

LAMSADE

M1 MIAGE

INTERNSHIP

---

# Internship Report

---

*Student:*

Elie ABI HANNA DAHER

*Supervisor:*

M. Olivier CAILLOUX

M. Michel ZAM

25th August, 2017



### **Abstract**

This report documents the work done during the summer internship at LAMSADE Dauphine, under the supervision of Mr. Olivier Cailloux and Mr. Michel Zam. The report give an overview of the projects completed during the internship.

I have worked mainly on the UTA method which is an ordinal regression method, proposed by E. Jacquet-Lagereze and J. Siskos, which adjust a system of additive utility function.

# Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Context of the projects</b>	<b>3</b>
<b>3</b>	<b>Description of the projects</b>	<b>4</b>
3.1	UTA . . . . .	4
3.1.1	Introduction of the aid decision . . . . .	4
3.1.2	Principles and Notation . . . . .	6
3.1.3	Development . . . . .	7
3.1.4	Example . . . . .	8
3.2	Linear Program Solver . . . . .	9
3.3	Research . . . . .	11
<b>4</b>	<b>Conclusion</b>	<b>12</b>
	<b>Bibliography</b>	<b>13</b>
	<b>Table of figures</b>	<b>14</b>
	<b>Appendices</b>	<b>15</b>
<b>A</b>	<b>Compte Rendus avec M. Cailloux</b>	<b>16</b>
A.1	CR1 - 15 juin 2017 . . . . .	16
A.2	CR2 - 21 juin 2017 . . . . .	17
A.3	CR3 - 28 juin 2017 . . . . .	18
A.4	CR4 - 18 juillet 2017 . . . . .	19
A.5	CR5 - 25 juillet 2017 . . . . .	20
<b>B</b>	<b>Example of UTASTAR - Analyzing the choice of transportation</b>	<b>21</b>

# Chapter 1

## Introduction

During my first year in Masters MIAGE at Paris Dauphine University, I had the opportunity to achieve an internship at LAMSADE (Laboratoire d'analyse et modélisation de systèmes pour l'aide à la décision). LAMSADE is a Paris-Dauphine research laboratory specialized in decision theoretic approaches. Decision theory aims at helping individuals who face decision problem by elaborating a study of the reasoning underlying agent's choices.

From 29th May, 2017 to 1st September, 2017, I completed a research internship at LAMSADE at Paris Dauphine University. My internship was made remotely, so I had a meeting with Mr. Cailloux every week to get feedback of the work done. And I communicated with Mr. Zam via mails or calls.

This internship was very important for my professional career because it may be my last internship before I get my Masters's degree in Information systems for finance at Paris Dauphine University. Let's not also forget that this internship will allow me to practice the different courses learned during my academic years.

One of the objectives is to propose the UTA method as an open source software component. Another objective is to integrate this open source software into DecisionCloud, a software based on MyDraft a tool developed by KarmicSoft a LAMSADE spin-off. Researching similar literature and research represent an important objective in this internship.

In this report, I will present the context of the internship. It contains an overview of the projects realized: UTA, LinearProgramSolver, Research, DecisionCloud... Writing this report, I will also conclude by reflecting on my learning objects and goals achieved and not achieved.

## Chapter 2

# Context of the projects

# Chapter 3

## Description of the projects

### 3.1 UTA

#### 3.1.1 Introduction of the aid decision

In decision theory a basic problem involving several criterias, concerns the way that the final decision should be made. However, this problem is posed in the opposite way: assuming that the decision is given, how is it possible to find the rational basis for the decision being made? Or how is it possible to assess the model leading to exactly the same decision as the actual one or the most "similar" decision.

In multicriteria decision aid methods (MCDA), the following concepts play a fundamental role in decision-making problems:

1. object of the decision, definition of the set of potential actions(alternatives) and the determination of a problem statement
2. modeling a consistent family of criteria
3. defining a global preference model
4. decision-aid or decision support

#### Example: Buying a new car

Let's consider we are trying to figure out which car to buy. Using the methodology represented above, we state that the objective of this problem is **buying a new car**. After stating the objective, we can list the potential of actions that represent the list of cars that we may buy. By potential action, we designate that which constitutes the object of the decision. An action is qualified as potential when it is deemed possible to implement it or if it has some interest within the decision aiding process. So the following list represent the potential actions of this example:

- Peugeot 208 GTi
- Nissan Sentra
- Citroen C4
- Peugeot 308 berline

After listing the list of potential cars, we can define a list of criteria that we will base our decision on. A criterion is constructed to evaluate and compare potential actions according to a point of view. So when defining the list of criteria you should always remember that they must be easy to evaluate (easy to convert to a scale) and should be logical. Let's say we will base our purchase on the following criteria:

- price (in Euro)
- comfort (0, +, ++, +++) *0 being not comfortable and +++ very comfortable*

- safety (1, 2, 3, 4, 5) *1 being not safety and 5 safe*

During the decision process, we will determinate the global preferences of the potential actions:

1. Citroen C4
2. Peugeot 208 GTi
3. Peugeot 308 berline
4. Nissan Sentra

Once the global preference is defined, we can start the decision support.

One of the multi-criteria decision analysis methods is the UTA method, which was proposed by E. Jacquet-Lagrèze and J. Siskos in 1982. This method is proposed by the Multi-Attribute Utility Theory (MAUT) that build a utility function based on the DM<sup>1</sup> preferences.

The UTA method is used to solve a multi-criteria problem by building a utility function based on the preferences of the DM and solving a linear program (LP). It adopt the aggregation-disaggregation principles: where the model is based on a given preferences.

The UTASTAR, a variant of the UTA method, has been considered a better algorithm than UTA. Better result were found using the UTASTAR algorithm. So this is why we will focus on this method rather than the UTA method.

The aim of the UTASTAR method is to estimate a set of additive utility functions which are as consistent as possible with the decision maker's preferences.

At the begining of the problem, the DM should present the following information

- rank of the actions
- give the criteria he want to base his decision on
- evaluate the action compared to the criterion

Once those information are presented, the UTASTAR algorithm can be executed.

---

<sup>1</sup>Decision Maker

### 3.1.2 Principles and Notation

Let's call  $A = a, b, c, \dots$  the set of potential actions and  $g_1, g_2, g_3, \dots, g_n$  the family of criteria. Where  $g_i(a)$  represent the funtion of an action(alternative) $a$  on the criteria  $g_i$  with  $a \in A_R$ .

We define  $g_{i*}$  as the least preferred criteria:  $g_{i*} = \min_{a \in A} g_i(a)$  and  $g_i^*$  as the most preferred criteria:  $g_i^* = \max_{a \in A} g_i(a)$ . So the interval for each criteria  $g_i$  is:  $[g_{i*}, g_i^*]$ .

If we want to evaluate two actions, for example  $a$  and  $b$ , on only one criteria  $g_i$  we have the following relations:

$$\begin{cases} a \succ b \Leftrightarrow g_i(a) > g_i(b) & \text{preference} \\ a \sim b \Leftrightarrow g_i(a) = g_i(b) & \text{indifference} \end{cases} \quad (3.1)$$

The criteria aggregation model in UTASTAR has the following form:

$$v(g(a)) = \sum_{i=1}^n v_i(g_i(a)) \quad (3.2)$$

subject to normalization constraints:

$$\begin{cases} \sum_{i=1}^n v_i(g_i^*) = 1 \\ v_i(g_{i*}) = v_i(g_i^1) = 0, \forall i = 1, 2, \dots, n \end{cases} \quad (3.3)$$

where  $v_i, i = 1, 2, \dots, n$  are non decreasing real valued function.

In UTASTAR we have

$$w_{ij} = v_i(g_i^{j+1}) - v_i(g_i^j) \geq 0 \quad \forall i \quad j \quad (3.4)$$

Which will allow us to write:

$$v_i(g_i^j) = \sum_{t=1}^{j-1} w_{it} \quad \forall i = 1, 2, \dots, n \quad \text{and} \quad j = 2, 3, \dots, \alpha_i - 1 \quad (3.5)$$

With the evaluation of an action a  $g(a) = [g_1(a), g_2(a), \dots, g_n(a)]$ , we have the following relation:

$$\begin{cases} v[g(a)] > v[g(b)] \Leftrightarrow a \succ b \\ v[g(a)] \sim v[g(b)] \Leftrightarrow a = b \end{cases} \quad (3.6)$$



### 3.1.3 Development

The updated version of UTA, UTASTAR, propose a double error function for each action:  $\sigma^+(a)$  and  $\sigma^-(a)$ . So the value of each alternative  $a \in A_R$  can be written:

$$v'[g(a)] = \sum_{i=1}^n v_i[g_i(a)] - \sigma^+(a) + \sigma^-(a) \quad \forall a \in A_R \quad (3.7)$$

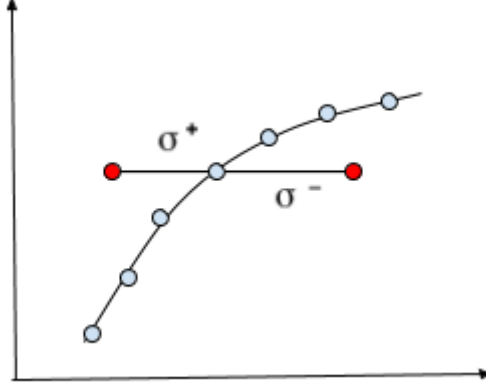


Figure 3.1 – Error function in UTASTAR

For each criteria, the interval  $[g_{i*}, g_i^*]$  is cut into  $(\alpha_i - 1)$  equals interval, and the end points  $g_i^j$  are given by the formula:

$$g_i^j = g_{i*} + \frac{j-1}{\alpha_i - 1} (g_i^* - g_{i*}) \quad \forall j = 1, 2, \dots, \alpha_i \quad (3.8)$$

The marginal value of an action  $a$  is calculated by a linear interpolation

$$v_i[g_i(a)] = v_i(g_i^j) + \frac{g_i(a) - g_i^j}{g_i^{j+1} - g_i^j} [v_i(g_i^{j+1}) - v_i(g_i^j)] \quad (3.9)$$

The set of reference action  $A_R = a_1, a_2, \dots, a_m$  is arranged where  $a_1$  is the best action and  $a_m$  is the worst action. Which indicate that we have two possible situations:

- $a_k \succ a_{k+1}$  preference
- $a_k \sim a_{k+1}$  indifference

So if we have that  $\Delta(a_k, a_{k+1}) = v'[g(a_k)] - v'[g(a_{k+1})]$  and  $\delta$  is a small positive number we will obtain the following relations:

$$\begin{cases} \Delta(a_k, a_{k+1}) \geq \delta \\ \Delta(a_k, a_{k+1}) = 0 \end{cases} \quad (3.10)$$

The marginal value functions are estimated by means of the Linear Programm with (3.2), (3.3), (3.10) as constraints, and an objective function depending on the  $\sigma^+$  and  $\sigma^-$ :

$$[min]z = \sum_{k=1}^m [\sigma^+(a_k) + \sigma^-(a_k)]$$

subject to:

$$\begin{cases} \Delta(a_k, a_{k+1}) \geq \delta \quad or \quad \Delta(a_k, a_{k+1}) = 0 \\ \sum_{i=1}^n \sum_{j=1}^{\alpha_i-1} w_{ij} = 1 \\ w_{ij} \geq 0, \quad \sigma^+(a_k) \geq 0, \quad \sigma^-(a_k) \geq 0, \quad \forall i, j \text{ and } k \end{cases} \quad (3.11)$$

### 3.1.4 Example

The implementation of UTASTAR algorithm is illustrated by the example from the Introduction chapter : **buying a new car**.

The DM is interested only in the following criteria:

- price (in Euro)
- comfort (0, +, ++, +++) 0 being not comfortable and +++ very comfortable
- safety (1, 2, 3, 4, 5) 1 being not safety and 5 safe

The evaluation of the previous criteria is presented in the following table:

Cars	Price	Comfort	Safety	Ranking of the DM
Nissan Sentra (ns)	17 000	+++	4	1
Citroen C4 (c4)	15 000	++	2	2
Peugeot 208 GT (p208)	25 000	+	3	3
Peugeot 308 berline (p308)	18 500	0	3	4

First of all, we should specify the scale <sup>2</sup> for each criteria.

- Price  $\Rightarrow [g_{1*}, g_1^*] = [25\ 000, 20\ 000, 15\ 000]$
- Comfort  $\Rightarrow [g_{2*}, g_2^*] = [0, +, ++, +++]$
- Safety  $\Rightarrow [g_{3*}, g_3^*] = [1, 3, 5]$

According to this formula:  $v(g(a)) = \sum_{i=1}^n v_i(g_i(a))$ , the value of each alternative may be written:

- $v(g(ns)) = 0.4v_1(15\ 000) + 0.6v_1(20\ 000) + v_2(+++) + 0.5v_3(3) + 0.5v_3(5)$
- $v(g(c4)) = v_1(15\ 000) + v_2(++) + 0.5v_3(1) + 0.5v_3(3) = v_1(15\ 000) + v_2(++) + 0.5v_3(3)$
- $v(g(p208)) = v_1(25\ 000) + v_2(+) + v_3(3) = v_2(+) + v_3(3)$
- $v(g(p308)) = 0.3v_1(15\ 000) + 0.7v_1(20\ 000) + v_2(0) + v_3(3) = 0.3v_1(15\ 000) + 0.7v_1(20\ 000) + v_3(3)$

We have that  $v_1(25\ 000) = v_2(0) = v_3(1) = 0$ .

Since the marginal value  $u_i(g_i)$  can be expressed in terms of variables  $w_{ij}$ :  $u_i(g_i^j) = \sum_{t=1}^{j-1} w_{it}$ , the value of each alternatie can be written:

- $v(g(ns)) = w_{11} + 0.4w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + 0.5w_{32}$
- $v(g(c4)) = w_{11} + w_{12} + w_{21} + w_{22} + 0.5w_{31}$
- $v(g(p208)) = w_{21} + w_{31}$
- $v(g(p308)) = w_{11} + 0.3w_{12} + w_{31}$

For each pair of consecutive alternatives, we express the difference between them:

- $\Delta(ns, c4) = -0.6w_{12} + w_{23} + 0.5w_{31} + 0.5w_{32} - \sigma_{ns}^+ + \sigma_{ns}^- + \sigma_{c4}^+ - \sigma_{c4}^-$
- $\Delta(c4, p208) = w_{11} + w_{12} + w_{22} - 0.5w_{31} - \sigma_{c4}^+ + \sigma_{c4}^- + \sigma_{p208}^+ - \sigma_{p208}^-$
- $\Delta(p208, p308) = w_{21} - w_{11} - 0.3w_{12} - \sigma_{p208}^+ + \sigma_{p208}^- + \sigma_{p308}^+ - \sigma_{p308}^-$

Having  $\delta = 0.05$ , we can solve the following LP:

Objective:

$$\text{Minimize} \quad \sum_{a \in A} \sigma_a^+ + \sigma_a^- \quad (3.12)$$

---

<sup>2</sup>the interval  $[g_{i*}, g_i^*]$  is cut into equal intervals

Subject to:

$$\begin{cases} -0.6w_{12} + w_{23} + 0.5w_{31} + 0.5w_{32} - \sigma_{ns}^+ + \sigma_{ns}^- + \sigma_{c4}^+ - \sigma_{c4}^- \geq 0.05 \\ w_{21} - w_{11} - 0.3w_{12} - \sigma_{p208}^+ + \sigma_{p208}^- + \sigma_{p308}^+ - \sigma_{p308}^- \geq 0.05 \\ -0.1w_{12} - w_{21} - w_{22} - w_{23} - 0.5w_{32} - \sigma_{p308}^+ + \sigma_{p308}^- + \sigma_{ns}^+ - \sigma_{ns}^- \geq 0.05 \\ w_{11} + w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + w_{32} = 1 \end{cases} \quad (3.13)$$

An optimal solution is  $w_{12} = 0.34$ ,  $w_{21} = 0.152$ ,  $w_{31} = 0.51$  with  $\sum_{a \in A} \sigma_a^+ + \sigma_a^- = 0$ . The utilities found for each alternative are as follows:

- $v(g(ns)) = 0.798$
- $v(g(c4)) = 0.747$
- $v(g(p208)) = 0.662$
- $v(g(p308)) = 0.62$

Those utilities are consistent with the DM's preference ranking.

The UTA method build a utility function based on the preferences of the DM and it consist in solving a linear program (LP) to solve a multi-criteria problem.

This method will elaborate a model of preferences which is as similiar as possible to the DM's preferences.

An improved version of the UTA is the UTASTAR. In UTA we used a single error  $\sigma(a)$  in UTASTAR we use a double positive error function. The updated version has performed better than the regular method.

### 3.2 Linear Program Solver

One of the steps of the UTA algorithm is solving the Linear Program. So we can complete the UTA algorithm, I created an independant java application that has the objectif of solving the LP by finding the optimal solution.

So i can achieve this goal I had to use the google ortools (Optimization tools) library. You can find those library in the github repository under src/libs.

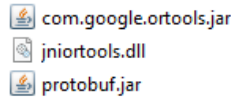


Figure 3.2 – Library used in the solver

Let's talk the following example:

Objective:

$$\text{Maximize } 10x_1 + 6x_2 + 4x_3 \quad (3.14)$$

subject to:

$$\begin{cases} x_1 + x_2 + x_3 \leq 100 \\ 10x_1 + 4x_2 + 5x_3 \leq 600 \\ 2x_1 + 2x_2 + 6x_3 \leq 300 \end{cases} \quad (3.15)$$

So we can solve the LP, we need to create an instance of the solver:

```

MPSolver solver = new MPSolver(
    "LinearProgrammingExample",
    MPSolver.OptimizationProblemType.valueOf("GLOP_LINEAR_PROGRAMMING")
);

```

Figure 3.3 – Java code to define the solver

After that, we define the 3 variables  $x_1$ ,  $x_2$  et  $x_3 \in [0; \infty]$ :

```

MPVariable x1 = solver.makeNumVar(0.0, infinity, "x1");
MPVariable x2 = solver.makeNumVar(0.0, infinity, "x2");
MPVariable x3 = solver.makeNumVar(0.0, infinity, "x3");

```

Figure 3.4 – Java code to define the variables of the LP

We define the objective  $Maximize \quad 10x_1 + 6x_2 + 4x_3$ :

```

MPObjective objective = solver.objective();
objective.setCoefficient(x1, 10);
objective.setCoefficient(x2, 6);
objective.setCoefficient(x3, 4);
objective.setMaximization();

```

Figure 3.5 – Java code to define the objective of LP

We do the same for the constraint:

```

MPConstraint c0 = solver.makeConstraint(-infinity, 100.0);
c0.setCoefficient(x1, 1);
c0.setCoefficient(x2, 1);
c0.setCoefficient(x3, 1);

MPConstraint c1 = solver.makeConstraint(-infinity, 600.0);
c1.setCoefficient(x1, 10);
c1.setCoefficient(x2, 4);
c1.setCoefficient(x3, 5);

MPConstraint c2 = solver.makeConstraint(-infinity, 300.0);
c2.setCoefficient(x1, 2);
c2.setCoefficient(x2, 2);
c2.setCoefficient(x3, 6);

```

Figure 3.6 – Java code to define the constraints of the LP

After setting all the constraints and variables we can execute the solver:

```

final MPSolver.ResultStatus resultStatus = solver.solve();

```

Figure 3.7 – Java code for running the solver

After executing the solver, we display the optimal value of the objective and the value of the variables:

```

System.out.println("Optimal objective value = " + solver.objective().value());
System.out.println("x1 = " + x1.solutionValue());
System.out.println("x2 = " + x2.solutionValue());
System.out.println("x3 = " + x3.solutionValue());

```

Figure 3.8 – Java code for displaying the result of the solver

Once you run the program, you will have the following the result:

```

Optimal objective value = 733.3333333333
x1 = 33.33333333336
x2 = 66.66666666666
x3 = 0.0

```

### 3.3 Research

A method of multicriteria analysis is chosen depending on the circumstances of the decision making.

UTA makes possible the estimation of a nonlinear additive function, which is obtained by the use of a linear program and the only information required from the decision maker are the global preferences between projects.

The value  $\delta$  must not be given too high an initial value.

After the execution of the algorithm we should always be possible to converse with the decision maker in order to specify the precision of the preferences stated

In the basic model, it was noted that the values given to  $\delta$  were to some extent arbitrary. UTAMP1 maximize  $\delta$ .

Variants of UTA

1. UTA
2. UTASTAR
3. UTA2
4. UTAMKEN
5. UTAMP1
6. UTAMP2
7. UTAMIME
8. UTASTARMIME
9. UTA2MIME

Should i make a comparison table ??

Disadvantages

1. We can always question the decision marker about his preferences
2. Solution may not be the only one as in any LP we can have different solutions

Chapter 4

Conclusion

# Bibliography

- [1] GRECO S., EHRGOTT M., RUI FIGUEIRA J., 2004, *Multiple Criteria Decision Analysis*
- [2] SISKOS J., 1979, *Cahier du LAMSADE n24 : Les programmes UTA*
- [3] SISKOS J., YANNAKOPOULOS D., 1982, *Cahier du LAMSADE n49 : Amélioration de la méthode UTA par introduction d'une double fonction d'erreurs*
- [4] HAMMAMI A., 2003, *Thèse : Modélisation technico-économique d'une chaîne logistique dans une entreprise réseau*
- [5] LUO H., SUN Z., 2014, *A study on Stock Ranking and Selection Strategy Based on UTA Method under the Condition of Inconsistence*
- [6] ZOPOUNIDIS C., DOUMPOS M., 1997, *A multicriteria decision aid methodology for the assessment of country risk*
- [7] EHSANIFAR M., ESHLAGHI A., KERAMATI M., NAZEMI J., 2013, *The Development of UTASTAR Method in Fuzzy Environment for Supplier Selection*
- [8] SISKOS J., 1983, *Application de la méthode UTA à un problème de sélection de points de vente mettant en jeu des critères multiples*

# List of Figures

3.1	Error function in UTASTAR . . . . .	7
3.2	Library used in the solver . . . . .	9
3.3	Java code to define the solver . . . . .	10
3.4	Java code to define the variables of the LP . . . . .	10
3.5	Java code to define the objective of LP . . . . .	10
3.6	Java code to define the constraints of the LP . . . . .	10
3.7	Java code for running the solver . . . . .	10
3.8	Java code for displaying the result of the solver . . . . .	11



# Appendices

# Annexe A

## Compte Rendus avec M. Cailloux

### A.1 CR1 - 15 juin 2017

15 juin 2017 / 15 :00 / Université Paris Dauphine

#### Participants

Olivier Cailloux, Elie Daher

#### Déroulement de la réunion

- Explication du plan du stage
- J'ai présenté les tâches effectuées
- J'avais quelques questions sur la méthode UTA et plus précisément sur un exemple "A Numerical Example" du chapitre 9 : "UTA Method" du livre : "Multiple Criteria Decision Analysis"
- Présentation des tâches à effectuer

#### Les tâches à faire

- Créer un repository sur github
- Ecrire un résumé sur la méthode UTA en utilisant LaTeX
- En parallèle, effectuer des recherches sur des littératures existantes

#### Pièces jointes

- First iteration : document présenté par M. Cailloux qui contient le plan de la première itération

La réunion a été levée à 15 :30

La prochaine réunion est prévue pour le mercredi 21 juin 2017 à 14 :00

## A.2 CR2 - 21 juin 2017

21 juin 2017 / 14 :00 / Université Paris Dauphine

### Participants

Olivier Cailloux, Elie Daher

### Déroulement de la réunion

- Présentation du travail effectué
- Résolution du projet java sur le programme linéaire

### Les tâches à faire

- Rendre le document plus détaillé et développé
- Rendre le document plus compréhensible pour les lecteurs qui n'ont pas un context décisionnel
- Fixer le programme linear programming sur Java

La réunion a été levée à 14 :40

La prochaine réunion est prévue pour le mercredi 28 juin 2017 à 11 :00

## A.3 CR3 - 28 juin 2017

28 juin 2017 / 11 :00 / Université Paris Dauphine

### Participants

Olivier Cailloux, Elie Daher

### Déroulement de la réunion

- Présentation du travail effectué
- Citer les remarques et les modifications à réaliser sur le document summary-uta et sur le repository de github (enlever le folder qui contient les \*.class)
- Présentation du travail à effectuer pour la semaine prochaine

### Les tâches à faire

- Corriger les remarques sur le document
- Développer  $v^R$  et  $v^T$
- Effectuer une recherche sur les littératures similaires déjà réalisées

La réunion a été levée à 11 :30

La prochaine réunion est prévue pour le mercredi 5 juillet 2017 à 10 :00

## A.4 CR4 - 18 juillet 2017

18 juillet 2017 / 10 :00 / Université Paris Dauphine

### Participants

Olivier Cailloux, Elie Daher

### Déroulement de la réunion

- Présentation du travail effectué
- Explication de la fusion entre le projet Alternative-Criteria et UTA
- Apprendre comment télécharger des littératures à partir du site de la BU de Dauphine

### Les tâches à faire

- Elaborer un document qui regroupe tout le travail effectué
- S'approfondir dans la recherche des littératures
- Fixer la méthode GenerateNumbers

La réunion a été levée à 10 :45

La prochaine réunion est prévue pour le mardi 25 juillet 2017 à 10 :00

## A.5 CR5 - 25 juillet 2017

25 juillet 2017 / 10 :00 / Université Paris Dauphine

### Participants

Olivier Cailloux, Elie Daher

### Déroulement de la réunion

### Les tâches à faire

La réunion a été levée à

La prochaine réunion est prévue pour le 2017 à

## Annexe B

# Example of UTASTAR - Analyzing the choice of transportation

A DM wants to analyse the choice of transportation. The DM is interested in the following criteria

1. price
2. time (min)
3. comfort (possibility to have a seat)

The evaluation of the previous criteria is presented in the following table :

Means of transportation	Price	Time	Comfort	Ranking of the DM
RER	3	10	+	1
METRO (1)	4	20	++	2
METRO (2)	2	20	0	2
BUS	6	40	0	3
TAXI	30	30	+++	4

DM's preferences :  $RER \succ Metro1 \approx Metro2 \succ Bus \succ Taxi$

First of all, we should specify the scale <sup>1</sup> for each criteria.

- Price  $\rightarrow [30, 16, 2]$
- Time  $\rightarrow [40, 30, 20, 10]$
- Comfort  $\rightarrow [0, +, ++, +++]$

According to this formula :  $v(g(a)) = \sum_{i=1}^n v_i(g_i(a))$  , the value of each alternative may be written :

- $v[g(RER)] = 0.07v_1(16) + 0.93v_1(2) + v_2(10) + v_3(+)$
- $v[g(METRO1)] = 0.14v_1(16) + 0.86v_1(2) + v_2(20) + v_3(++)$
- $v[g(METRO2)] = v_1(2) + v_2(20) + v_3(0) = v_1(2) + v_2(20)$
- $v[g(BUS)] = 0.29v_1(16) + 0.71v_1(2) + v_2(40) + v_3(0) = 0.29v_1(16) + 0.71v_1(2)$
- $v[g(TAXI)] = v_1(30) + v_2(30) + v_3(+++) = v_2(30) + v_3(+++)$

We have that  $v_1(30) = v_2(40) = v_3(0) = 0$ .

Since the marginal value  $u_i(g_i)$  can be expressed in terms of variables  $w_{ij}$  :  $u_i(g_i^j) = \sum_{t=1}^{j-1} w_{it}$  , the value of each alternative can be written :

- $v[g(RER)] = w_{11} + 0.93w_{12} + w_{21} + w_{22} + w_{23} + w_{31}$
- $v[g(METRO1)] = w_{11} + 0.86w_{12} + w_{21} + w_{22} + w_{31} + w_{32}$
- $v[g(METRO2)] = w_{11} + w_{12} + w_{21} + w_{22}$
- $v[g(BUS)] = w_{11} + 0.71w_{12}$
- $v[g(TAXI)] = w_{21} + w_{31} + w_{32} + w_{33}$

For each pair of consecutive alternatives, we express the difference between them :

- $\Delta(RER, METRO1) = 0.07w_{12} + w_{23} - w_{32} \geq \delta$
- $\Delta(METRO1, METRO2) = -0.14w_{12} + w_{31} + w_{32} = 0$
- $\Delta(METRO2, BUS) = 0.29w_{12} + w_{21} + w_{22} \geq \delta$
- $\Delta(BUS, TAXI) = w_{11} + 0.71w_{12} - w_{21} - w_{31} - w_{32} - w_{33} \geq \delta$

Having  $\delta = 0.05$ , we can solve the following LP :

Objective :

$$\text{Minimize } \sum_{a \in A} \sigma_a^+ + \sigma_a^- \quad (\text{B.1})$$

Subject to :

$$\begin{cases} 0.07w_{12} + w_{23} - w_{32} - \sigma_{RER}^+ + \sigma_{RER}^- + \sigma_{METRO1}^+ - \sigma_{METRO1}^- \geq \delta \\ -0.14w_{12} + w_{31} + w_{32} - \sigma_{METRO1}^+ + \sigma_{METRO1}^- + \sigma_{METRO2}^+ - \sigma_{METRO2}^- = 0 \\ 0.29w_{12} + w_{21} + w_{22} - \sigma_{METRO2}^+ + \sigma_{METRO2}^- + \sigma_{BUS}^+ - \sigma_{BUS}^- \geq \delta \\ w_{11} + 0.71w_{12} - w_{21} - w_{31} - w_{32} - w_{33} - \sigma_{BUS}^+ + \sigma_{BUS}^- + \sigma_{TAXI}^+ - \sigma_{TAXI}^- \geq \delta \\ w_{11} + w_{12} + w_{21} + w_{22} + w_{23} + w_{31} + w_{32} + w_{33} = 1 \end{cases} \quad (\text{B.2})$$

So by using the com.google.ortools library, we can solve the Linear Program above with  $\sigma = 0.05$ .

This Linear Program solution is coded in Java class ChoiceTransportation.

By executing the class ChoiceTransportation, you will have the following result :

An optimal solution is  $w_{11} = 0.5$ ,  $w_{22} = 0.05$ ,  $w_{23} = 0.05$ ,  $w_{33} = 0.4$  with  $\sum_{a \in A} \sigma_a^+ + \sigma_a^- = 0$ . The utilities found for each alternative are as follows :

- $v(g(RER)) = 0.6$

---

<sup>1</sup>the interval  $[g_{i*}, g_i^*]$  is cut into equal intervals



- $v(g(METRO1)) = 0.55$
- $v(g(METRO2)) = 0.55$
- $v(g(BUS)) = 0.5$
- $v(g(TAXI)) = 0.4$

Those utilities are consistent with the DM's preference ranking.