case any person had to choose one. To do this, several methods could have been used, but the UTA<sup>GMS</sup> method has been selected. The reasons why this decision is made is explained in Section 2.3. To implement the method there are multiple possibilities of programming languages and computer programs, and the selected software has been MATLAB complemented with IBM CPLEX. The reasons for this selection are explained in Section 4.1.

## 1.2 Objectives

The objectives of this dissertation are the following:

- Implement the UTA<sup>GMS</sup> method in a MATLAB-CPLEX environment.
- Apply the implemented method to a real case in health sector to get the additive value functions of the public Portuguese hospitals.

The accomplishment of these objectives is addressed in Section 6.

## 1.3 Methodology

With the purpose of achieving these objectives listed, the process followed during the realization of this thesis can be summarized in the following stages, also schematized in Figure 1:

1. Definition of the problem that motivates this thesis, by exploring the background and analysing the Portuguese public health system. Once identified, the problem can be described and detailed.
2. As MCDA methods were a topic unknown to us, we did a literature review of multiple documents related to the topic. We have acted the same way for the more specific methods UTA and UTA<sup>GMS</sup>.

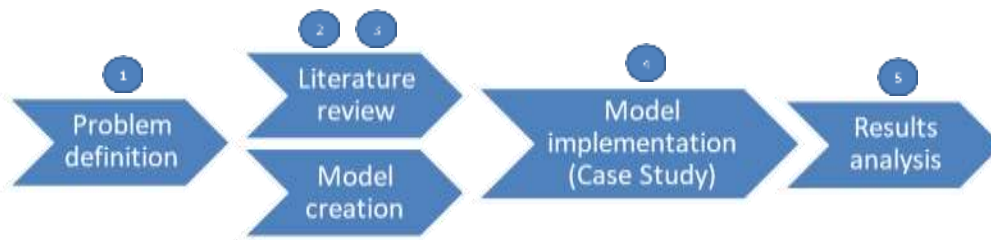


Figure 1: Sequential stages followed

3. Based on examples found in the documents during the literature review, we practiced with them in order to internalize the knowledge and be able to have a more solid basis for further learning. The model has been created little by little as the reading progressed.

The previous two steps have been done simultaneously, so that every time we learned new details or new methods, we practiced them even if they were not part of the final goal of the study.

4. We carried out a Case Study with the data provided by the project colleagues on the Portuguese public hospitals.
5. Finally, we analysed the results obtained and drew the corresponding conclusions.

## 1.4 hSNS project

This section presents the research project called hSNS, a global project in which this dissertation is included in. This project is funded by the Portuguese Foundation for Science and Technology (FCT) with the reference PTDC/EGE-OGE/30546/2017 (Figure 2). It began in 2018 and is expected to be completed in 2021.



Figure 2: hSNS project logo

This project aims to measure the Portuguese hospital performance in terms of its quality and access dimensions to:

- Improve the quality of healthcare services that will be delivered
- Support management by monitoring and controlling performance indicators
- Benchmark healthcare providers
- Accountability and governance
- Better finance healthcare providers (according to their performance)

In order to do this, seven different tasks have been defined and each one has its own objectives and research team. These tasks are the following ones:

1. Project management, coordination, and dissemination
2. Literature review and state of the art
3. Data collection
4. Performance model and conceptualization and formulation
5. Maturity model conceptualization and formulation
6. Models implementation and results achievement
7. Robustness analysis, conclusions and recommendations

The project is coordinated by Professor José Rui Figueira, and the project team is formed by himself, four senior researchers, two researchers, one PhD Student and nineteen MSc Students, for a total number of twenty-seven members.

The current outcomes of the project are multiple publications together with national and international events, which can be consulted in their webpage<sup>2</sup> in the *Outcomes* section.

## 1.5 Structure of the thesis

The thesis is structured in seven main chapters, which are in line with the objectives and the methodology just described.

Section 2 provides a theoretical view of MCDA methods and is followed by more technical part in Section 3, where the method used UTA<sup>GMS</sup> is formally described and the procedure subsequently implemented in Section 4 is described. This Section 4 also describes the software used and contains

---

<sup>2</sup> Specific information of the hSNS project can be found online at: <https://hsns.eu/>



the most practical part of the work. Section 5 carries out the Case Study, which describes the data used and its origin as well as the procedure carried out to implement the model, together with the results and its analysis. Finally, Section 6 assesses the work in the form of conclusions and presents possible future studies in the field developed.



## 2 Multiple-Criteria Decision Analysis (MCDA)

This chapter presents a literature review of MCDA. It consists of an introduction to the concept (Section 2.1), a description of its methods (Section 2.2) and the common uses of it in healthcare (Section 2.3).

### 2.1 Introduction to MCDA

Taking decisions in your daily life may usually seem an easy work, but every decision requires balancing a set of multiple factors that have an effect on the outcome of it. For many years, a decision problem relied merely on the definition of a single criterion, uniting the aspects of the decision situation into a single scale of measure. In reality, however, a decision problem is the existence of a bunch of complex and conflicting information. In line with this, a new way to look at decision problems has gained more and more attention of researchers - the Multiple Criteria Decision-Analysis (MCDA).

The study of decision problems has a long history, and in the last decades has been one of the major research fields in decision sciences. The earliest known reference relating to Multiple Criteria Decision Making can be traced to Benjamin Franklin (1706-1790), who allegedly had a simple paper system for deciding important issues [7]. Take a sheet of paper. On one side, write the arguments in favour of a decision; on the other side, write the arguments against. Strike out arguments on each side of the paper that are relatively of equal importance. When all the arguments on one side are struck out, the side which has the remaining arguments is the side of the argument that should be supported. Supposedly Franklin<sup>3</sup> used this in making important decisions.

It was in late 60s that multicriteria decision making research experimented an explosive growth. In 1972 the first international conference in Multiple Criteria Decision Making was done. From then on, this is an active area of research as can be observed in the many conferences organised every year and the specializes journals (e.g. *Journal of Multi-Criteria Decision Analysis*) [8].

Although MCDM and MCDA may seem the same concept, it is actually not. MCDA researchers put their efforts to develop methods to help the user to understand the preference model behind his/her decisions. This approach includes tools to represent the data or interactive methods to build a model of the preferences [9]. On the other hand, the American School of MCDM takes a more descriptive

---

<sup>3</sup> More information about this procedure and other related stories can be found online at the website of "MCDM Society": <https://www.mcdmsociety.org/content/short-mcdm-history-0>

approach. The goal is to build a model of behaviour of the decision makers and let them apply the model to solve new problems. MCDM is mainly developed in United States of America, while MCDA is the one adopted by most of the European researchers [10].

The method that is explained in this thesis fits better with the MCDA approach, as it takes into account that the decision maker must understand the solution and the degree of confidence attributed to it, so the solution can be improved by interacting with him/her.

## 2.2 Concepts

In this section, we define the main concepts and nomenclature used in this document [11].

**Decision-maker:** Individual or group of individuals for whom the methods are developed and implemented. It will be shortened as DM.

**Alternative:** Object of the decision. We will follow the next notation, where  $m$  is the number of possible alternatives in the set  $A$ .  $A = a_1, a_2, \dots, a_m$

**Scale:** Set of elements ranked in a complete order, reflecting the preferences of the decision-maker. The scales are usually numerical, ordinal or categorical.

**Criterion:** Application  $g$  from the set of alternatives to a scale, giving meaning to compare two alternatives  $a_1, a_2$  by comparing the value of  $g(a_1), g(a_2)$ . We will follow the next notation, where  $n$  is the number of criteria.  $G = g_1, g_2, \dots, g_n$

**Threshold:** Value used to consider the imprecision at comparing alternatives.

**Preference relation:** Binary relation between two alternatives that expresses the preference of one over the other.

## 2.3 MCDA methods

Multi-criteria decision Analysis (MCDA) is a discipline that helps decision-makers to make decisions when several conflicting criteria need to be evaluated. When facing a decision problem, the first task of a decision-maker is to identify the type of problem. There are four problem formulations within the MCDA context [12]:

- Choice problem, which aims to select a single best alternative or to reduce the group of alternatives to a subset of equivalent or incomparable alternatives.
- Classification or sorting problem, which aims to sort the alternatives into predefined categories.
- Ranking problem, which aims to rank the alternatives from best to worst. The order can be complete or partial if we consider incomparable alternatives.
- Description problem, which aims to help the description of alternatives and their consequences.

Table 1: MCDA problems and methods

Choice problems	Ranking problems	Sorting problems	Description problems
PROMETHEE	PROMETHEE	FlowSort	GAIA
ELECTRE I	ELECTRE III	ELECTRE-Tri	
UTA	UTA	UTADIS	
AHP	AHP	AHPSort	

Selecting the appropriate method for the decision problem is very important (see Table 1). These methods can be also classified into two different families: Aggregation approaches (based on the Multi-Attribute Utility Theory) and Order-focussed approaches (based on Outranking relations). In the next sections there are both main ideas explained [13].

### 2.3.1 Multi-Attribute utility theory

Multiple Attribute Utility Theory (MAUT) is a model of preference that assigns a utility value to each alternative. The utility is a real number that represents the preference of the considered alternative, and it is usually calculated as the sum of the marginal utilities in each criterion [14].

In MAUT, data is often provided as a decision matrix (see Table 2), where rows are the different alternatives and columns are the criteria.

Table 2: Decision matrix

	$g_1$	$g_2$	...	$g_n$
$a_1$	$g_1(a_1)$	$g_2(a_1)$	...	$g_n(a_1)$
$a_2$	$g_1(a_2)$	$g_2(a_2)$	...	$g_n(a_2)$
...	...	...		...
$a_m$	$g_1(a_m)$	$g_2(a_m)$	...	$g_n(a_m)$

There exist different models for aggregating the utility of each criterion, although it usually is the additive one. In the additive model,  $U$  is the additive combination of utility of the criteria [15].

$$U(a) = \sum_{i=1}^n U_i(g_i(a))$$

Equation 1

The main issue in utility-based methods is the determination of the marginal utility functions,  $U_i$ . These functions transform the criterion scale into numerical utility values. Once they are known, the global value for each alternative is calculated by aggregation,  $U(a)$ , and then the alternatives are ranked depending on its value to find the best alternative.

The UTilités Additives (UTA) method, the Analytic Hierarchy Process (AHP) method or the Measuring Attractiveness by a Categorical Based Evaluation Technique (MACBETH) approach are examples of MAUT models.

### 2.3.2 Outranking methods

Outranking methods perform pairwise comparisons of alternatives to determine the preferability of each alternative over the others for each particular criterion. Then, a concordance relation is established by aggregating the relative preferences [16].

An outranking relation is defined as a binary relation  $S$  on the set  $A$  of alternatives such that  $xSy$  if there are enough arguments to declare that  $x$  is at least as good as  $y$  and there are no essential reasons to refute that statement. More technically, two conditions must be fulfilled in order to be able to declare that an alternative  $x$  is at least as good as another alternative  $y$ :

- Concordance condition: a majority of criteria supports this statement.

- Non-discordance condition: the opposition of the other criteria is not too strong.

There exist different ways of applying these conditions and several methods are based in this principle, such as Elimination and Choice Expressing Reality (ELECTRE) or Preference Ranking Organization Method for Enrichment of Evaluations (PROMETHEE).





### 3 UTA method

This chapter presents the UTA method and its extension  $UTA^{GMS}$ , which is the one implemented in this thesis. It consists of an introduction to the general method (Section 3.1), a formalisation of the method (Section 3.2) and the description of the extension  $UTA^{GMS}$  (Section 3.3).

#### 3.1 Overview

UTA methods are regression-based approaches that assume the MAUT bases and adopt the preference disaggregation principle. Its methodology uses linear programming techniques to get the optimal additive utility functions that are consistent with the decision-maker preferences.

The disaggregation-aggregation paradigm goal is to analyse the behaviour of the Decision-Maker (DM) by an iterative and interactive procedure, as it is schematized in Figure 3. The main objective of it is to improve the knowledge the DM has about the decision problem and the preference information given [17].

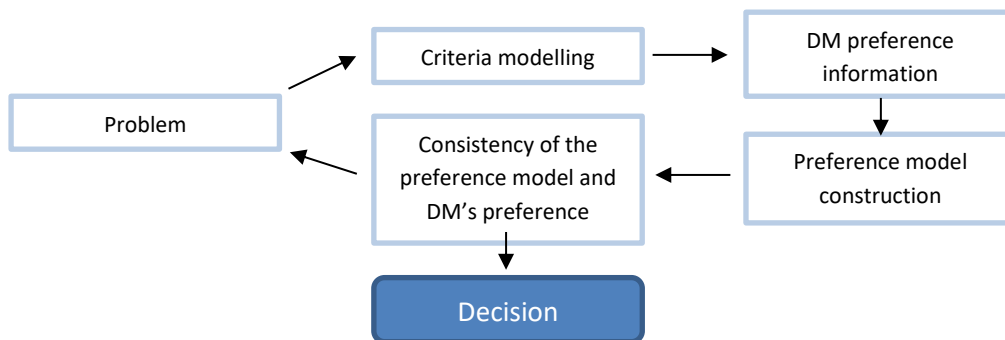


Figure 3: The disaggregation-aggregation approach

We are considering a decision problem in which there are a total of  $m$  alternatives in  $A$  set  $A = a_1, a_2, \dots, a_m$  evaluated in a family of  $n$  criteria in  $G$  set  $G = g_1, g_2, \dots, g_n$  [18]. Assuming a numerical scale on the criteria, it can be stated that the greater the evaluation of any alternative  $a$  in any criterion  $g, g(a)$ , the better this alternative is on that criterion. Every criterion can have its own numerical scale, that ranges from the worst to the best level:  $g_i \in [g_{i*}, g_i^*], \forall i = 1 \dots n$ , where  $g_{i*} < g_i^*$ . The set of criteria  $G$  should satisfy these consistency conditions:

- Exhaustivity: if two alternatives have the same evaluation on all criteria are considered indifferent.

- Monotonicity: while comparing two alternatives, if one of them has an improvement in at least one criterion from  $G$ , this should not deteriorate the comparison.
- Non-redundancy: if a criterion  $g_i$  is eliminated, the problem is not consistent.

Once this is set, it can be considered to be a multiple-criteria ranking problem. The goal of the DM is to rank the alternatives in  $A$  from the best to the worst taking in mind its own preferences. This ranking solution can be partial or complete, depending on the amount of information given and the way of using it. At the beginning of the procedure, the DM can provide preference information about the data, in a direct or indirect way [19].

- Direct preference information consists on specifying values of some parameters in the model. It is used in the traditional aggregation paradigm, where the aggregation model is first defined and then applied on the data to obtain the ranking of the different alternatives.
- Indirect preference information consists on specifying some examples of holistic judgements so a preference model is built. It is used in the disaggregation paradigm, where preferences on a subset of  $A$  called reference set  $A_R \subseteq A$  are known, and this information is used to create an aggregation model to be applied on set  $A$  to rank the alternatives from best to worst.

This direct or indirect information is essential to enrich the weak dominance relation that would result without it. A weak dominance relation is a partial preorder, where an alternative  $a$  is preferred to an alternative  $b$  if and only if  $g_i(a) \geq g_i(b)$  for all  $i \in G$  (with at least one inequality). Also,  $a$  is indifferent to  $b$  if and only if  $g_i(a) = g_i(b)$  for all  $i \in G$  and the alternative  $a$  is incomparable to  $b$  if none of the above are true. Specifically,  $a$  is incomparable to  $b$  if  $g_i(a) > g_i(b)$  for at least one criterion  $i \in G$  and also  $g_i(a) < g_i(b)$  for at least another criterion  $i \in G$ . As the usual relation obtained is incomparability, the weak dominance relation gives poor information. So, having preference information from the DM, either direct or indirect, helps in having a more complete solution. Usually, requesting indirect information from the DM is more common rather than direct, as it requires less knowledge in the optimisation model.

This being explained, the preference information that is used in this thesis is provided in the form of a set of pairwise comparisons of the reference alternatives  $A_R \subseteq A$ . It is also considered the criteria aggregation model in form of an unweighted additive value function  $U$  of the following form:

$$U(g) = \sum_{i=1}^n u_i(g_i)$$

subject to normalization constraints:

$$\begin{cases} \sum_{i=1}^n u_i(g_i^*) = 1 \\ u_i(g_{i*}) = 0 \quad \forall i = 1, 2, \dots, n \end{cases}$$

Equation 2

where  $u_i \in [0,1]$  are non-decreasing real valued functions (named utility functions) normalized between 0 and 1.

### 3.2 Development of the UTA method

As initially exposed, the UTA method follows the additive model and takes into account the preference conditions given by the DM. Being so, the value of the utility function of each alternative  $a \in A_R$  is written as:

$$U'(g(a)) = \sum_{i=1}^n u_i(g_i(a)) + \sigma(a) \quad \forall a \in A_R$$

Equation 3

Where  $\sigma(a)$  is a potential error over the alternative  $a$ .

The UTA method proposes the minimization of each  $\sigma(a)$ ,  $\forall a \in A_R$ , via linear programming [20]. To do so, the interval  $[g_{i*}, g_i^*]$  of each criterion is cut into  $(\alpha_i - 1)$  equals intervals, so the endpoints  $g_i^j$  are written as:

$$g_i^j = g_i + \frac{j-1}{\alpha_i-1} (g_i^* - g_{i*}) \quad \forall j = 1, 2, \dots, \alpha_i$$

Equation 4

The marginal value of an alternative  $a$ ,  $u_i(g_i(a))$ , is approximated by a linear interpolation in order to get a piecewise linear form. For  $g_i(a) \in [g_i^j, g_i^{j+1}]$ ,

$$u_i(g_i(a)) = u_i(g_i^j) + \frac{g_i(a) - g_i^j}{g_i^{j+1} - g_i^j} [u_i(g_i^{j+1}) - u_i(g_i^j)]$$

Equation 5

Next step is to sort the set  $A_R$  from the best ( $a_1$ ) to the worst ( $a_m$ ) alternative, so it holds that for each pair of consecutive alternatives ( $a_k, a_{k+1}$ ) either  $a_k$  is preferred to  $a_{k+1}$  ( $a_k \succ a_{k+1}$ ) or  $a_k$  is indifferent to  $a_{k+1}$  ( $a_k \sim a_{k+1}$ ). This being said, it is clear that:

$$\Delta(a_k, a_{k+1}) = U'(g(a_k)) - U'(g(a_{k+1})) \implies \begin{cases} \Delta(a_k, a_{k+1}) \geq \delta & \text{iff } a_k > a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0 & \text{iff } a_k \sim a_{k+1} \end{cases}$$

Equation 6

where  $\delta$  is a real, small positive number that is defined in the beginning.

With the objective of avoiding that  $u_i(g_i^j) = u_i(g_i^{j+1})$  happen, even when  $g_i^{j+1} > g_i^j$ , a new constraint is added, where  $s_i$  is a positive indifference threshold defined on each criterion.

$$u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \quad \forall j = 1, 2, \dots, \alpha_i - 1, \forall i = 1, 2, \dots, n$$

Equation 7

Finally, the optimal marginal value functions are estimated by the following linear program, where the sum of all alternatives associated errors  $\sigma(a)$  is minimized:

$$F^* = \min \left\{ F = \sum_{a \in A_R} \sigma(a) \left| \begin{array}{l} \Delta(a_k, a_{k+1}) \geq \delta \quad \text{iff } a_k > a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0 \quad \text{iff } a_k \sim a_{k+1} \\ u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \\ \sum_{i=1}^n u_i(g_i^*) = 1 \\ u_i(g_{i^*}) = 0, u_i(g_i^j) \geq 0, \sigma(a) \geq 0, \forall a \in A_R, \forall i, j \end{array} \right. \right\}$$

Equation 8

When the optimal value  $F^*$  of the objective function is greater than zero ( $F^* > 0$ ), then there is no value function compatible with the reference preorder on  $A_R$ . In such a case, there are different actions to consider [21]:

- Increase the number of linear pieces for one or several marginal value functions, as this could help finding an additive value function compatible with the reference preorder on  $A_R$ .
- Take another look at the reference preorder on  $A_R$ , because a simple modification could lead to find an additive value function compatible with the new preorder.
- Search over a relaxed domain such as  $F \leq F^* + \eta$ , possibly leading to an additive value function giving a preorder on  $A_R$  sufficiently close to the reference preorder.

This basic UTA method has a lot of variants and extensions, and this paper is focused in UTA<sup>GMS</sup> method, which is the topic in the next section.

To simplify notation, in this thesis it will sometimes be written  $U(a)$  instead of  $U(g_1(a), \dots, g_n(a))$  and  $u_i(a)$  instead of  $u_i(g_i(a))$  for any alternative  $a \in A$ .

### 3.3 UTA<sup>GMS</sup>

#### 3.3.1 Introduction to UTA<sup>GMS</sup>

UTA<sup>GMS</sup> is an ordinal regression method that generalizes UTA method mainly by considering the set of all value functions compatible with the indirect preference information instead of choosing a single value function from this set. UTA<sup>GMS</sup> also differs by using general non-decreasing marginal value functions rather than piecewise linear only [22].

The preference information given by the DM is a partial preorder on  $A_R$  in form of pairwise comparisons of reference alternatives from  $A_R \subseteq A$ . When a value function is able to restore this partial preorder is called *compatible*, and each compatible value function induces a ranking on the whole set  $A$ . For two alternatives  $a, b \in A$ , a compatible value function  $U$  ranks  $a$  and  $b$  in one of the following ways:

- $U(a) > U(b)$ , then  $a$  is preferred to  $b$ .
- $U(a) < U(b)$ , then  $b$  is preferred to  $a$ .
- $U(a) = U(b)$ , then  $a$  is indifferent to  $b$ .

In the solving of the problem,  $U$  can have multiple solutions compatible with preference information. In such a case, it makes no sense to compare  $a$  and  $b$  by one simple compatible value function, it is needed to know whether *all* compatible value functions rank them in the same way or there is *at least one* compatible value function that ranks them differently. This is the main approach of UTA<sup>GMS</sup> method.

UTA<sup>GMS</sup> method provides two different relations of the alternatives  $A$ , a *necessary* weak preference relation and a *possible* weak preference relation. For any pair of alternatives  $a, b \in A$ :

- *Necessary* relation:  $a$  is ranked at least as good as  $b$  if and only if,  $U(a) \geq U(b)$  for *all* value functions compatible with the preference information. It is expressed as  $a \succeq^N b$ .
- *Possible* relation:  $a$  is ranked at least as good as  $b$  if and only if,  $U(a) \geq U(b)$  for *at least one* value function compatible with the preference information. It is expressed as  $a \succeq^P b$ .

Putting words into formal expressions, a general additive compatible value function is an additive value function  $U(a) = \sum_{i=1}^n u_i(a)$  that satisfies the following set of constraints:

$$\left. \begin{aligned}
& U(c) > U(d) \Leftrightarrow c > d \\
& U(c) = U(d) \Leftrightarrow c \sim d \\
& u_i(g_i(a_{\tau_i(j)})) - u_i(g_i(a_{\tau_i(j-1)})) \geq 0, \quad i = 1, \dots, n \quad j = 2, \dots, m \\
& u_i(g_i(a_{\tau_i(1)})) \geq 0, \quad u_i(g_i(a_{\tau_i(m)})) \leq u_i(g_i^*), \quad i = 1, \dots, n \\
& u_i(g_{i^*}) = 0, \quad i = 1, \dots, n \\
& \sum_{i=1}^n u_i(g_i^*) = 1
\end{aligned} \right\} E^{A_R}$$

Equation 9

Where  $\tau_i$  is the permutation on the set of indices of alternatives from  $A_R$  that reorders them according to the increasing evaluation on criterion  $g_i$ :

$$g_i(a_{\tau_i(1)}) \leq g_i(a_{\tau_i(2)}) \leq \dots \leq g_i(a_{\tau_i(m)})$$

Equation 10

### 3.3.2 Properties of the relations $\succsim^N$ and $\succsim^P$

Table 3: Properties of the binary relations  $\succsim^N$  and  $\succsim^P$  [22]

Property	Proof
$\succsim^P \supseteq \succsim^N$	If $U(a) \geq U(b)$ for all compatible value functions, then there is at least one compatible value function $U'$ such that $U'(a) \geq U'(b)$
For all $a, b \in A$ , $a \succsim^N b$ or $b \succsim^P a$	Being $\mathfrak{U}$ the set of value functions compatible with $\succsim$ , for all $a, b \in A$ , $U(a) \geq U(b)$ for all $U \in \mathfrak{U}$ or $\exists U \in \mathfrak{U}$ such that $U(a) > U(b)$
$\text{not}(a \succsim^N b) \Rightarrow b \succsim^P a$	Since $a \succsim^N b$ or $b \succsim^P a$ , if $\text{not}(a \succsim^N b)$ , then $b \succsim^P a$
$\text{not}(a \succsim^P b) \Rightarrow b \succsim^N a$	Since $a \succsim^P b$ or $b \succsim^N a$ , if $\text{not}(a \succsim^P b)$ , then $b \succsim^N a$
$\succsim^N$ is a partial preorder (reflexive and transitive)	For all $a \in A$ , $U(a) = U(a)$ .  For $a, b, c \in A$ , if we have $a \succsim^N b$ and $b \succsim^N c$ means that $U(a) \geq U(b)$ and $U(b) \geq U(c)$ , which implies that also $U(a) \geq U(c)$
$\succsim^P$ is strongly complete and negatively transitive	For all $a, b \in A$ , it holds $U(a) \geq U(b)$ or $U(b) \geq U(a)$ , so also $a \succsim^P b$ or $b \succsim^P a$ .  $\text{not}(a \succsim^P b)$ means that there does not exist any compatible value function that $U(a) \geq U(b)$ , $\text{not}(b \succsim^P c)$ means that there does not exist any compatible value function that $U(b) \geq U(c)$ . Therefore, there does not exist any compatible value function that $U(a) \geq U(c)$ , which means that $\text{not}(a \succsim^P c)$

From the properties included in Table 3 it is clear that from the necessary weak preference relation  $\succeq^N$  is possible to obtain:

$$\left. \begin{array}{l} \text{Preference: } a \succ^N b \Leftrightarrow a \succeq^N b \text{ and } \text{not}(b \succeq^N a) \\ \text{Indifference: } a \sim^N b \Leftrightarrow a \succeq^N b \text{ and } b \succeq^N a \\ \text{Incomparability: } a ?^N b \Leftrightarrow \text{not}(a \succeq^N b) \text{ and } \text{not}(b \succeq^N a) \end{array} \right\} \text{ Necessary ranking}$$

Also, from the possible weak preference relation  $\succeq^P$  is possible to obtain:

$$\left. \begin{array}{l} \text{Preference: } a \succ^P b \Leftrightarrow a \succeq^P b \text{ and } \text{not}(b \succeq^P a) \\ \text{Indifference: } a \sim^P b \Leftrightarrow a \succeq^P b \text{ and } b \succeq^P a \end{array} \right\} \text{ Possible ranking}$$

### 3.3.3 Implementation of UTA<sup>GMS</sup>

The objective of this stage is to evaluate each pair of alternatives in the decision problem. The model will, for each pair  $a, b \in A$ , look for the maximum difference between  $U(a)$  and  $U(b)$  and the other way round. To do so, and also be compatible with all preference information, the linear program will include information from  $a, b$  and also from the reference alternatives in  $A_R$ , maximizing or minimizing this mentioned difference [22].

First of all, the information of the alternatives  $a, b \in A$  and the reference alternatives in  $A_R$  is combined ( $A_R \cup \{a, b\}$ ) into the same table and ordered in an increasing evaluation on criterion  $g_i$ :

$$g_i(a_{\pi_i(1)}) \leq g_i(a_{\pi_i(2)}) \leq \dots \leq g_i(a_{\pi_i(w)})$$

Equation 11

Where  $\pi_i$  is the permutation of the indices of alternatives from the set, and  $w$  is a value that can change its value, due to the possibility that an alternative that is being evaluated is also part of the reference information  $A_R$ .

- If  $A_R \cap \{a, b\} = \emptyset$ , then  $w = m + 2$
- If  $A_R \cap \{a, b\} = \{a\}$  or  $A_R \cap \{a, b\} = \{b\}$ , then  $w = m + 1$
- If  $A_R \cap \{a, b\} = \{a, b\}$ , then  $w = m$

This way, the characteristic points of  $u_i$  can be redefined as:

$$g_i^0 = g_{i*}, \quad g_i^j = g_i(a_{\pi_i(j)}) \text{ for } j = 1, 2, \dots, w, \quad g_i^{w+1} = g_i^*$$

Where, just to remember, every criterion can have its own numerical scale that ranges from the worst to the best level:  $g_i \in [g_{i*}, g_i^*], \forall i = 1 \dots n$ , where  $g_{i*} < g_i^*$ .

Once these values are actualised, the set  $E(a, b)$  of ordinal regression constraints is the following:

$$\left. \begin{aligned} & \left. \begin{aligned} & U(c) \geq U(d) + \varepsilon \Leftrightarrow c \succ d \\ & U(c) = U(d) \Leftrightarrow c \sim d \end{aligned} \right\} \text{ for all } c, d \in A_R \\ & u_i(g_i^j) - u_i(g_i^{j-1}) \geq 0, \quad i = 1, \dots, n \quad j = 2, \dots, w + 1 \\ & u_i(g_i^0) = 0, \quad i = 1, \dots, n \\ & \sum_{i=1}^n u_i(g_i^{w+1}) = 1 \end{aligned} \right\} E(a, b) \end{aligned}$$

Equation 12

where  $\varepsilon$  is an arbitrarily small positive value.

This set of constraints changes depending on the pair of alternatives being evaluated, because some of the characteristic points of the marginal value function may change its value. Also, it is known that  $E(a, b) = E(b, a)$  for all  $a, b \in A$ .

The objective function of this model is the maximization or minimization of the difference  $U(a) - U(b)$ . Both optimizations are useful and give an interesting result, which will be analysed further in this section. Then, for all  $a, b \in A$ , we have that:

$$d(a, b) = \min\{U(a) - U(b)\} \quad \text{subject to set } E(a, b) \text{ of constraints}$$

$$D(a, b) = \max\{U(a) - U(b)\} \quad \text{subject to set } E(a, b) \text{ of constraints}$$

Note also that given these, other relations also hold:

$$\left. \begin{aligned} & d(a, b) \geq 0 \Leftrightarrow a \succeq^N b \\ & D(a, b) \geq 0 \Leftrightarrow a \succeq^P b \end{aligned} \right\}$$

$$\left. \begin{aligned} & d(a, b) \geq 0 \Leftrightarrow D(b, a) \leq 0 \\ & D(a, b) \geq 0 \Leftrightarrow d(b, a) \leq 0 \\ & d(a, b) = 0 \Leftrightarrow D(b, a) = 0 \end{aligned} \right\}$$

The latter relations can be explained by the following equalities:

$$d(a, b) = \min_{s.t. E(a,b)}\{U(a) - U(b)\} = -\max_{s.t. E(b,a)}\{U(b) - U(a)\} = -D(b, a)$$

Equation 13

According to this, it is clear that both relations  $\succeq^N$  and  $\succeq^P$  can be calculated by both  $d(a, b)$  or  $D(a, b)$ . So, in order to simplify the model, in this thesis it has been only used  $d(a, b)$ . Table 4 and Table 5 show how are these concepts related, first for the necessary ranking and then for the possible ranking.



Table 4: Necessary ranking computed in terms of  $d(a,b)$

		$b \succsim^N a$		$not(b \succsim^N a)$
		$d(b,a) > 0$	$d(b,a) = 0$	$d(b,a) < 0$
$a \succsim^N b$	$d(a,b) > 0$	---	---	$a \succ^N b$
	$d(a,b) = 0$	---	$a \sim^N b$	$a \succ^N b$
$not(a \succsim^N b)$	$d(a,b) < 0$	$b \succ^N a$	$b \succ^N a$	$a?b$

Table 5: Possible ranking computed in terms of  $d(a,b)$

		$b \succsim^P a$		$not(b \succsim^P a)$
		$d(b,a) > 0$	$d(b,a) = 0$	$d(b,a) < 0$
$a \succsim^P b$	$d(a,b) > 0$	---	---	$a \succ^P b$
	$d(a,b) = 0$	---	$a \sim^P b$	$a \sim^P b$
$not(a \succsim^P b)$	$d(a,b) < 0$	$b \succ^P a$	$a \sim^P b$	$a \sim^P b$

There exist the same exact types of tables for  $D(a,b)$ , but they are skipped as its totally indifferent in terms of results to use one relation or the other.

These rankings are presented to the DM as end results of the  $UTA^{GMS}$  method at the current stage of iteration. At this moment, the DM would be able to look at them and add or modify some preference information, to do the next iteration of the method. As it can be seen, this is an iterative and interactive method, where the information provided by the DM can be added in parts, or even change its mind and modify some information that does not think anymore after being able to see it graphically. The iterations end when the DM is confident enough about the ranking.



## 4 Implementation of UTA<sup>GMS</sup> in MATLAB

This chapter presents the implementation of the UTA<sup>GMS</sup> method, which is the one that has been just described. It consists of a brief description of the software used (Section 4.1) and the implementation of the model (Section 4.2).

### 4.1 Software description

During the learning and comprehension of the method used in this thesis, multiple examples and cases have been developed to make sure that the concepts were clear. Starting with the UTA method, the software used to test the specific ideas of the model is IBM ILOG CPLEX Optimization Studio (Section 4.1.1). Later, for the UTA<sup>GMS</sup> method, the software MATLAB in combination with CPLEX will be used and explained (Section 4.1.2).

#### 4.1.1 IBM ILOG CPLEX Optimization Studio

The IBM ILOG CPLEX Optimization Studio software uses decision optimization technology to optimize business decisions, develop and deploy optimizations models quickly and create real-world applications that can significantly improve business outcomes. It consists of an integrated development environment (IDE), a mathematical optimization engine (CPLEX), a constraint programming engine (CP), and a set of APIs (Python, Java, C, etc.) for modelling, solving and embedding optimization solutions.

At the start a new OPL project, two main windows are created: Model and Data. It should also be noted that the data of the project is often extracted from an Excel file.

In the Model window (.mod) is where the optimization problem is written. The code in it is divided into three different parts:

- Definition of the data: the data available for the model and its dimensions are introduced in here. Depending on the type of numbers the variable contains, can be classified as integer (*int*) or as float (*float*). If the data is exported from an external source (Excel, for example) the information is loaded from the Data window, and the way of indicating it is by "...". An example of this can be found in Figure 4.

```

int n=...;
int m=...;
float delta=...;
float s=...;
int endpoints=...;
float coef_u [1..m][1..12]=...;

```

Figure 4: Example of data definition

- Definition of the decision variables: decision variables are variables in the model that will take a value during or at the end of the optimization, but do not have it in the beginning. They are defined by adding “*dvar*” in front of the name and dimensions of the variable. As before, it is needed to specify if the variable is an integer or float, as it is shown in Figure 5.

```

dvar float+ f;
dvar float+ sigmaa [1..m];
dvar float+ u_final [1..m];
dvar float+ uij [1..endpoints][1..n];
dvar float+ uga [1..m][1..n];

```

Figure 5: Example of decision variables definition

- Linear program: in this part the full model is written. Starting with the objective function (“*minimize*” or “*maximize*”) and following with all the constraints, the model is built. In Figure 6 there is an example of UTA method definition in IBM CPLEX software.

```

minimize F;
subject to {
  F==sum(a in 1..m) sigmaa[a];

  forall (k in 1..4) {
    u_final[k] - u_final[k+1] >= delta;
  }

  forall (i in 1..n) {
    forall (j in 1..2) {
      uij [j+1][i] - uij [j][i] >= s;
    }
  }

  sum(i in 1..n) uij [3][i] == 1;

  forall (i in 1..n) {
    uij [1][i] == 0;
  }

  forall (a in 1..m) {
    u_final [a] == sum(i in 1..n) (uga [a][i]) + sigmaa [a];
  }

  forall (a in 1..m) {
    uga [a][1] == coef_u [a][1] * uij [1][1] + coef_u [a][2] * uij [2][1] + coef_u [a][3] * uij [3][1];
    uga [a][2] == coef_u [a][4] * uij [1][2] + coef_u [a][5] * uij [2][2] + coef_u [a][6] * uij [3][2];
    uga [a][3] == coef_u [a][7] * uij [1][3] + coef_u [a][8] * uij [2][3] + coef_u [a][9] * uij [3][3];
    uga [a][4] == coef_u [a][10] * uij [1][4] + coef_u [a][11] * uij [2][4] + coef_u [a][12] * uij [3][4];
  }
}

```

Figure 6: Example of a UTA method linear program

In the Data window (.dat) is where the data is imported and exported. In the beginning of the code it is needed to define the file where the data is stored. This is done by the command “*SheetConnection*” and it allows to read and write on the specific file (see Figure 7). The code is divided into two different parts:

- Reading file: to import the data stored in the external file, the command “*SheetRead*” is used, and then the exact location of the corresponding data is indicated.
- Writing file: to export the data and the solution to the external file, the command “*SheetWrite*” is used, and then the exact location desired for storing the corresponding data is indicated.

```
SheetConnection excel("Scenario1.xlsx");

n from SheetRead(excel,"Scenario1\F12");
m from SheetRead(excel,"Scenario1\F13");
delta from SheetRead(excel,"Scenario1\C12");
s from SheetRead(excel,"Scenario1\C13");
endpoints from SheetRead(excel,"Scenario1\C14");
coef_u from SheetRead(excel,"Scenario1\C18:N22");

F to SheetWrite(excel,"Scenario1\Q3");
sigmaa to SheetWrite(excel,"Scenario1\P3:P7");
u_final to SheetWrite(excel,"Scenario1\U11:U15");
uij to SheetWrite(excel,"Scenario1\J11:M13");
uga to SheetWrite(excel,"Scenario1\P11:S15");
```

Figure 7: Example of importing and exporting data

Therefore, the solution of this optimization can be seen both in the IBM program itself (*Problem browser* window) and in the Excel file where it was exported.

#### 4.1.2 MATLAB - CPLEX

As explained above, the IBM ILOG CPLEX Optimization Studio program is a very powerful tool for mathematical optimization, providing high-performance component libraries for large and difficult problems. In the case of the UTA<sup>GMS</sup> method, however, many optimizations have to be done with different data each time. MATLAB is a program widely used for mathematical computing, providing a high-performance interactive environment for computation, visualization and programming. Combining both MATLAB and CPLEX allows taking advantage of each best feature.

To integrate CPLEX on MATLAB in Windows, the following steps must be completed [23]:

- Open MATLAB
- Execute command *pathtool*

- Click on *Add folder* button and add the path “C:\Program Files\IBM\ILOG\Cplex\_Studio\_Community1263\cplex\matlab\x64\_win64”, then save and close.

After these simple steps, CPLEX is linked to MATLAB and its functions can now be used.

IBM ILOG CPLEX Optimizers provides a tool for solving optimization, or mathematical programming, problems. The most basic mathematical programming problem is a Linear Programming (LP) problem. The basic form of an LP problem is included in Table 6.

Table 6: LP problem

Minimize	$f * x$
Subject to	$A_{eq} * x = b_{eq}$
	$A_{ineq} * x \leq b_{ineq}$
With these bounds	$l \leq x \leq u$

where  $A_{eq}$  and  $A_{ineq}$  are matrices,  $f, b_{eq}, b_{ineq}, l$  and  $u$  are vectors.

In Table 7 there are the input data for a basic LP problem.

Table 7: Inputs for an LP problem

Objective function coefficients	$f$
Constraint coefficients	$A_{eq}$
	$A_{ineq}$
Righthand sides	$b_{eq}$
	$b_{ineq}$
Upper and lower bounds	$u$
	$l$

The optimal solution that CPLEX computes and returns is the one in Table 8.

Table 8: Output for an LP problem

Variables	$x$
-----------	-----

In order to execute this optimization for a basic LP problem, the function call would be the following one:

$$x = cplexlp(f, A_{ineq}, b_{ineq}, A_{eq}, b_{eq}, l, u)$$

There are other return values that can be useful, as is the value of the objective function at the solution  $x$  ( $fval$ ) or an integer identifying the reason the optimization algorithm terminated ( $exitflag$ ). If  $exitflag > 0$ , then  $x$  is a solution; otherwise,  $x$  is the value of the optimization when it terminated prematurely. The meaning of the value  $exitflag^4$  is described in Table 9:

Table 9: Meaning of  $exitflag$  values

exitflag	Description
6	Non-optimal Solution available
5	Solution with numerical issues
1	Function converged to a solution $x$
0	Number of iterations exceeded options
-1	Aborted
-2	No feasible point was found
-3	Problem is unbounded
-4	NaN value was encountered during execution of the algorithm
-5	Both primal and dual problems are infeasible
-7	Search direction became too small, no further progress could be made
-8	Problem is infeasible or unbounded
-9	Limit reached

The function call after adding these outputs is:

$$[x, fval, exitflag] = cplexlp(f, A_{ineq}, b_{ineq}, A_{eq}, b_{eq}, l, u)$$

## 4.2 Implementation of the model

The UTA<sup>GMS</sup> algorithm for MATLAB-CPLEX is as follows. The code is divided into three sections: importing the data, checking that there is at least one value function compatible with the reference information introduced, and the UTA<sup>GMS</sup> model itself.

- Importing the data for the model only requires having the performances entered in an Excel file in the form of a table, where the rows are the different alternatives and the columns are the criteria (as in Table 2). The command used to read this information is *readtable* and requires a path to the exact part of the document where the data is stored. Aside from reading the values, the real name of the alternatives can be also read.

---

<sup>4</sup> This documentation is found online in an IBM Corporation webpage: <https://perso.ensta-paris.fr/~diam/ro/online/cplex/cplex1271/refmatlabcpex/html/cplexmiqcp-m.html>

The reference information from the DM is entered manually, so that the reference alternatives  $a_k \in A_R$  and their corresponding preference ranking are indicated in the form of lists. Also, the threshold  $\varepsilon$  is defined. An example is shown in Figure 8.

```
% IMPORTING DATA

Initial_table = readtable('UTA_GMS.xlsx','Sheet','Case_Study','Range','D3:H28');
Names_candidates = readtable('UTA_GMS.xlsx','Sheet','Case_Study','Range','C3:C28');

Ref = [2,13,24,23,18];
InitialPref = [1,2,3,4,5]';

epsilon = 0.01;
```

Figure 8: Example of importing data

- Checking if there exists at least one value function compatible with the reference information is to run the UTA method with only the data of the reference alternatives. The procedure is described in section 3.3.1: Introduction to UTAGMS. The LP problem to put into code is the following one:

$$F^* = \min \left\{ F = \sum_{a \in A_R} \sigma(a) \left| \begin{array}{l} \Delta(a_k, a_{k+1}) \geq \varepsilon \quad \text{iff } a_k > a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0 \quad \text{iff } a_k \sim a_{k+1} \\ u_i(g_i(a_{\tau_i(j)})) - u_i(g_i(a_{\tau_i(j-1)})) \geq 0, \quad i = 1, \dots, n \quad j = 2, \dots, m \\ \sum_{i=1}^n u_i(g_i^*) = 1 \\ u_i(g_i(a_{\tau_i(1)})) \geq 0, \quad u_i(g_i(a_{\tau_i(m)})) \leq u_i(g_i^*), \forall i \\ u_i(g_i^*) = 0, \sigma(a) \geq 0, \quad \forall a \in A_R, \forall i \end{array} \right. \right.$$

where  $\tau_i$  is the permutation on the set of indices of alternatives from  $A_R$  that reorders them according to the increasing evaluation on criterion  $g_i$ :

$$g_i(a_{\tau_i(1)}) \leq g_i(a_{\tau_i(2)}) \leq \dots \leq g_i(a_{\tau_i(m)})$$

The output  $x$  of the problem is a vector of  $(n + 1) * m$  rows containing the marginal values of the alternatives and has the following form shown in Table 10.



Table 10: Variables

		<b>x</b>
<i>i</i> = 1	<i>j</i> = 1	$u_1(g_1(a_{\tau_1(1)}))$
	...	...
	<i>j</i> = <i>m</i>	$u_1(g_1(a_{\tau_1(m)}))$
<i>i</i> = 2	<i>j</i> = 1	$u_2(g_2(a_{\tau_2(1)}))$
	...	...
	<i>j</i> = <i>m</i>	$u_2(g_2(a_{\tau_2(m)}))$
...	...	...
<i>i</i> = <i>n</i>	<i>j</i> = 1	$u_n(g_n(a_{\tau_n(1)}))$
	...	...
	<i>j</i> = <i>m</i>	$u_n(g_n(a_{\tau_n(m)}))$
$\sigma(a_k)$	<i>k</i> = 1	$\sigma(a_1)$
	...	...
	<i>k</i> = <i>m</i>	$\sigma(a_m)$

The objective function  $F = fval = f * x$  of the problem is a single value to be minimized that comes from multiplying the vector  $x$  with the vector  $f$ , which need to have the same size. Being so, the vector  $f$  has  $(n + 1) * m$  values containing the coefficients of the variables  $x$  (see Table 11).

Table 11: Objective function coefficients

	$u_i(g_i(a_{\tau_i(j)}))$ for $i = 1, \dots, n$ and $j = 1, \dots, m$			$\sigma(a_k)$ for $k = 1, \dots, m$		
<b>f</b>	0	...	0	1	...	1

Starting with the creation of the constraints, the first thing to do is to create a matrix that contains each criterion  $g_i$  in an increasing evaluation as shown in Table 12:

Table 12: Ordered performances

	$g_1$	$g_2$	...	$g_n$
<i>j</i> = 1	$g_1(a_{\tau_1(1)})$	$g_2(a_{\tau_2(1)})$	...	$g_n(a_{\tau_n(1)})$
<i>j</i> = 2	$g_1(a_{\tau_1(2)})$	$g_2(a_{\tau_2(2)})$	...	$g_n(a_{\tau_n(2)})$
...	...	...		...
<i>j</i> = <i>m</i>	$g_1(a_{\tau_1(m)})$	$g_2(a_{\tau_2(m)})$	...	$g_n(a_{\tau_n(m)})$

The next matrix to be created (Table 13) is of the coefficients that relate each alternative to the position in the order matrix just performed. Being  $a_1, a_2, \dots, a_m \in A_R$  the reference alternatives, the matrix has the following form:

Table 13: Coefficients table

		$a_1$	$a_2$	...	$a_m$
$i = 1$	$j = 1$	[0,1]	[0,1]	...	[0,1]
	...	...	...		...
	$j = m$	[0,1]	[0,1]	...	[0,1]
$i = 2$	$j = 1$	[0,1]	[0,1]	...	[0,1]
	...	...	...		...
	$j = m$	[0,1]	[0,1]	...	[0,1]
...	...	...	...	...	...
$i = n$	$j = 1$	[0,1]	[0,1]	...	[0,1]
	...	...	...		...
	$j = m$	[0,1]	[0,1]	...	[0,1]

This matrix is very useful, as it is used in the creation of some of the constraints.

For defining the equality constraints ( $A_{eq} * x = b_{eq}$ ) two matrixes need to be created,  $A_{eq}$  and  $b_{eq}$ , which refer to the constraints  $\sum_{i=1}^n u_i(g_i^*) = 1$  (first row in Table 14 and Table 15) and  $\Delta(a_k, a_{k+1}) = 0$  iff  $a_k \sim a_{k+1}$  (from second row to end of matrix, as many rows as constraints). Thus, these matrixes have the following form:

Table 14: Matrix of equality constraints coefficients

$A_{eq}$			$i = 2$				...	$i = n$			$\sigma(a_k)$		
$i = 1$			$i = 2$				$i = n$			$\sigma(a_k)$			
$j = 1$	...	$j = m$	$j = 1$	...	$j = m$	...	$j = 1$	...	$j = m$	$k = 1$	...	$k = m$	
0	0	1	0	0	1	...	0	0	1	0	0	0	
[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	...	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	
[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	...	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	

Table 15: Matrix that completes the equality constraints

	$b_{eq}$
$\sum_{i=1}^n u_i(g_i^*) = 1$	1
$\Delta(a_k, a_{k+1}) = 0$ iff $a_k \sim a_{k+1}$	0
	0

In a similar way, for defining the inequality constraints ( $A_{ineq} * x \leq b_{ineq}$ ) two other matrixes need to be created,  $A_{ineq}$  and  $b_{ineq}$ , which refer to the constraints  $\Delta(a_k, a_{k+1}) \geq \varepsilon$  iff  $a_k > a_{k+1}$  (firsts rows in X and Y, as many as constraints) and  $u_i(g_i(a_{\tau_i(j)})) - u_i(g_i(a_{\tau_i(j-1)})) \geq 0$ ,  $i = 1, \dots, n$   $j = 2, \dots, m$  (rest of the matrix, there are  $(m - 1) * n$  rows). Thus, these matrixes have the following form:

Table 16: Matrix of inequality constraints coefficients

$A_{ineq}^*$												
$i = 1$			$i = 2$			...	$i = n$			$\sigma(a_k)$		
$j = 1$	...	$j = m$	$j = 1$	...	$j = m$	...	$j = 1$	...	$j = m$	$k = 1$	...	$k = m$
[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	...	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]
[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	...	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]	[0,1]

The matrix  $A_{ineq}^*$  shown in Table 16 is incomplete, as there are constraints missing. For each criterion,  $(m-1)$  rows should be added with all values to 0 except for pairs of (1, -1) in consecutive cells. As an example, for any criterion and for  $m=7$ , matrix in Table 17 extended and filled with empty values to 0 should be added to  $A_{ineq}^*$  to create  $A_{ineq}$  together with the other criterion matrixes.

Table 17: Example of auxiliary matrix

$\forall i$						
$j = 1$	$j = 2$	$j = 3$	$j = 4$	$j = 5$	$j = 6$	$j = 7$
1	-1					
	1	-1				
		1	-1			
			1	-1		
				1	-1	
					1	-1

Table 18: Matrix that completes the inequality constraints

	$b_{ineq}$
$\Delta(a_k, a_{k+1}) \geq \varepsilon$ iff $a_k > a_{k+1}$	$\varepsilon$
	$\varepsilon$
$u_i(g_i(a_{\tau_i(j)})) - u_i(g_i(a_{\tau_i(j-1)})) \geq 0, i = 1, \dots, n \quad j = 2, \dots, m$	0

where  $\varepsilon = 0.01$  in Table 18 is defined in the beginning of the model.

With all the constraints being defined, it is needed to define the lower and upper bounds of the variables  $x$  (following the constraint  $l \leq x \leq u$ ). Variables  $x$  are divided into two types, first part are  $u_i(g_i(a_{\tau_i(j)})) \in [0, \infty]$  and second part are  $\sigma(a_1) \in [0, \infty]$ . Thus, these vectors have the following form (see Table 19 and Table 20):

Table 19: Lower bounds of  $x$

	$u_i(g_i(a_{\tau_i(j)}))$ for $i = 1, \dots, n$ and $j = 1, \dots, m$			$\sigma(a_k)$ for $k = 1, \dots, m$		
$l$	0	...	0	0	...	0

Table 20: Upper bounds of  $x$

	$u_i(g_i(a_{\tau_i(j)}))$ for $i = 1, \dots, n$ and $j = 1, \dots, m$			$\sigma(a_k)$ for $k = 1, \dots, m$		
$u$	$\infty$	...	$\infty$	$\infty$	...	$\infty$

After executing this LP model, the output  $fval$  delivers the information of the objective function. If  $fval = 0$  then the model is valid for continuing the execution of the code, as it means that there exists at least one value function that is compatible with the reference information delivered by the DM.

- The UTA<sup>GMS</sup> method is based on a similar approach as the one just explained. For each pair of variables, the code minimizes an objective function subject to some constraints which are not much different than the ones developed. The procedure is formally explained in section 3.3.3: Implementation of UTAGMS. The LP problem to put into code is the following one.

For each pair of alternatives  $a, b \in A$ :

$$d(a, b) = \min\{U(a) - U(b)\} \quad \text{subject to set } E(a, b) \text{ of constraints}$$

$$\left. \begin{array}{l} U(c) \geq U(d) + \varepsilon \Leftrightarrow c \succ d \\ U(c) = U(d) \Leftrightarrow c \sim d \\ u_i(g_i^j) - u_i(g_i^{j-1}) \geq 0, \quad i = 1, \dots, n \quad j = 2, \dots, w + 1 \\ u_i(g_i^0) = 0, \quad i = 1, \dots, n \\ \sum_{i=1}^n u_i(g_i^{w+1}) = 1 \end{array} \right\} E(a, b)$$

where  $\varepsilon$  is an arbitrarily small positive value.

So, in this case,  $fval = d(a, b)$ , and then the command used is:

$$[x, d(a, b), exitflag] = cplexlp(f, A_{ineq}, b_{ineq}, A_{eq}, b_{eq}, l, u)$$

For each pair of alternatives, the matrix of data is different, so the constraints are also distinct, and so are the input matrixes  $f, A_{ineq}, b_{ineq}, A_{eq}, b_{eq}, l, u$ . The code implemented in MATLAB (see Appendix A: Some parts of the Matlab code) builds the model for every combination and reaches a solution that considers the optimization of  $d(a, b)$  and also of  $d(b, a)$ , so by the information in Table 4: Necessary ranking computed in terms of  $d(a, b)$  the relation between

both alternatives is known. The model writes this relation into a greater matrix ( $m \times m$ ) until it is totally completed. This code is shown in Figure 9.

```

if or(and(d_a_b>0, d_b_a<0), and(d_a_b==0, d_b_a<0))
    Necessary_Ranking(candidate_a,candidate_b) = 1; %a>b
end
if and(d_a_b==0, d_b_a==0)
    Necessary_Ranking(candidate_a,candidate_b) = 2; %a=b
end
if or(and(d_a_b<0, d_b_a>0), and(d_a_b<0, d_b_a==0))
    Necessary_Ranking(candidate_a,candidate_b) = 3; %a<b
end
if and(d_a_b<0, d_b_a<0)
    Necessary_Ranking(candidate_a,candidate_b) = 4; %a?b
end

```

Figure 9: Building of Necessary Ranking matrix

where  $1 = \succ$ ,  $2 = \sim$ ,  $3 = \preccurlyeq$  and  $4 = ?$  are the preference relations.

Table 21: Partial preorder

	$a_1$	$a_2$	...	$a_m$
$a_1$	$\sim$	$\succ, \sim, \preccurlyeq, ?$	...	$\succ, \sim, \preccurlyeq, ?$
$a_2$	$\succ, \sim, \preccurlyeq, ?$	$\sim$	...	$\succ, \sim, \preccurlyeq, ?$
...	...	...	...	...
$a_m$	$\succ, \sim, \preccurlyeq, ?$	$\succ, \sim, \preccurlyeq, ?$	...	$\sim$

Once the matrix in Table 21 is completed (end of the iterations for each pair of alternatives), the code implemented plots the results in form of a directed graph. To eliminate unnecessary edges of the graph, the transitive property of the binary relation  $\succ$  is used. Finally, the graph is coded and plotted (see Figure 10).

```

% Designing graph matrix

Matrix_graph = zeros(candidates);

for row=1:candidates
    for column=1:candidates
        if Necessary_Ranking(row,column)==1
            Matrix_graph(row,column)=1;
        end
    end
end

Graph=digraph(Matrix_graph,Names_candidates.Var1);
plot(Graph)

```

Figure 10: Code for Necessary Ranking graph



## 5 Case study

This chapter presents the results of implementing of the UTA<sup>GMS</sup> method to a sample of Portuguese hospitals. It consists of a description of data (Section 5.1), an explanation of the interaction with the Decision-Maker and the information obtained (Section 5.2), and the analysis and discussion of the results (Section 5.3).

### 5.1 Data and sample

This thesis is based on Portuguese hospitals' quality and efficiency, so the data has been collected from an official source from the Portuguese Central Administration of the Health System (ACSS). The website (<https://benchmarking-acss.min-saude.pt/>) includes a publicly available benchmarking database from year 2018 that has information of a total of 43 institutions and 34 indicators.

#### 5.1.1 Alternatives

From the 43 institutions in the database, some were excluded due to different reasons:

- 4 public-private partnerships (PPP), because their database is not completed.
- 3 oncology institutions (IPOs), as they are really specialized with their technology of production.
- 8 local health units, which include primary care units so they are incomparable to others.
- The institutions *Figueira da Foz District Hospital*, *Santa Maria Maior Hospital* and *Oeste Hospital Centre*, since they do not provide a complete information on all criteria.

After these 18 institutions being excluded, a total number of 25 institutions remain. From these, 5 are hospitals and 20 are hospital centres. The name of each alternative,  $a_k$  for  $k = 1, \dots, 25$ , is presented in Table 22.

#### 5.1.2 Criteria

The 34 indicators considered in the database were initially classified into six categories: Access, Performance Assistance, Productivity, Economic-Financial, Safety and Volume and Usage. To carry out the study, some new indicators are defined grouping the information from various initial indicators,

while others remain intact. In this way, they are regrouped into a total of 24 without losing information. This new group of indicators is classified into the following five categories: Access, Care Appropriateness, Patients Safety, Efficiency, Caesarean Appropriateness. These criteria,  $g_n$  for  $n = 1, \dots, 5$ , are presented in Table 23 and a brief explanation of them is shown after it.

Table 22: Name of the alternatives

$a_1$	Médio Ave Hospital Centre, CPE
$a_2$	Póvoa do Varzim / Vila do Conde Hospital Centre, CPE
$a_3$	Barreiro / Montijo Hospital Centre, CPE
$a_4$	Leiria Hospital Centre, CPE
$a_5$	Setúbal Hospital Centre, CPE
$a_6$	Baixo Vouga Hospital Centre, CPE
$a_7$	Entre Douro e Vouga Hospital Centre, CPE
$a_8$	Médio Tejo Hospital Centre, CPE
$a_9$	Tâmega e Sousa Hospital Centre, CPE
$a_{10}$	Cova da Beira University Hospital Centre, CPE
$a_{11}$	Senhora da Oliveira (Guimarães) Hospital, CPE
$a_{12}$	Santarém District Hospital, CPE
$a_{13}$	Tondela-Viseu Hospital Centre, CPE
$a_{14}$	Trás-os-Montes e Alto Douro Hospital Centre, CPE
$a_{15}$	Algarve University Hospital Centre, CPE
$a_{16}$	Vila Nova de Gaia / Espinho Hospital Centre, CPE
$a_{17}$	Espírito Santo de Évora Hospital, CPE
$a_{18}$	Fernando da Fonseca Hospital, CPE
$a_{19}$	Garcia da Orta Hospital, CPE
$a_{20}$	Lisboa Oidental Hospital Centre, CPE
$a_{21}$	Coimbra University Hospital Centre, CPE
$a_{22}$	Lisboa Central Hospital Centre, CPE
$a_{23}$	São Joao Hospital Centre, CPE
$a_{24}$	Porto University Hospital Centre, CPE
$a_{25}$	Lisboa Norte Hospital Centre, CPE

Table 23: Name of the criteria

$g_1$	Access
$g_2$	Care Appropriateness
$g_3$	Patient Safety
$g_4$	Efficiency
$g_5$	Caesarean Appropriateness



- **Access,  $g_1$** : measures the institution ability to provide care services when any person requires it. It covers aspects as the application of fees per medical act to the users (affordability), the location and distance to the user (physical accessibility) and the size of the waiting lists and waiting times of patients (timeliness).
- **Care Appropriateness,  $g_2$** : measures the ability to deliver the correct treatment to the patients based on medical evidence or scientific knowledge. In other words, it dimensions the capacity to solve the patient's main problem.
- **Patient Safety,  $g_3$** : measures the absence of avoidable harm to the patient during any process and the reductions of all unnecessary risks. It also considers the exposition of patients to chemicals, foreign bodies or infectious agents.
- **Efficiency,  $g_4$** : measures the institution ability to achieve the objectives in relation to the resources consumed. The optimal value of efficiency is reached by being financially sustainable and at the same time deliver the best care possible to the users, taking care of safety, care appropriateness and access and also being cost-effective.
- **Caesarean Appropriateness,  $g_5$** : measures the ability to deliver this practice being based on evidence guidelines. Caesarean sections are very effective at preventing and decreasing mortality in some medically justified situations, but are also dangerous and can cause multiple complications.

The procedure followed to transform the initial database into a workable performances table as is described in an open access document via the *hSNS* project website [24]. In this procedure, the criteria and their values are processed until they reach the form shown in Table 24, where the values are on a numerical scale from 1 to 5 and represent the overall assessment of the alternative in that criterion, being 1 the worst performance and 5 the best. This thesis develops the steps to follow once a performances table like this is available.

Table 24: Performances of each alternative

	$g_1$	$g_2$	$g_3$	$g_4$	$g_5$
$a_1$	3	4	4	4	3
$a_2$	5	4	5	2	3
$a_3$	3	2	4	3	3
$a_4$	2	4	5	3	2
$a_5$	3	4	4	3	3
$a_6$	2	4	3	3	3
$a_7$	2	4	3	3	4
$a_8$	2	2	5	2	3
$a_9$	2	4	4	3	5
$a_{10}$	3	3	3	1	2
$a_{11}$	1	3	4	3	3
$a_{12}$	2	3	3	2	3
$a_{13}$	3	5	2	4	5
$a_{14}$	2	4	4	3	2
$a_{15}$	3	3	3	2	4
$a_{16}$	3	4	3	3	3
$a_{17}$	2	3	2	2	2
$a_{18}$	2	4	3	3	4
$a_{19}$	3	4	3	4	4
$a_{20}$	2	3	2	4	3
$a_{21}$	2	3	4	4	3
$a_{22}$	3	4	2	3	3
$a_{23}$	3	4	3	4	4
$a_{24}$	3	5	3	5	3
$a_{25}$	3	2	2	4	4

## 5.2 Preference parameters

Once the performances table is ready to be used, the method UTA<sup>GMS</sup> can start by interacting with the DM in order to obtain some preference information. The DM for this thesis must be someone who has deep knowledge about the Portuguese health system, specially about NHS.

In this thesis, the Decision-Maker is the Professor *Diogo Filipe Cunha Ferreira*, from the *Civil Engineering Research and Innovation for Sustainability* (CERIS) department in *Instituto Técnico Superior* (IST) from *University of Lisbon* (UL).

The first interaction with the DM is for showing and describing all the data used for the decision problem. After he has understood the model, he is asked if when giving the preference ranking, he wants to specify the level of confidence or to consider the intensity of preference between some pairs of reference alternatives. As the answer is negative, he is asked to analyse the alternatives and the criteria and give an initial ranking of preference of them.

The first iteration is computed with the method UTA<sup>GMS</sup> based on this preference information:

$$\text{Iteration 1: } a_2 > a_{13} > a_{24} > a_{23} > a_{18}$$

The partial preorder resulting from this first iteration and its discussion is presented in the next section (Table 25 and Figure 11).

After seeing the results, the DM asks for a change in the preference information. He is now less confident about the last pairwise comparison  $a_{23} > a_{18}$ , so asks to change it from preference relation to an indifference relation  $a_{23} \sim a_{18}$ . Then, this second iteration is computed with the method UTA<sup>GMS</sup> based on this preference information:

$$\text{Iteration 2: } a_2 > a_{13} > a_{24} > a_{23} \sim a_{18}$$

The results from this second iteration and their discussion are presented in the next section (Table 26 and Figure 12).

The DM asks to test more cases, assuming indifference between two mentioned alternatives and maintaining the other relations of preference. It is decided to make an iteration for each possibility, these being iterations 3, 4 and 5.

$$\text{Iteration 3: } a_2 \sim a_{13} > a_{24} > a_{23} > a_{18}$$

The results from this third iteration and their discussion are presented in the next section (Table 27 and Figure 13).

$$\text{Iteration 4: } a_2 > a_{13} \sim a_{24} > a_{23} > a_{18}$$

The results from this fourth iteration and their discussion are presented in the next section (Table 28 and Figure 14).

$$\text{Iteration 5: } a_2 > a_{13} > a_{24} \sim a_{23} > a_{18}$$

The results from this fifth iteration and their discussion are presented in the next section (Table 29 and Figure 15).

### 5.3 Results and discussion

After applying the developed code of the UTA<sup>GMS</sup> method to the data in Section 5.1, and given the reference information of the DM for the first iteration, the results obtained confirm that there is at least one value function that satisfies this reference information. The partial preorder ( $\succsim_1^N$ ) that has been computed is shown in form of pairwise comparisons in Table 26 and in a graphical form in Figure 11.

Each value in the tables corresponds to the preference relation of the row alternative over the column one.

Table 25 shows the relation between each of the alternatives. In order to be able to represent these values in the form of a graph, it is needed to eliminate the unnecessary relations. Since the binary relation  $\succsim$  is transitive (see properties in Table 3), this property is applied to remove all unnecessary edges of the graph. The graph resulting from this process is shown in Figure 11.

Table 25: Ranking for the 1st iteration

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25	
a1	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	
a2	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a3	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a4	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a5	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a6	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a7	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a8	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a9	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a10	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a11	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a12	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a13	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?
a14	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?
a16	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?
a17	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?
a18	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?
a19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?
a20	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?
a21	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?
a22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?
a23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?
a24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?
a25	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~

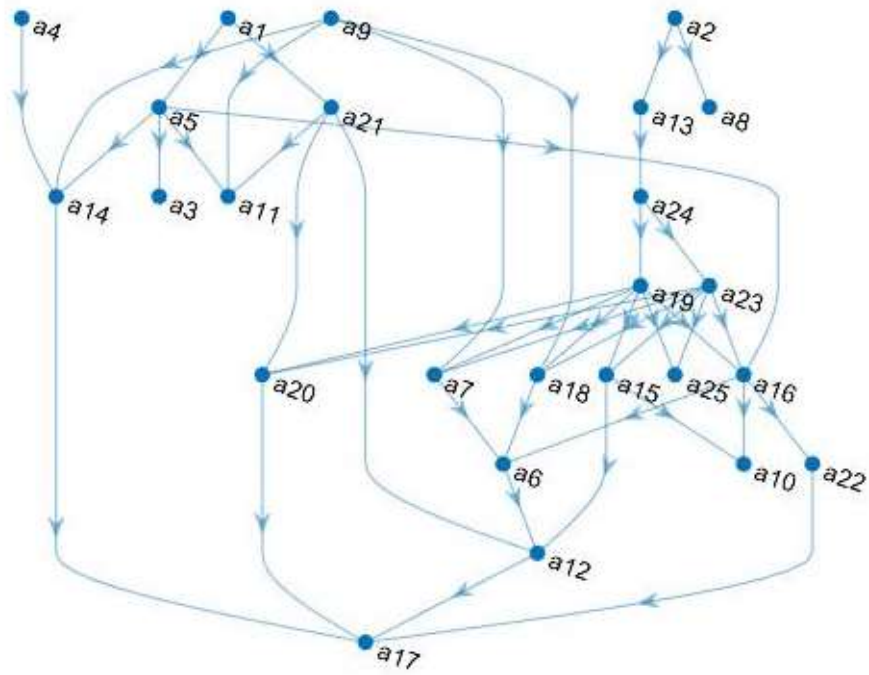


Figure 11: Ranking for the 1st iteration

The exact same procedure is performed with the following iterations. The table-shaped results are shown in Table 26, Table 27, Table 28 and Table 29 and the resulting graphs are shown in Figure 12, Figure 13, Figure 14, Figure 15. In these results it can be seen how the modification of a preference ratio slightly modifies the final ranking.

In none of the iterations are there any incompatibilities with the reference information. If there had been, the code would have detected it and stopped the execution.

In the event of an incompatibility, the DM is informed and must decide whether to accept the incompatibility and follow the current model (bearing in mind that its reference information may not be fully complied with) or whether to modify the information provided in order to make it compatible. Both solutions are valid but involve a different procedure.

Table 26: Ranking for the 2nd iteration

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25	
a1	~	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	
a2	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a3	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a4	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a5	~	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a6	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a7	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a8	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a9	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a10	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a11	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a12	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a13	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?
a14	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?
a16	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?
a17	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?
a18	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?
a19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?
a20	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?
a21	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?
a22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?
a23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?
a24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?
a25	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~

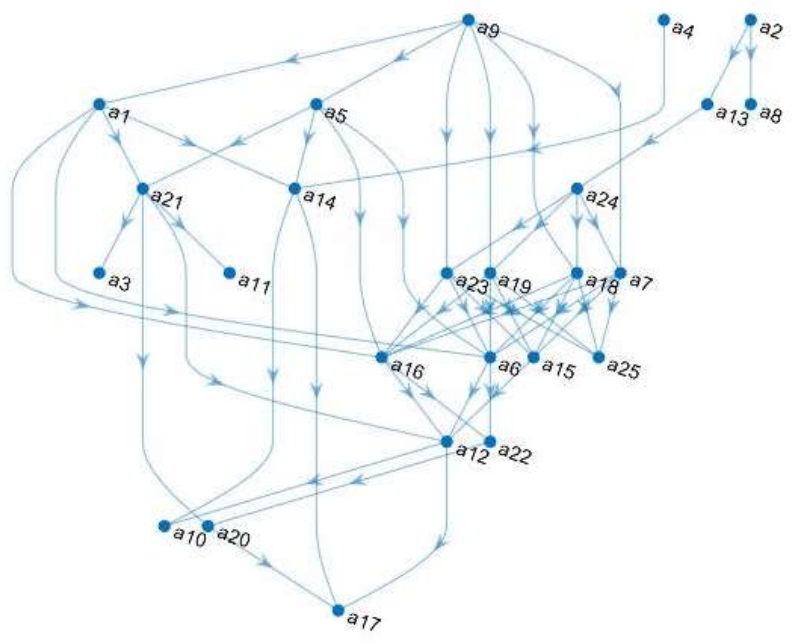


Figure 12: Ranking for the 2nd iteration

Table 27: Ranking for the 3rd iteration

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25
a1	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a2	?	~	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a3	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a4	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a5	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a6	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a7	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a8	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a9	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a10	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a11	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a12	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?
a13	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a14	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?
a15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?
a16	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?
a17	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?
a18	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?
a19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?
a20	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?
a21	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?
a22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?
a23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?
a24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?
a25	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~

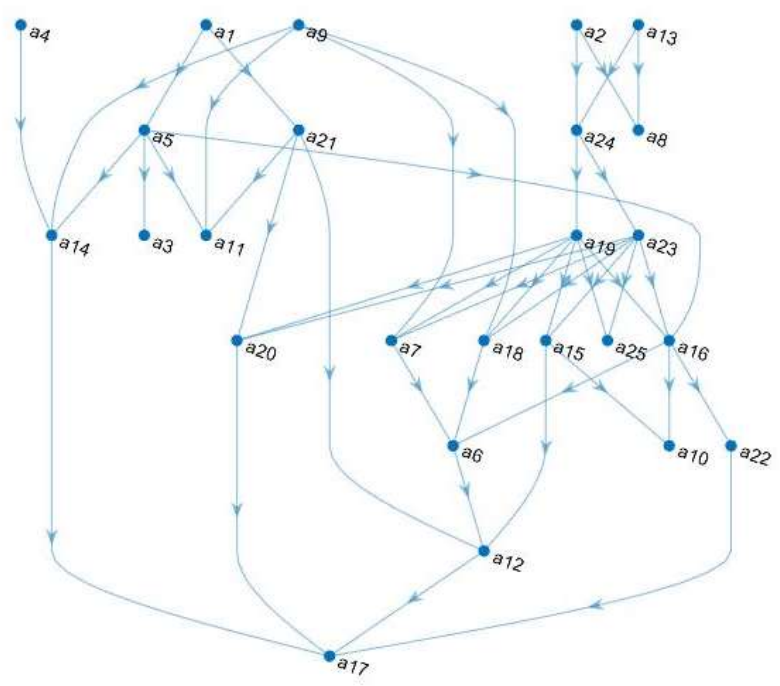


Figure 13: Ranking for the 3rd iteration

Table 28: Ranking for the 4th iteration

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25
a1	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a2	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a3	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a4	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a5	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a6	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a7	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a8	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a9	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a10	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a11	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a12	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?
a13	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a14	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?
a15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?
a16	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?
a17	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?
a18	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?
a19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?
a20	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?
a21	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?
a22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?
a23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?
a24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?
a25	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~

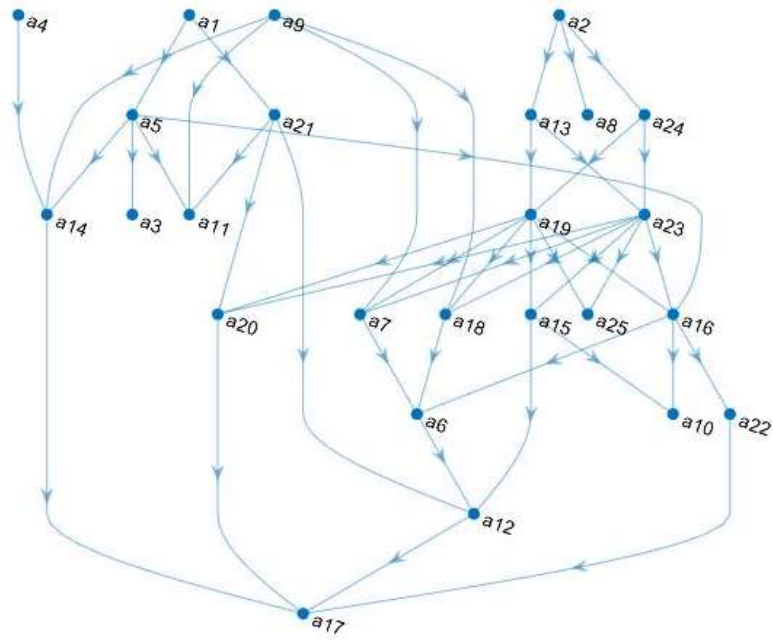


Figure 14: Ranking for the 4th iteration



Table 29: Ranking for the 5th iteration

	a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	a11	a12	a13	a14	a15	a16	a17	a18	a19	a20	a21	a22	a23	a24	a25
a1	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a2	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a3	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a4	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a5	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a6	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a7	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a8	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a9	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a10	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a11	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?	?
a12	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?	?
a13	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?	?
a14	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?	?
a15	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?	?
a16	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?	?
a17	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?	?
a18	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?	?
a19	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?	?
a20	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?	?
a21	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?	?
a22	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?	?
a23	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?	?
a24	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~	?
a25	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	?	~

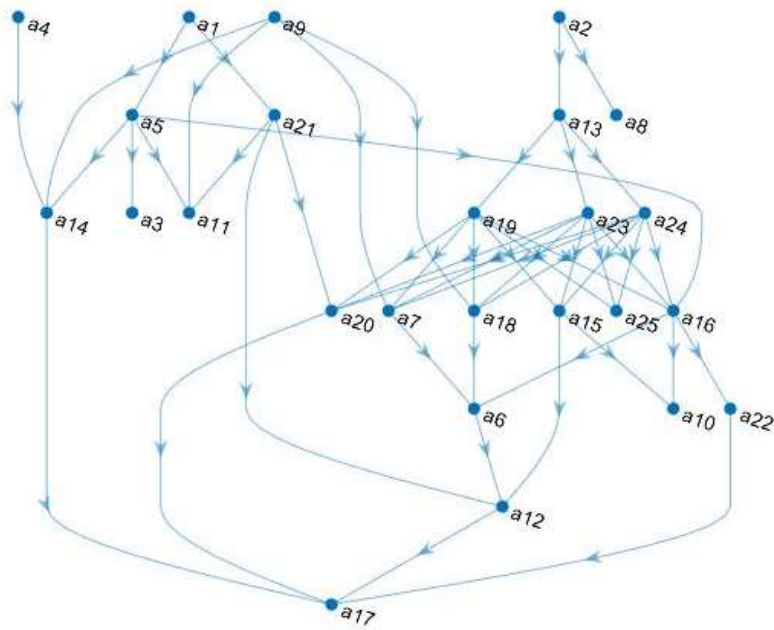


Figure 15: Ranking for the 5th iteration

As can be seen in the graphs above, there are iterations of the procedure that cause more changes in the overall ranking of alternatives than others. Specifically, it can be seen how iterations 1, 3, 4 and 5 have very similar forms, where they only change 3 or 4 relationships between alternatives. This fact is caused directly by the change of the reference relationship and their rearrangement. On the other hand, the 2nd iteration presents many more changes with respect to the others, causing a structural change that modifies an important part of the ranking obtained.

Therefore, the DM needs to focus on the modified preference ratio in the 2nd iteration.

$$\text{Iteration 2: } a_2 > a_{13} > a_{24} > a_{23} \sim a_{18}$$

In this particular iteration the DM presents doubts about the relationship between the alternatives  $a_{23}$  and  $a_{18}$ , modifying the initial information from  $a_{23} > a_{18}$  to  $a_{23} \sim a_{18}$ . What may seem like a decision without too much impact, causes substantial changes. Therefore, the DM would need to make a decision in this regard. Choosing one of the options will significantly change the overall ranking and is therefore an important choice.

At this point, it is decided to make a quick exploration of how the changes made in iterations 3, 4 and 5 would affect once the modification of iteration 2 has been made. In this way, it will be possible to identify whether the DM should make any more important decisions regarding the relationships that are now considered irrelevant. This way, the iterations 6, 7, and 8 are implemented and the results are shown in Figure 16, Figure 17 and Figure 18.

$$\text{Iteration 6: } a_2 \sim a_{13} > a_{24} > a_{23} \sim a_{18}$$

$$\text{Iteration 7: } a_2 > a_{13} \sim a_{24} > a_{23} \sim a_{18}$$

$$\text{Iteration 8: } a_2 > a_{13} > a_{24} \sim a_{23} \sim a_{18}$$

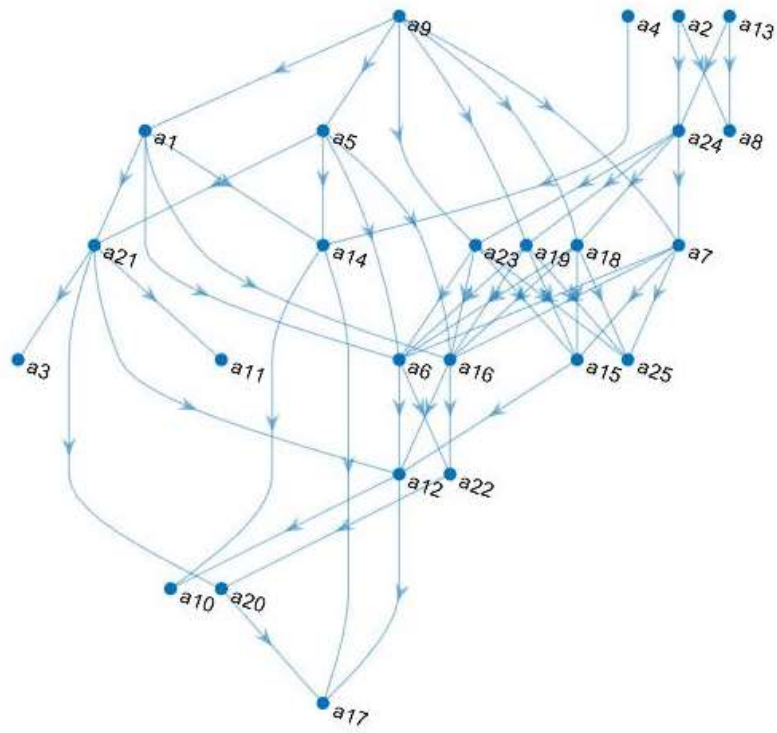


Figure 16: Ranking for the 6th iteration

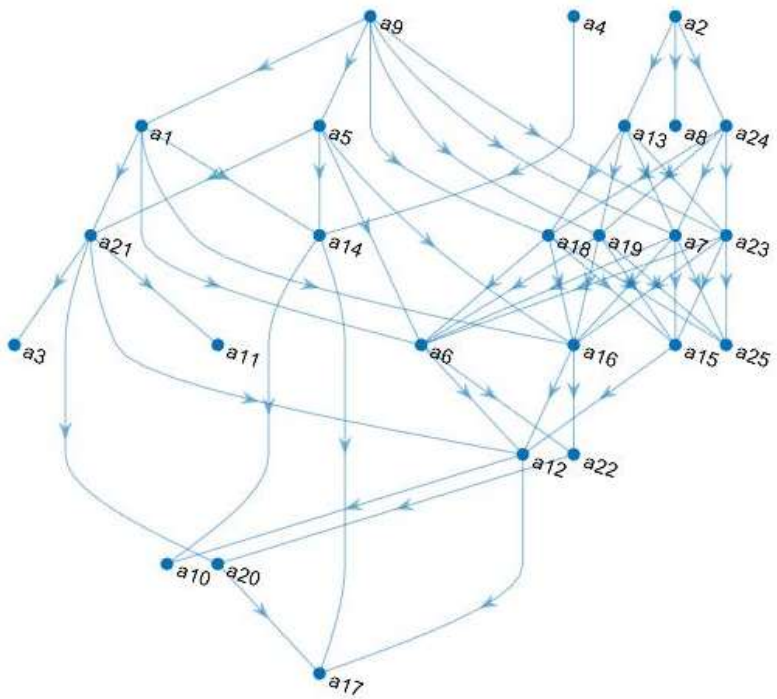


Figure 17: Ranking for the 7th iteration

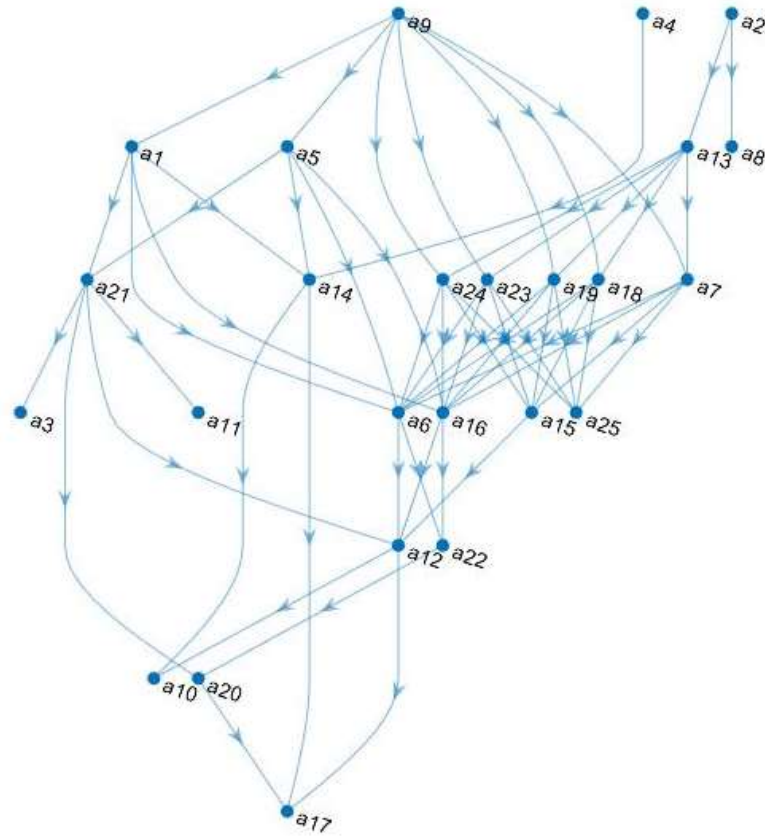


Figure 18: Ranking for the 8th iteration

From the observation of the previous figures and the analysis of the corresponding tables of values, it is concluded that the changes of reference information made in iterations 6, 7 and 8 do not have a great impact on the final result. Thus, it is clear that the main case to be addressed is the relationship between alternatives 23 and 18, which is up to the DM.

The DM is finally given the resulting rankings taking into account the two possible relationships between the alternatives  $a_{23}$  and  $a_{18}$  and its small variations depending on other modifications.

Further studies could be conducted, either in relation to the combinations between relationships of these alternatives or by adding and/or removing alternatives from the reference information. The implemented code provides results very quickly, so it is possible to explore hypothetical situations without having to spend too many resources.

## 6 Conclusions

This chapter presents the general and specific conclusions of the work in this thesis (Section 6.1) and proposes some ideas to be explored in a future work in the topic (Section 6.2).

### 6.1 Conclusions

Portugal's public health system (NHS) has suffered economically in recent years, gradually reducing the quality of its service. Public health offers service to all citizens so it is paid by general taxation. It is important, therefore, to analyse their satisfaction with the service they receive from the country's public hospitals.

The *hSNS* project aims to analyse the performance of Portuguese public hospitals in order to find the reasons why patients are dissatisfied and try to improve it. This thesis focuses on obtaining an objective ranking of these hospitals based on the assessment that each Portuguese public hospital obtains in certain previously selected criteria. The objectives of this thesis have been described in Section 1.2.

The first one was to implement the UTA<sup>GMS</sup> method in a MATLAB-CPLEX environment. The objective was achieved by doing a literature review on both the theoretical part of the method (Section 2 and Section 3) and the technical part of implementing it in the MATLAB software (Section 4). Achieving this goal has involved more work than expected and has therefore taken up most of the time devoted to this thesis.

The second objective was to apply the implemented method to a real case in health sector to get the additive value functions of the public Portuguese hospitals. The work delivers a ranking of the hospitals analysed based on their additive value functions and some information delivered by the DM (Section 5), and therefore, although not being exactly the description of the objective, it can be considered as accomplished as this was the main idea. In the next section, a method is presented to, later to obtain this ranking, also to obtain a representative additive value function of each hospital.

Then, it can be considered that both objectives set in the beginning of this thesis have been accomplished and the work is complete.

## 6.2 Future research

In line with what was mentioned in the previous section, we can delve further into the method implemented in this thesis in future studies.

By applying the  $UTA^{GMS}$  method to the data of Portuguese hospitals, a ranking of the different alternatives is obtained according to their performance and the information provided by the DM. This ranking is obtained from maximizing and minimizing the difference of the additive value functions of the different alternatives, but does not calculate their specific value. In this way, a more than interesting result is obtained in the form of a ranking among the hospitals, but its function is not obtained. If in later studies it is wanted to obtain a representative function of each alternative, a possible method to be used is the one described in the document [25].

Briefly describing this new method, it aims at selecting a single compatible value function which constitutes a synthetic representation of the outcomes of the  $UTA^{GMS}$  method. To make this selection, it exploits the necessary and possible results of applying all compatible value functions, and subsequently enhances differences between comprehensive values of pairs of alternatives, taking into account what is observed for them. Whether this difference is emphasized or reduced in the representative value function depends on the existence of, respectively, undeniable advantage of some alternative over the others or ambiguity in the statement of such an advantage in the context of recommendations provided by all compatible value functions. The representative value function represents all compatible value functions, which also do contribute to its definition. The representative instance should be rather perceived as continuation of the trend of taking into account all compatible value functions, which allows to address potential problems with interpretation of the necessary, possible, and extreme results. Note that introducing the concept of a representative value function, it extends  $UTA^{GMS}$  in its capacity of explaining the final output in terms of a value function that can be displayed to the DM.

Also, although the possible incompatibilities that may arise when applying the  $UTA^{GMS}$  method have been studied, it might be interesting to find a way to automate a solution in this regard, since the code provided what it does is indicate that there is some incompatibility and what may be the reason.

Finally, the usefulness of the written code in other areas and sectors could be studied, so that by making slight modifications it adapts even more to the data entered and that the results obtained can be displayed in a more appropriate way if necessary.

# References

- [1] Simões J, Augusto GF, Fronteira I, Hernández-Quevedo, “C. Portugal: Health system review,” in *Health Systems in Transition*, 2017, p. 1–184.
- [2] Campos-Matos I., Russo G., Perelman J., “Connecting the dots on health inequalities - A systematic review on the social determinants of health in Portugal,” *International Journal for Equity in Health*, 2016.
- [3] A. Nunes, “The recovery of the Portuguese national health service: new perspectives for health,” *Health Systems Management Journal*, p. 16–28, 2017.
- [4] A. Nunes, D. Ferreira, and A. Fernandes., “Financial Crisis in Portugal : Effects in the Health,” *International Journal of Health Services*, pp. 237-259, 2019.
- [5] Russo G., Rego I., Perelman J., “A tale of loss of privilege, resilience and change: the impact of the economic crisis on physicians and medical services in Portugal,” *Health Policy*, pp. 1079-1086, 2016.
- [6] D. Ferreira and R. Marques, «Do quality and access to hospital services impact on their technical efficiency?,» *Omega*, pp. 218-236, 2019.
- [7] M. KOKSALAN, J. Wallenius, and S. Zionts, “An Early History of Multiple Criteria Decision Making,” in *Journal of Multi-Criteria Decision Analysis*, 2013, pp. 87-94.
- [8] B. Roy., “Paradigms and Challenges.,” in *Multiple Criteria Decision Analysis: State of the Art Surveys. International Series in Operations Research & Management Science*, 2005, pp. 70-82.
- [9] V. Belton and T. Stewart., *Multiple Criteria Decision Analysis: An Integrated Approach.*, US: Springer, 2002.
- [10] Roy, B., Vanderpooten, D., “The European School of MCDA: Emergence, Basic Features and Current Works,” in *Journal of Multi-criteria Decision Analysis*, v.5, 1996, pp. 22-38.
- [11] J. Almeida-Dias, J. Figueira, and B. Roy., “A multiple criteria sorting method where each category is characterized by several reference actions : The ELECTRE TRI-nC method,” in *European Journal of Operational Research*, 2012, pp. 567-579.
- [12] Ishizaka, Alessio & Pearman, Craig & Nemery, Philippe., “AHPSort: an AHP-based method for sorting problems.,” in *International Journal of Production Research*, 2012, pp. 1-18.
- [13] J. Wa\_trobski, J. Jankowski, P. Ziemba, A. Karczmarczyk, and M. Ziolo, in *Generalised framework for multi-criteria method selection*, Omega, 2019, pp. 107-124.

- [14] Dyer J.S., "Maut — Multiattribute Utility Theory," in *Multiple Criteria Decision Analysis: State of the Art Surveys. International Series in Operations Research & Management Science, vol 78*, New York, Springer, 2005.
- [15] E. Triantaphyllou, in *Multi-criteria decision making methods: A comparative study*, Kluwer Academic Publishers, 2002.
- [16] A. Costa and J. Figueira, "O metodo ELECTRE TRI-nC," in *A investigação Operacional em Portugal: Novos desafios, Novas ideias*, 2016, pp. 203-218.
- [17] Siskos, Y. and Spyridakos, "Intelligent multicriteria decision support: Overview and perspectives," in *European Journal of Operational Research*, 1999, pp. 236-246.
- [18] Roy, B., *Méthodologie Multicritère d'Aide à la Décision*, Paris: Economica, 1985.
- [19] Keeney, R.L., *Decisions with multiple objectives: Preferences and value tradeoffs*, New York: John Wiley and Sons, 1976.
- [20] Jacquet-Lagrèze, E. and J.Siskos, "Assessing a set of additive utility functions for multicriteria decision making: The UTA method," in *European Journal of Operational Research*, 1982, pp. 151-164.
- [21] G. S. Michel Beuthe, "Comparative analysis of UTA multicriteria methods," in *European Journal of Operational Research*, 2001, pp. 246-262.
- [22] Salvatore Greco, Vincent Mousseau, Roman Slowinski, "Ordinal regression revisited: Multiple criteria ranking using a set of additive value functions," in *European Journal of Operational Research*, 2008, pp. 416-436.
- [23] Diogo Ferreira, Isabel Camoes, «On implementing the UTA method on MATLAB-CPLEX,» 2016.
- [24] Rocha, António Maria de Macieira Condeixa da Conceição, "Quality assessment of the Portuguese public hospitals: A multicriteria approach," 2019.
- [25] Milosz Kadzinski, Salvatore Greco, Roman Slowinski, "Selection of a representative value function in robust multiple criteria ranking and choice," in *European Journal of Operational Research*, 2011, pp. 541-553.
- [26] Almeida Simoes J, Augusto G, Fronteira I, Hernandez-Quevedo C, "Portugal: Health System Review," *Health systems in transition*, pp. 1-184, 2017.



# APPENDICES

## Appendix A: Some parts of the Matlab code

```
% CREATING THE MODEL

% objective function f
f=[zeros(1, (Nendpoints-1)*n), ones(1,p)];

% lower and upper bounds lb
lb=[0*ones(1, (Nendpoints-1)*n), zeros(1, p)];
ub=Inf*ones(1, (Nendpoints-1)*n + p);

% matrixes A, b, Aeq and beq
A=[]; b=[]; Aeq=[]; beq=[];

for k=1:p-1
    ak=InitPref(k); ak1=InitPref(k+1);
    if ak<ak1 % ak is preferred over a(k+1)
        v=zeros(1,p); v(k)=1; v(k+1)=-1;
        w=Mat(k,:)-Mat(k+1,:);
        A=[A; -[w, v]];
        b=[b; -epsilon];
    else
        v=zeros(1,p); v(k)=1; v(k+1)=-1;
        w=Mat(k,:)-Mat(k+1,:);
        Aeq=[Aeq; [w, v]];
        beq=[beq; 0];
    end
end

mat2=zeros(n*(Nendpoints-1), (Nendpoints-1)*n);
for i=1:n*(Nendpoints-1)-1
    mat2(i,i)=1; mat2(i,i+1)=-1;
end

i=(1:n)*(Nendpoints-1);
```

```

mat2(i,:)=[];
mat2=[mat2, zeros(n*(Nendpoints-2),p)];
A=[A; mat2];
b=[b; -Sji*ones(n*(Nendpoints-2),1)];

vaux=zeros(1,(Nendpoints-1)*n);
for k=1:n
    vaux((Nendpoints-1)*k)=1;
end

vaux=[vaux, zeros(1,p)];
Aeq=[Aeq; vaux];
beq=[beq; 1];

options = cplexoptimset;

[F, ~, flag]=cplexlp(f,A,b,Aeq,beq,lb,ub);

```

```
% FINDING INCOMPATIBILITIES
```

```
if flag==1
    Vaux=F(1:(Nendpoints-1)*n);
    F(1:(Nendpoints-1)*n)=[];
    sigma=F;
    margUtF=[];
    for k=1:n
        margUtF=[margUtF; [0 Vaux((Nendpoints-1)*(k-1)+1:(Nendpoints-1)*k) ']];
    end
    Maux= repmat(Vaux',p,1) .* Mat;
    Uga=[];
    for k=1:n
        Uga=[Uga sum(Maux(:,(Nendpoints-1)*(k-1)+1:(Nendpoints-1)*k),2)];
    end
    globalUtF=sum(Maux,2)+sigma;
    globalUtFnoerror=sum(Maux,2);
    Maux2=[Vec, globalUtF, globalUtFnoerror,Uga];
    Maux2=sortrows(Maux2,1);
    globalUtF=Maux2(:,2);
    globalUtFnoerror=Maux2(:,3);
    Uga=Maux2(:,4:end);
    Resultats = [Uga, globalUtFnoerror, globalUtF]
    if not(sum(sigma)==0)
        INCOMPATIBLE = 1
    end
end
end
```

```
% OPTIMIZATION
```

```
    % d(a,b)
    f = Mat(end-1,:) - Mat(end,:);
    [F, d_a_b, flag] = cplexlp(f,A,b,Aeq,beq,lb,ub,options);
    if flag <= 0
        NotSolution = "Incompatibility has ben found"
    end
```

```
    % d(b,a)
    f = - Mat(end-1,:) + Mat(end,:);
    [F, d_b_a, flag] = cplexlp(f,A,b,Aeq,beq,lb,ub,options);
    if flag <= 0
        NotSolution = "Incompatibility has ben found"
    end
```

```
% ADDING OPTIMIZATION INFORMATION TO GENERAL MATRIX
```

```
    if or(and(d_a_b>0, d_b_a<0), and(d_a_b==0, d_b_a<0))
        Necessary_Ranking(candidate_a,candidate_b) = 1; %a>b
    end
    if and(d_a_b==0, d_b_a==0)
        Necessary_Ranking(candidate_a,candidate_b) = 2; %a=b
    end
    if or(and(d_a_b<0, d_b_a>0), and(d_a_b<0, d_b_a==0))
        Necessary_Ranking(candidate_a,candidate_b) = 3; %a<b
    end
    if and(d_a_b<0, d_b_a<0)
        Necessary_Ranking(candidate_a,candidate_b) = 4; %a?b
    end

end

end
```

```

% ELIMINATING UNNECESSARY LINES IN GRAPH

Necessary_Ranking_large = Necessary_Ranking;
for row=1:candidates
    worse=[];
    better=[];
    for column=1:candidates
        if Necessary_Ranking(row,column)==3
            worse(end+1)=column;
        end
        if Necessary_Ranking(row,column)==1
            better(end+1)=column;
        end
    end
    if and(not(size(worse)==0),not(size(better)==0))
        for eliminat=worse
            for pitjor=better
                Necessary_Ranking(eliminat,pitjor)=0;
            end
        end
    end
end

% DESIGNING GRAPH MATRIX

Matrix_graph = zeros(candidates);
for row=1:candidates
    for column=1:candidates
        if Necessary_Ranking(row,column)==1
            Matrix_graph(row,column)=1;
        end
    end
end

Graph=digraph(Matrix_graph,Names_candidates.Var1);
plot(Graph)

```

