

**Louvain School of Management**

# **Design and implementation of a course scheduling system using Random Search**

Author : Jeanne Marquagnies  
Supervisor(s) : Pr. Daniele Catanzaro  
Academic year 2019-2020

## **Acknowledgments**

I would like to express my deepest appreciation to Professor D. Catanzaro who supervised me during this master thesis. I would like to thank him for his many practical and theoretical advice, for his availability and for his infectious passion for teaching. With the hope that we could collaborate again on other projects.

I would also like to thank F. Seha, my mother, for her capacity to encourage people to surpass themselves, for her patience, for her professional and critical look at my work and for these five years at university.

I would like to thank N. Maton, my mother-in-law, too, for carrying out the necessary corrections this work could need.

Finally, I would like to say thank you to UCLouvain FUCaM Mons and UCLouvain, especially the Louvain School of Management, and their educational teams to allow me to acquire the knowledge and the expertise to conduct this modest work.

## Abstract

Scheduling is a discipline that has always existed in many domains. This work is focused on a particular context: education at university level. The goal of course scheduling is to assign courses, also known as events, into a timetable formed of time slots, under a set of constraints.

The University Course Timetabling Problem (UCTP) is a NP-hard problem. This means that it is not possible to solve it exactly in practical time.

Carter and Laporte (1998) exposed different methods to solve different types of UCTPs, including Course Timetabling, Class-Teacher Timetabling, Student Scheduling, Teacher Assignment and Classroom Assignment.

In 2012, Feizi-Derakhshi, Babei and Heidarzadeh provided a classification of methods to solve the UCTP, including Operational Research methods (Graph Colouring, Integer Programming, Constraint Satisfaction Programming), meta-heuristics (Evolutionary and Genetic Algorithms, Ant Colony Algorithms, Hill Climbing, Tabu Search, Simulated Annealing) and modern intelligent methods (hybridisation, hyper heuristics methods). Those methods will be reviewed in this paper, so as to present the current state-of-the-art on UCTP.

Random Search is not a method that has been tested a lot. This procedure performs a swap between a course and a block of Time slots available to welcome the selected course. Both the course and its destination are randomly selected. To evaluate the evolution of the results, a penalty function has been set up. This function evaluates six constraints. Four of them should absolutely be respected: no teacher can have simultaneous courses, no students can have simultaneous courses, no teacher can have courses on both campuses on the same day, no courses requiring a specific room can be assigned to another room than the required one. These constraints are characterized as hard constraints. Their cost should equal zero at the end of the computation of the model. For the two last ones (avoiding courses late in the day and during lunchtime and fitting the capacity of the classroom with the number of students enrolled in the course assigned to that class), the procedure tries to reduce their cost as much as possible with a predefined number of iterations. They are characterized as soft constraints.

This method is the core of this thesis and has been implemented at the UCLouvain, more specifically at the Louvain School of Management (the business engineering and management faculty of the university).

## Table of contents

|                                                            |    |
|------------------------------------------------------------|----|
| Acknowledgments.....                                       | I  |
| Abstract.....                                              | II |
| Introduction.....                                          | 3  |
| 1. The University Course Timetabling Problem (UCTP) .....  | 4  |
| 2. Review of the literature .....                          | 5  |
| 2.1. Mathematical Programming-based methods .....          | 6  |
| 2.2. Meta-heuristics .....                                 | 10 |
| 2.3. Other intelligent methods.....                        | 14 |
| 3. The Random Search .....                                 | 17 |
| 3.1. The UCTP instances of Mons and Louvain-la-Neuve ..... | 17 |
| 3.2. Hypotheses.....                                       | 17 |
| 3.3. Inputs.....                                           | 18 |
| 3.4. Outputs.....                                          | 18 |
| 3.5. Random Search and Room Assignment .....               | 19 |
| 3.6. Penalty function.....                                 | 20 |
| 3.7. Methodology.....                                      | 21 |
| 4. Results .....                                           | 24 |
| 5. Critical analysis .....                                 | 33 |
| Conclusion .....                                           | 35 |
| References .....                                           | 36 |
| Appendices .....                                           | 39 |
| <i>Appendix 1</i> – Courses data .....                     | 39 |
| <i>Appendix 2</i> – Rooms data .....                       | 54 |
| <i>Appendix 3</i> – Room timetable ‘Final_Sched’ .....     | 58 |

|                                                                                |           |
|--------------------------------------------------------------------------------|-----------|
| <i>Appendix 4 – Final timetables representation ‘Sched_by_year_Mons’ .....</i> | <i>59</i> |
| <i>Appendix 5 – Penalty costs.....</i>                                         | <i>60</i> |
| <i>Appendix 6 – Initial timetable creation Mosel code.....</i>                 | <i>60</i> |
| <i>Appendix 7 – Penalties calculation Mosel code.....</i>                      | <i>65</i> |
| <i>Appendix 8 - Random Search Mosel code .....</i>                             | <i>68</i> |
| <i>Appendix 9 – Tabu check Mosel code .....</i>                                | <i>70</i> |
| <i>Appendix 10 – Room assignment Mosel code.....</i>                           | <i>71</i> |
| <i>Appendix 11 – Display timetable by year Mosel code.....</i>                 | <i>72</i> |

## **Introduction**

Scheduling is a discipline that has always existed. We can meet this problem in a wide variety of fields like education, sports, or transportation (Gunawan et al., 2007). This work is focused on a particular context: education at university level. The goal of scheduling at university level is to assign courses, classes, exams, lessons or talks, also known as events, into a timetable formed of time periods, under a set of constraints (Kostuch, 2005). The university course timetabling problem has been studied for several decades. This is a problem that every university faces every year. And it is not an easy task. It can take up to several weeks to elaborate a feasible, acceptable and operational timetable that is suitable for everyone and that is conflict free.

At the present time, no magic bullet has been found yet. All methods already implemented in the literature provide good results, but none of them is universal and provides a hundred percent optimal and operational solution. Indeed, each university considers different constraints and they consider them at different levels of priority. This explains why a universal solution is hard to find.

The aim of this work is to provide a first approach or simply to demonstrate that an automated method can make administrative personnel's job easier in order to elaborate an operational timetable. It will allow revealing that such a method could save time, energy and money to universities. The final goal is not to create a full automated method but to provide a first approach and to determine if the method explored in this work was a good choice or not. The model that will be exposed was implemented in a simplified real case. It was developed for the Louvain School of Management, the business engineering and management faculty of UCLouvain.

This work is divided into five chapters. The first one will provide a definition of the university course timetabling problem. Chapter two will give a review of what has been already done in this domain. It is a review of the literature. Afterwards, chapter three will expose the approach explored for the real case and its development. Chapter four will present the results and their analysis. And finally, chapter five will present a quick criticism of this work.

## 1. The University Course Timetabling Problem (UCTP)

Schedule elaboration is a challenge faced each year by university administrative personnel. In this chapter, the UCTP will be characterized and defined.

One way to classify problems is by means of their inherent complexity of being solved. We identify two types of complex problems: P problems and NP-hard problems. P problems are problems that are easily solved with an exact algorithm, which counts a polynomial number of operations to solve a problem. NP-hard problems are problems which generally require an exponential number of operations to be solved exactly. Because solving them exactly is time-consuming, we generally use heuristics (Vandaele, 2018; Cruz-Chávez et al., 2016), some of which will be described later in this work.

The UCTP is a NP-hard problem. A solution to this problem is achievable by using a combination of several methods, without guarantying to reach the optimal solution (Al-Milli, 2010; Schaerf, 1999). Thus, this problem is very hard to solve, and even harder when the problem size increases.

Timetabling problems differ from one university to another, but authors agree that the UCTP can be defined as versions of an assignment problem. It consists in scheduling a set of courses to a predefined set of rooms and time slots, under a set of constraints (Kostuch, 2005; Schaerf, 1999; Abdennadher & Marté, 1998). Carter and Laporte (1999), for instance, include the addition of other assignment operations. They view course timetabling as a multi-dimensional assignment problem. First, teachers are assigned to courses and students to courses, course sections or classes. And then courses, also called events, are assigned to rooms and time periods.

The set of constraints to satisfy can be divided into two types: soft constraints and hard constraints. Hard constraints are the ones that must be satisfied by any means. For the soft constraints, we must minimize their violation. Consequently, hard constraints have a higher priority than soft ones (Al-Milli, 2010).

## 2. Review of the literature

In this chapter, a non-exhaustive review of the literature about UCTP approaches will be exposed. A lot of searches have been undertaken since the 1960s.

Carter and Laporte (1998) exposed several approaches for different UCTPs. Indeed, they considered that the UCTP could be classified in categories requiring different ways to solve them:

- Course Timetabling: the aim is to assign courses or course sections to time slots, under a set of constraints ;
- Class-Teacher Timetabling: the aim is to ensure the meeting between teachers and students, since teachers have already been assigned to courses and classes. This way mostly applies to high schools ;
- Student Scheduling: the aim is to assign students, that already have selected the courses they want to follow, to sections and to provide a balanced section sizes and respecting room capacities schedule ;
- Teacher Assignment: the aim is to assign teachers to courses by maximizing a preference function ;
- Classroom Assignment: the aim is to assign events to a room by ensuring that the size of the room can host all the students enrolled for the course.

For each category, the authors displayed several solving methods in a table, such as Tabu Search, Genetic and Evolutionary Algorithms, Integer Programming, Constraint Satisfaction Programming, Network Flow, Simulated Annealing, Branch-and-Bound, and other hybrid methods.

If we consider the current problem addressed in this work, we can classify it in two of these categories: Course Timetabling and Classroom Assignment. Indeed, students and teachers are already assigned to courses. Consequently, the aim of this work is to assign courses to the best fitting classrooms and the best Time slots.

In 1999, Schaerf also provided a list of techniques to solve the UCTP. He suggested techniques like Graph Colouring, Integer Programming, Network Flow, Tabu Search, Rule-based Approach, Constraint Logic Programming, and so on.

In 2015, Ilis et al. listed the same methods as above; but they also wrote about Particle Swarm Optimization, or Ant Colony Optimization. In their paper, they implemented

Artificial Neural Networks and Clustering on a real case in order to find a good method to solve the UCTP.

In 2012, Feizi-Derakhshi, Babei and Heidarzadeh provided a good survey of approaches to solve that problem and they presented the ones most frequently applied. In this chapter, approaches, methods and techniques will be exposed in accordance with their way of thinking. They categorized methods in 3 groups: Operational Research methods, meta-heuristics and modern intelligent methods.

Only a few kinds of methods per group will be introduced. Techniques from the authors mentioned above will be gathered. Obviously, there exist other methods but here are the most known and the most implemented.

## 2.1. Mathematical Programming-based methods

Operational Research gathers a set of methods, techniques and approaches together in order to make the best possible choice when a problem comes up. Those tools became essential to expose complex systems (Vandaele, 2018). In this section, Graph Colouring theory, Integer Programming methods and Constraint Satisfaction Programming-based methods will be introduced followed by some real case implementations.

### 2.1.1. Graph Colouring theory

In the Graph Colouring method, we model a timetabling problem by using a non-directional graph,  $G=(V,E)$ . The aim is to assign a Time slot to each course under a set of constraints. The Time slots are depicted by the colours, the courses by the nodes/vertices and the constraints by the edges. With this method, we try to colour the graph by using the least number of colours and by attempting to avoid neighbour vertices (with an edge between them) having the same colour (*Figure 1*).

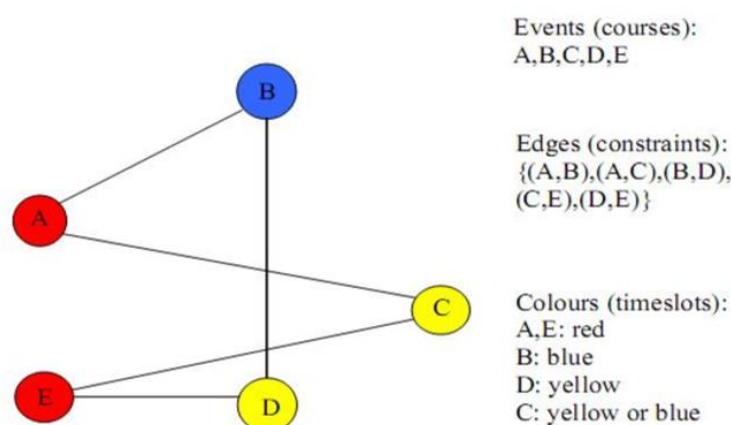


Figure 1 - The UCTP displayed with the Graph Colouring method (Feizi-Derakhshi, 2012).

The constraints can be understood as courses that cannot not be scheduled at the same time. For instance, the edge (A,B): A and B are two courses that cannot be scheduled at the same time because that is the same teacher teaching those courses or because these are courses for students of the same year or the same group. So, they must be coloured differently to ensure to be free of conflict (Feizi-Derakhshi et al., 2012 ; Schaerf, 1999).

### 2.1.2. Integer Programming Methods (IP)

IP is a Mathematical Programming-based method. With this kind of method, the goal is to try to maximise the preferences/interests and to minimise the cost in order to find a schedule that assigns courses to classrooms and time slots.

In order to provide a valid IP formulation for the UCTP we introduce the following notation and definitions that will be used throughout the chapter. Specifically, let:

- $C=\{c_1,...,c_c\}$  denote the set of courses with a fixed number of hours per week;
- $R=\{r_1,...,r_r\}$  denote the set of classrooms. To each element of the set is associated to a fixed capacity, a specific equipment and a location;
- $T=\{t_1,...,t_t\}$  denotes the set of teachers with their course assignment;
- $D=\{d_1,...,d_5\}$  denotes the set of days (from Monday to Friday);
- $H=\{h_1,...,h_h\}$  denotes the time periods of equal length (one hour) of one day;
- In some cases, there also exist some pre-assignments and forbidden periods for courses, rooms or teacher. In this case, they are not mentioned since no information could be obtained from the administrative services of the faculty.

In our situation, the objective function of the UCTP can be expressed as a minimisation of the penalties associated to six constraints. In order to give an IP formulation, we need to define the variables:

$$x_{c r}^{d h} = \begin{cases} 1 & \text{if course } c \text{ is given on day } d \text{ and starts at hour } h \text{ in room } r, \\ 0 & \text{otherwise.} \end{cases}$$

In addition of the input data, other elements are needed :

- $CT_k$  denotes the set of course of teachers  $t_k$ ;
- $d_c$  denotes the duration of course  $c$ ;
- $rc_i$  denotes the set of classrooms of campus  $i$ ;

- $Y_c$  denotes the year of course  $c$ ;
- $O_c$  denotes the option of course  $c$ ;
- $S_c$  denotes the section of course  $c$ ;
- $TR$  denotes the set of different types of classrooms;
- $R_a$  denotes the set of classrooms of type  $a$ ;
- $CR_a$  denotes the set of courses requiring a specific room of type  $a$ ;
- $m_a$  denotes the number of classrooms of type  $a$  available.

Then the problem can be formulated as follow:

$$\text{Min } f(x) = \omega_{st} * f_{st}(x) + \omega_t * f_t(x) + \omega_{camp} * f_{camp}(x) + \omega_{rr} * f_{rr}(x) + \omega_h * f_h(x) + \omega_{cap} * f_{cap}(x)$$

subject to

$$\sum_r \sum_d \sum_h x_{cr}^{dh} = 1, \quad \forall c \in C, \quad (1)$$

$$\sum_r \sum_{c \in Y_c \cap O_c \cap S_c} \sum_{\eta=h-d_c+1}^h x_{cr}^{d\eta} \leq 1, \quad \forall d \in D, \forall h \in H, \forall c \in C, \quad (2)$$

$$\sum_r \sum_{c \in CT_k} \sum_{\eta=h-d_c+1}^h x_{cr}^{d\eta} \leq 1, \quad \forall d \in D, \forall h \in H, \forall k \in T, \quad (3)$$

$$\sum_h \sum_c \sum_{rc_i}^{r_{cLLN}} x_{cr_i}^{dh} + \sum_h \sum_c \sum_{rc_j}^{r_{cMONS}} x_{cr_j}^{dh} \leq 1, \quad \forall d \in D, \forall k \in T, \quad (4)$$

$$\sum_c \sum_{\eta=h-d_c+1}^h x_{cr}^{d\eta} \leq 1, \quad \forall r \in R, \forall d \in D, \forall k \in T, \quad (5)$$

$$\sum_{r \in R_a} \sum_{c \in CR_a} \sum_{\eta=h-d_c+1}^h x_{cr_i}^{d\eta} \leq m_a, \quad \forall d \in D, \forall h \in H, \forall a \in TR \quad (6)$$

$$x_{cr}^{dh} \in \{0,1\}, \quad \forall c \in C, \forall r \in R, \forall d \in D, \forall h \in H, \quad (7)$$

Each term in the objective function encodes a specific soft constraint and ponderates it in function of its importance or desirability. In particular, the weight  $\omega_{st}$  encodes the desirability of avoiding conflicts for the students;  $\omega_t$  encodes the desirability of avoiding conflicts for the teachers;  $\omega_{camp}$  encodes the desirability of avoiding conflicts with the campuses;  $\omega_{rr}$  encodes the desirability of avoiding conflicts for the required rooms;  $\omega_h$  encodes the desirability of avoiding some hours; and  $\omega_{cap}$  encodes the desirability of assigning the best fitting room to the courses. Constraints (1)-(7) are the conditions to create a feasible timetable. (1) means that every course has to be assigned; (2) that no course of the same year, same section and same option can be given simultaneously with another course of the same year, section and option; (3) that no teacher can give two or more simultaneous courses; (4) that no teachers can give courses

of both campuses on the same day; (5) that no room can be assigned to two simultaneous courses; and (6) that the required room must be assigned to the course that requires it.

The UCTP can be described in different ways. A first way is to try to define groups of students that have selected a certain number of courses and then schedule them at different times. It can involve teacher assignment too. A second way is to organize courses and teachers based on available Time slots and classrooms. In this case, the goal is to minimize the dissatisfactions of students and teachers, or to maximize their preferences, under a set of constraints (Feizi-Derakhshi et al., 2012).

In his thesis, Siddharth (2015) uses that method to develop an initial schedule before using a Genetic Algorithm in order to improve his results. He used a multi-objective function. One of his objective functions was to maximize the preferences of students and teachers.

Another example is presented by Perera and Lanel (2016). After creating groups of students with a graph construction and applying a Graph Colouring algorithm to create a conflict matrix, they used an IP model to assign each group to a time period.

It is very hard to exactly solve such a formulation of the UCTP. Firstly, a lot of variables are considered. Secondly, some parts of the objective function can be non-linear. Thirdly, with the lack of information from the administrative services of both campuses of Mons and Louvain-la-Neuve, it is clearly impossible to exactly solve this problem that way. However, another method will be proposed later in this work.

### **2.1.3. Constraint Satisfaction Programming-based methods (CSPs)**

These methods can be described as a computing-based system. In this situation, constraints are possible values that a variable can take. Obviously, constraints also depend on a set of variables. Thus, the goal is to find values that can be assigned to variables in order to satisfy constraints imposed by the problem we attempt to solve (Feizi-Derakhshi et al., 2012).

In 2016, Cruz-Chávez et al. tested this method in a real case study, for the Faculty of Chemical Sciences and Engineering, in the city of Cuernavaca, in Mexico. They considered a set of variables to write eight constraints they wanted to satisfy with their model. A feasible schedule was obtained when courses were assigned to time periods, while respecting the eight constraints. They ran their model thirty times and only got twelve

feasible schedules. The time needed to perform the CSP and find those feasible schedules could vary between more or less three hours and a half and more or less six hours and a half.

## **2.2. Meta-heuristics**

Let's start by defining a heuristic. A heuristic is an inexact method that allows to provide a good solution, non-optimal but acceptable, to a problem that is difficult to solve. It allows to find solutions fast. A meta-heuristic is a high-level heuristic developed in order to find an approximate solution to a problem. This kind of method does not ensure to find an optimal solution but allows finding a good solution with less calculation efforts than exact methods. They are also defined as strategies which guide search processes. Actually, their aim is to effectively explore the solution space in order to find the best possible solution without being trapped by a local optimum. Those methods are mainly inspired by natural mechanisms (Vandaele, 2018; Fahmy, 2016; Eurodecision.com, n.d.).

### **2.2.1. Evolutionary and Genetic Algorithms**

Those kinds of algorithms come from Darwin's theories of natural selection and evolution. It efficiently combines a Random Search with information about previous iterations in order to guide the search where the best solutions are located. In this situation, we work with a population of solutions which evolves (Vandaele, 2018).

Genetic Algorithms (GAs) are a subset of Evolutionary Algorithms (EAs). EAs include three phases: 1) a selection of "elites"; 2) a regeneration, with crossover and mutation mechanisms on the selected individuals; and 3) a replacement of those new individuals in the initial population. Thus, GAs work this way. First, we must create an initial population randomly or by using a predefined technique. Then, we evaluate the quality of the individuals of this population using an objective function (single or multiple objectives). After that, we create a new generation of individuals by performing a selection of the best individuals in this population (parents); by applying a crossover on those parents to create offspring (children); by putting together those children with the parents selected previously; by applying a mutation process on this generation in order to ensure further exploration of solutions-space (*Figure 2*).

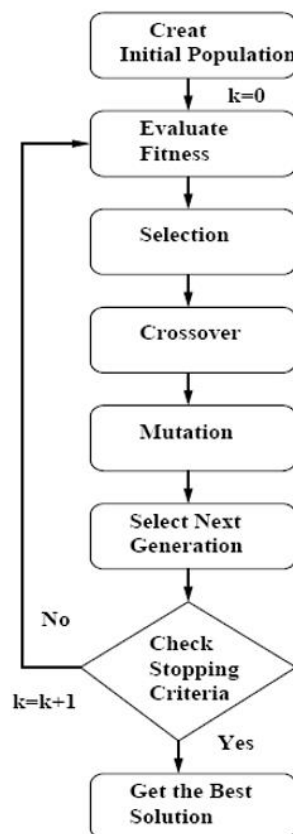


Figure 2 – GAs flow chart (Al-Milli, 2010)

This type of algorithms ends when the solution can reach a predefined value of the objective function or when the algorithm cannot provide any improvements after a certain number of iterations (Vandaele, 2018; Fahmy, 2016; Siddharth, 2015; Ni & Yang, 2013; Feizi-Derakhshi et al., 2012; Al-Milli, 2010; Schaerf, 1999; Paechter et al., 1998).

### 2.2.2. Ant Colony Algorithms

This method is inspired by ants' behaviour which place a track of pheromone on the ground to guide other ants because their movements are completely random. In other words, ants, located in the colony (point A), want to reach food (point B) but there are many pathways to reach it. To go to B, each ant randomly chooses a pathway. Once food has been reached, ants pick up food and go back to A. On their way back to the colony, they drop off pheromones on the ground. They all have a predefined quantity of pheromones to drop off. So, the longer the journey, the less pheromones are dropped off on the ground. After that, ants follow the pathway that releases the most pheromones, making pheromones disappear on longer pathways (Figure 3). This method used by ants allows to limit and guide the Random Search (Vandaele, 2018; Fahmy, 2016).

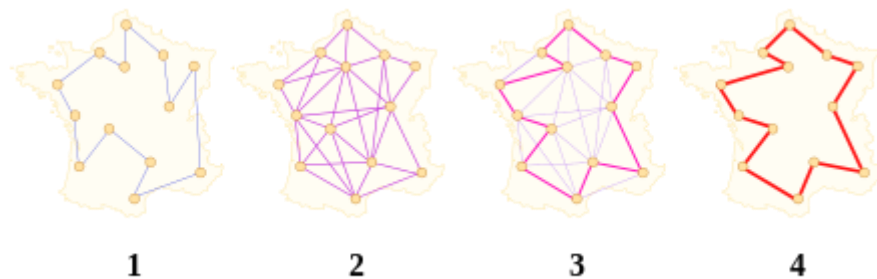


Figure 3 – Ant colony algorithm illustration (Vandaele, 2018)

It is even possible to apply this method to the UCTP. Masri and Ghaith implemented it in 2009 in combination with other methods: Simulated Annealing and Tabu Search. They performed an Ant Colony Algorithm in order to assign courses to Time slots (Feizi-Derakhshi et al., 2012).

These two first meta-heuristics are considered as population-based methods. These kinds of methods start from an initial population. At each iteration, a selection mechanism is used to select the best individuals. Then, changes are performed on those individuals in order to improve the solution. Finally, they are replaced in the initial population. This ends when we have reached a desirable solution.

The three following meta-heuristics are single solution-based methods. They differ from population-based methods because they do not start from an entire population to solve the problem but from a unique, single solution. Similar mechanisms are operated on the individual solution in order to improve the solution. This ends when the final condition is satisfied or when a predefined number of iterations is performed (Feizi-Derakhshi et al., 2012).

### 2.2.3. Local Search (Hill Climbing)

Local Search is a useful building heuristic to find a first solution. The idea is to try to improve a solution by exploring its neighbourhood. Hill Climbing is one example of Local Search. Starting from a feasible solution, let's explore its neighbourhood. If a neighbour is better than the current solution, we consider this neighbour as the new solution. We repeat this mechanism until no better solution can be found in the neighbourhood. An issue could be that we can easily be trapped by a local optimum (Vandaele, 2018; Feizi-Derakhshi et al., 2012).

Rossi-Daria et al. (2006) cited many implementations of Local Search approaches in the UCTP. Local Search is applied in many works about UCTP but, every time, it is combined with another method because Local Search is too simple to solve such a complex problem. For instance, in 2010, Joudaki, Imani and Mazhari implemented it with a Memetic Algorithm or with a GA for Shengxiang and Sadaf (Feizi-Derakhshi et al., 2012).

#### **2.2.4. Tabu Search**

Tabu Search is another example of Local Search and an improved Hill Climbing. It works like the latter: looking at the neighbours of the current solution. With Tabu Search, we replace the current solution by the best neighbour if it does not belong to the tabu list, whether the solution is better than the previous solution or not. Tabu list is updated at each new movement. This method allows avoiding cycling by discouraging from coming back to previously visited solutions. A memory mechanism is introduced to avoid stepping back to previous solutions. Sometimes, it is possible to override the tabu status of a solution: if the new solution allows a large improvement of the final result, the tabu status is dropped and the solution found is accepted as the new current solution (Vandaele, 2018; Feizi-Derakhshi et al., 2012; Schaerf, 1999; Hertz, 1990).

This method was used in a lot of papers. Alvarez-Valdes et al. (2002) used this method in a three phases course scheduling system. The aim of the Tabu Search was to improve the quality of the initial timetable constructed in the first phase. White and Zang (1998) also used a Tabu Search in order to improve the results they get. They combined it with a Constraint Logic and concluded that the use of a tabu list with constraint logic provided better results than using constraint logic alone.

#### **2.2.5. Simulated Annealing**

Simulated Annealing is a third example of Local Search. The basic idea of this method comes from the steel and glass industry, where steel or glass is heated up in order to make internal tensions disappear and then slowly cooled down to give it the desired shape. Applied to optimization, Simulated Annealing allows the rise, not only the decline, of the objective function at the beginning of the execution of the method. Step by step these increases are less and less allowed to finally stay in a minimum energy consuming state to solve the problem. As Tabu Search, this method allows to avoid being trapped in a local optimum (Vandaele, 2018; Feizi-Derakhshi et al., 2012).

In the schedule construction, Simulated Annealing works this way: first, we create an initial schedule. This schedule is set as the best schedule which is evaluated by its cost/penalty function. Then we define the initial temperature that will allow determining the probability to select each worse schedule in the neighbourhood of the current schedule. According to the probability formula, inspired by Boltzmann distribution, more the temperature is high, more the probability is high too. After setting the temperature, we enter a loop that ends according to a predefined criterion. This loop consists in randomly selecting a neighbour of the current schedule and evaluating its cost/penalty function. If the result of the neighbour is better than the current schedule, the neighbour becomes the current schedule. If that result is also better than the one of the best current schedule, the neighbour becomes the best schedule. Otherwise, if the result of the neighbour is worse than the current schedule, we randomly choose a value between 0 and 1 to compare it to the probability of this poorer schedule. If this value is lower than the probability, the neighbour becomes the current schedule. Finally, we reduce the temperature (Vandaele, 2018; Schaerf, 1999; Elmohamed et al., 1998). This mechanism that enables to visit worse schedule enables to avoid cycling and to look further away to find the best schedule.

Philipp Kostuch (2005) applied this method in a three-phase approach to the UCTP. The aim of the Simulated Annealing was to order time periods in an optimal way and to improve the quality of the schedule during a swap phase.

### **2.3. Other intelligent methods**

These methods include methods like combinational (hybrid) methods, fuzzy approach, hyper heuristic approaches or artificial intelligence. Only combinational methods and hyper heuristic methods will be reviewed in this section.

#### **2.3.1. Hybrid methods**

Hybridisation is the fact of mixing several methods in order to solve an optimization problem. They can be combined in an incredible number of ways. A first example is a combination by level. A low level of combination means that a method is a component of another method. A high level of combination means that the methods are structurally independent. Methods can also be used together for their mode. We can use them in a successive way: the output of a method becomes the input of another one; or in a cooperative way: methods work together and exchange data. Another combination can be

done by the type of the methods. They can be homogeneous if they are of the same nature or heterogeneous if they are different. Finally, they can be combined by their way of exploring the solution space: a global or partial exploration domain (Vandaele, 2018).

In the timetabling literature, this kind of approach is the most implemented in real case studies. They allow tackling such a complex problem more easily and more completely in order to reach a schedule that is close to an optimal solution. They seem very efficient and perform well in solving this kind of NP-hard problems. Methods are combined in order to efficiently and usefully use features of algorithms or techniques. Actually, previous methods applied alone are generally not efficient enough and does not allow having a complete and feasible schedule (Feizi-Derakhshi et al., 2012).

It is the case in Alvarez-Valdes et al.'s paper (2001). They first used a non-linear programming model in order to build an initial feasible schedule and then used a Tabu Search to improve it. Siddharth (2015) also combined methods to perform its schedule system: first a CSP is performed in order to create a population of feasible schedules and then he applied a GA to improve his results. In their paper, Gunawan, Ming NG and Leng POH (2007) used a hybrid algorithm which combines an IP approach, a Greedy heuristic and a modified Simulated Annealing algorithm sequentially. Or else, in 2016, Perera and Lanel exposed their process where they used a Graph Colouring algorithm to create a conflict matrix and then used it as an input for their IP. Most of the papers mentioned in the sections above describe hybrid methods. Indeed, using a single method is clearly not enough to find a complete, valid and feasible solution.

### **2.3.2. Hyper Heuristic approaches**

Hyper heuristic methods are high-level search methods. They allow choosing the best heuristic to be applied at each iteration in order to improve the results. At each iteration the approach uses a set of heuristics on the candidate and evaluates its results. Then it chooses to apply the best one and conserve its result.

Kendall and Hussin (2005) used this approach in order to solve examination and course timetabling problems. They implemented a Tabu Search-based hyper heuristic method. The method executed Tabu Search to define the best heuristic to process at each iteration. With another method they constructed an initial schedule. Then they did a loop that performed a Tabu Search to determine the heuristic to use. The first step was to consider

only heuristics not in tabu list. Then it chose the heuristic that best improved the schedule, applied it and made it tabu for future iterations. Afterwards it updated the solution and the tabu list. It came to an end when the condition was achieved.

As you might imagine there exist a huge number of ways to solve the UCTP. Moreover, each method mentioned above can be applied in several and different ways to find a feasible solution. UCTP is a NP-hard problem for which an exact and universal method does not already exist. Since the 1960s people have implemented Mathematical Programming-based methods, meta-heuristics and, more recently, other intelligent methods with the aim of solving that problem.

In the following chapter, another type of Local Search method will be exposed: the Random Search. It will be then applied to the instances of the Louvain School of Management (LSM), at UCLouvain, scheduling system.

### 3. The Random Search

This chapter will expose the current instances of the UCTP in Mons and Louvain-la-Neuve to elaborate course timetables. All along the chapter, we will express the hypotheses considered to develop that model; the inputs and their initial format; the final outputs; the approaches performed to develop an automated scheduling system at LSM; the penalty function and what it encompasses; and the methodology to create the model.

#### 3.1. The UCTP instances of Mons and Louvain-la-Neuve

Currently, timetable elaboration is a work done in a rudimentary way: by hands. Both in Mons and Louvain-la-Neuve, they use timetables of the previous year and adapt them to the new course organisation when new courses are integrated, and others are removed from programs. If a new teacher comes to teach a course, he or she must first adapt him- or herself to the preestablished timetable before considering changing the course period. In Mons, they even use transparencies and try to make matches between courses in order to avoid conflicts.

LSM has a list of 188 courses dispatched on two campuses (Mons and Louvain-la-Neuve). A teacher is already assigned to each course. 105 courses are given in the first semester, representing 495 hours to give, and 83 courses in the second semester, which represents 378 hours. In Mons, we count 20 available classrooms. In Louvain-la-Neuve, we count 155 classrooms. On both campuses, classrooms are open from 8 a.m. to 10 p.m.

#### 3.2. Hypotheses

We are going to consider one faculty of UCLouvain, LSM. On both campuses, classrooms are common for all faculties, so we do not consider courses of other faculties. This situation limits the efficiency of the model implemented.

We will consider a week of five days. Each day, rooms are available from 8 a.m. to 10 p.m. Time slots are divided in periods of one hour, so that we have to consider fourteen time slots per day.

The model does not consider the entirety of LSM courses. Actually, it does not consider courses that are not taught on regular basis (bimonthly or monthly courses). It does not take into account language courses because they are considered as practical courses that are not discussed in this model. Practical/exercise courses are organized after lecture

course timetables are validated. In addition, we only consider one teacher per course even if sometimes several teachers are assigned to a single course.

### **3.3. Inputs**

Input and output data are gathered together in an Excel file, called 'DataFile'. In this exercise we need information about two things: courses and classrooms.

Courses are listed in a sheet called 'Courses' (*Appendix 1*). This sheet resumes information about each course at LSM: its course code, a course ID, its name, its number of hours per week, the professor teaching it, the number of students enrolled, its semester, its year, its campus, its section, its option and its required room if necessary.

The other type of data needed to develop this automated model to schedule university courses is data about classrooms and their capacity. They are depicted in another sheet, called 'Rooms' (*Appendix 2*).

### **3.4. Outputs**

The final output of the model implemented is a 3-dimensional array. This array is a timetable of each room, for each day and each hour of a week. The final output is then a (class)room timetable. For each room we assign a course that fits best with its capacity and material.

Obviously, it is not possible to display a 3-dimensional array in an excel file. To do this, we transform the 3D array in a 2D array when the model is executed. At this point, we can have a representation of the timetable, in the sheet 'Final\_Sched' (*Appendix 3*). However, a room timetable is not an ideal way to present these results. As a student, it is preferable to display the result as a schedule of one's own year, section and option. Thus, we made a procedure to perform that operation and display them in the sheets 'Sched\_by\_year\_Mons' and 'Sched\_by\_year\_LLN' (*Appendix 4*).

### 3.5. Random Search and Room Assignment

The chosen method is a Random Search. It is a type of Local Search that is completely random. So, we work with a single solution-based method. In an initial schedule, we randomly select origin and destination points (which represent courses or block of hours of the same size). We swap those two blocks and evaluate if the schedule penalty function is improved. If it is the case, we save the current swap, otherwise we swap again the courses to go back to the initial situation. We iterate that operation 250.000 times (*Figure 4*).

---

```

initialize a timetable
count := 0;
repeat
    count +=1;
    random selection of an origin point
    random selection of a destination point (same size of the origin)
    swap(origin,destination)
    evaluate penalty function
    if better then save
    else swap(destination,origin)
    end-if
until count = 250.000
  
```

---

*Figure 4 – Random Search pseudo-code*

During the Random Search, we do not try to respect the required room assignment. So, after Random Search, penalties due to required rooms are still not equal to zero. Thus, we perform another procedure to assign courses involved by this assignment (*Figure 5*).

---

```

procedure RoomAssignment
    forall(i in 1..rooms) do
        forall(j in 1..days) do
            forall(k in 1..hours) do
                if course in (i,j,k) requires a specific room then
                    put course in (i,j,k) in the right room
                end-if
            end-do
        end-do
    end-do
end-procedure
  
```

---

Figure 5 – Room assignment pseudo-code (Mosel language)

### 3.6. Penalty function

To evaluate the schedule established at each stage, we use a penalty function. We try to reduce its cost at each iteration of Random Search. This penalty function can be considered as a list of constraints that we want to respect. Respecting them means that we want to associate a minimal cost to each of these constraints.

We can divide constraints into two sets: hard and soft constraints. Hard constraint must be respected at any cost. This means that they must be equal to zero at the end of the running time. In this model, four hard constraints are defined :

- Professor conflicts: a professor cannot be scheduled for several courses at the same time.
- Student conflicts: a student must attend every course of his/her year program, ensuring that he or she is free of conflict. For instance, this means that students from BAC1 cannot have several courses of BAC1 at the same time. We do not consider students straddling two different years.
- Campus conflicts: a teacher cannot be scheduled on both campuses on the same day.
- Required rooms: if a course requires a specific room, we must assign this course to it.

A feasible schedule must satisfy these four constraints. Consequently, their cost must equal zero.

Soft constraints can be violated but we must reduce their cost as much as possible. The following constraints are characterized as soft constraints :

- Capacity constraint: we try to assign the best fitting room to a course. If the room is too small to welcome a course a big penalty is associated. If it is too large, the penalty is the delta between the room capacity and the number of students enrolled to the course.
- Hours constraint: we try to avoid course sessions between 12 a.m. and 1 p.m., 6 p.m. and 8 p.m., and 8 p.m. and 10 p.m. Each time interval is associated with a different penalty cost.

The cost of each constraint is listed in *Appendix 5*. The penalty function is then the sum of each constraint penalty.

### 3.7. Methodology

There were various stages in order to create the program that runs to define a feasible timetable. In this section, those stages will be chronologically described.

Initially, the program was created to be implemented in the campus of Mons only. Mons campus encompasses less classrooms and courses. It was then easier to implement it there and above all it allowed having a much smaller running time and then see the results faster.

The first stage was to create an initial schedule. This schedule should not be feasible. The aim was to schedule all the courses in available rooms, while avoiding that a lecture course was spread out on several days. At first, we should consider that courses of more than four hours a week should be taught on several days. Then, we should ensure that a block of hours, representing a course lecture, was not spread out on two days because there was not enough time to finish the course that day. After that operation we had our initial timetable (*Appendix 6*).

The second stage was to create the procedures to calculate the penalty function (*Appendix 7*). According to the way the initial schedule was created, it seemed logical that its penalty function would be very high. Indeed, since it violated almost all the hard constraints and since all the courses were assigned to entire days before going to the next day, this initial schedule was not acceptable at all. But due to it and its penalty result we could start to develop the Random Search (*Appendix 8*) and Room Assignment (*Appendix 10*) procedures and finally display the results in a good way to understand them, in other words like timetables distributed to students (*Appendix 11*).

Since the results were good and encouraging, we implemented it to Louvain-la-Neuve campus. At that moment, we could realize that it took much more time to run the program only for Louvain-la-Neuve courses because many more classrooms had to be taken into account.

The following stage was to adapt the code to ensure that it works for Mons and Louvain-la-Neuve at the same time. It was easy to adapt but again we noticed that it took even more time to compute. Since the programming code was working and provided good results, we could start to find ways to reduce the running time of the program.

In order to reduce the running time, we first had to find what was slowing the program down. It turned out that it was because each penalty constraint was calculated individually, whereas we could gather several penalty constraints together. When this operation was done, we noticed that the running time decreased by more than half.

Now we had reduced the running time, we had to implement Tabu Lists during the Random Search. This operation aimed at avoiding making previous swaps again or using origin and destination points that were previously checked, and then it allowed exploring much better the space of potential solutions. In *Figure 6* the pseudo-code of the Random Search in which we added Tabu Lists is displayed (*Appendix 9*).

---

```

initialize a timetable
count := 0;
repeat
    count += 1;
    check := 0;
    while(check = 0) do
        random selection of an origin point
        random selection of a destination point (same size of the origin)
        if origin is in origin tabu list OR destination is in destination tabu list then
            check = 0;
        else check = 1;
        end-if
    end-do
    update tabu lists with the current selected points
    swap(origin,destination)
    evaluate penalty function
    if better then save
    else swap(destination,origin)
    end-if
until count = 250.000

```

---

*Figure 6 – Random Search with a Tabu List pseudo-code*

The final stage of this model development was to find a comparison point. This comparison was a penalty calculation of the current UCTP instances published by the SAFI, the students' service in the campus of Mons. To do so, we had to translate the current schedules so that they could be read by the program. Each year, section and option schedules were manually transformed to enter in the room timetable. After that, the procedure to calculate the penalty function was applied.

This chapter described the chronological sequence of designing and implementing a course scheduling system using Random Search for LSM, the UCLouvain business engineering and management faculty. It started with creating an invalid initial timetable to a feasible timetable, while reducing computing time and implementing tabu list concepts.

## 4. Results

Now the model has been described and implemented, it is time for results. In this chapter, the results of thirty computations for both semesters will be exposed and analysed. There will also be a review of the results from the UCTP instances of Mons that are currently applied compared to the Random Search method implemented on Mons campus only.

First the results will be exposed after thirty computations of the model. The model was implemented in Mosel with the help of the optimisation tool Fico Xpress. The results are displayed in *Table 1* and *Table 2*.

| Run | Running time | Initial |         | Optimized |        | Final  |        |
|-----|--------------|---------|---------|-----------|--------|--------|--------|
|     |              | Q1      | Q2      | Q1        | Q2     | Q1     | Q2     |
| 1   | 5h18'        | 141.254 | 105.089 | 88.557    | 78.861 | 84.992 | 78.935 |
| 2   | 6h02'        | 141.254 | 105.089 | 91.556    | 80.936 | 91.592 | 82.465 |
| 3   | 6h01'        | 141.254 | 105.089 | 90.759    | 76.498 | 88.442 | 76.058 |
| 4   | 5h33'        | 141.254 | 105.089 | 88.722    | 79.804 | 81.229 | 80.936 |
| 5   | 5h33'        | 141.254 | 105.089 | 94.198    | 80.141 | 89.970 | 77.431 |
| 6   | 6h30'        | 141.254 | 105.089 | 90.580    | 78.953 | 86.137 | 76.241 |
| 7   | 6h35'        | 141.254 | 105.089 | 91.265    | 78.655 | 92.103 | 79.295 |
| 8   | 6h44'        | 141.254 | 105.089 | 100.722   | 81.477 | 96.215 | 82.002 |
| 9   | 6h43'        | 141.254 | 105.089 | 88.023    | 78.363 | 83.094 | 81.718 |
| 10  | 8h06'        | 141.254 | 105.089 | 90.275    | 79.640 | 85.204 | 80.118 |
| 11  | 8h06'        | 141.254 | 105.089 | 90.106    | 81.608 | 84.066 | 79.021 |
| 12  | 6h25'        | 141.254 | 105.089 | 92.767    | 78.785 | 88.735 | 79.726 |
| 13  | 6h26'        | 141.254 | 105.089 | 94.365    | 77.210 | 87.963 | 77.437 |
| 14  | 7h13'        | 141.254 | 105.089 | 90.232    | 80.700 | 85.038 | 81.539 |
| 15  | 7h13'        | 141.254 | 105.089 | 87.215    | 80.054 | 80.598 | 81.717 |
| 16  | 6h38'        | 141.254 | 105.089 | 93.289    | 80.486 | 86.698 | 79.072 |
| 17  | 6h38'        | 141.254 | 105.089 | 92.181    | 81.942 | 88.023 | 85.115 |
| 18  | 6h31'        | 141.254 | 105.089 | 90.759    | 79.563 | 84.653 | 79.345 |
| 19  | 6h36'        | 141.254 | 105.089 | 93.205    | 79.560 | 89.053 | 76.911 |
| 20  | 7h35'        | 141.254 | 105.089 | 94.690    | 80.649 | 92.136 | 80.775 |
| 21  | 7h34'        | 141.254 | 105.089 | 92.739    | 79.808 | 87.621 | 81.846 |
| 22  | 6h50'        | 141.254 | 105.089 | 91.371    | 80.467 | 90.052 | 80.936 |
| 23  | 6h49'        | 141.254 | 105.089 | 89.410    | 80.610 | 84.994 | 80.043 |
| 24  | 7h21'        | 141.254 | 105.089 | 94.349    | 79.239 | 89.942 | 74.929 |
| 25  | 7h16'        | 141.254 | 106.489 | 93.084    | 81.288 | 87.816 | 82.292 |
| 26  | 6h21'        | 141.254 | 105.089 | 90.075    | 81.327 | 83.434 | 83.780 |
| 27  | 6h20'        | 141.254 | 105.089 | 89.451    | 78.364 | 86.266 | 76.889 |
| 28  | 6h16'        | 141.254 | 105.089 | 94.006    | 79.418 | 86.766 | 80.488 |
| 29  | 6h15'        | 141.254 | 105.089 | 96.192    | 81.882 | 90.297 | 81.989 |
| 30  | 5h21'        | 141.254 | 105.089 | 89.262    | 78.621 | 86.063 | 79.837 |

Table 1 – Total results for thirty computations of the model

The first thing we can notice is that each computation ends with a different result. This is caused by the stochastic characteristic of a Random Search. Four results are highlighted in the table above: two in orange/red and two in green. They respectively represent the worst and best results from the thirty computations of both semesters. For the first semester, the results vary from a penalty cost of 80.598 to a penalty cost of 96.215. For the second semester, the results vary from a penalty cost of 74.929 to a penalty cost of 85.115. We can note that the first semester results are nearly all the time worse than those from the second semester. It is partially due to the fact that in the second semester there

are less courses, and then less hours, to schedule and then less opportunities of conflicts and penalties generations.

Another thing that we can realize is that the ‘optimized’ result is sometimes better than the final result. This is even the case for the second semester worst result. This happens when during Random Search, the courses requiring a specific room have been assigned to a classroom that fits better their number of enrolled students than the required room. In addition, for most of these courses, the specific required room is too small to welcome the number of enrolled students. This results in higher capacity penalties after assigning these courses to their specific class.

When analysing *Table 2*, we can also observe that on thirty computations it was not possible to reach a feasible solution every time since the required room constraint was not respected. These timetables could be solved or repaired manually to be applied and operational but the aim of this work is to find an automated method that would help administrative personnel to save time and energy with the scheduling task which, for now, takes several days or weeks and is done by hands. Despite this, the other hard constraints are all the time respected which ensures a feasible timetable.

Consequently, we can consider that the results are satisfying since it allows creating a feasible timetable 23 times for the first semester (in 77% of cases) and 29 times for the second semester (in 97% of cases). Unfortunately, even if they are feasible they are not acceptable. Too many courses stay very late in a day, in order words between 8 p.m. and 10 p.m. To avoid this, it would have been necessary to increase the number of iterations to several millions or to change the condition to stop Random Search. It would explore much better the solution space but would take much more time. For now, Random Search computes 250.000 iterations and takes between five and eight hours to create a timetable for both semesters. For both first and last runs the duration of the code computation is about five hours. They are the lowest durations. It is partially because those computations were performed alone. For the other computations, two runs were computed at the same time.

| Run | Running time     | Penalty function        | Details          | Penalties               | Initial |         | Optimized |        | Final  |        |
|-----|------------------|-------------------------|------------------|-------------------------|---------|---------|-----------|--------|--------|--------|
|     |                  |                         |                  |                         | Q1      | Q2      | Q1        | Q2     | Q1     | Q2     |
| 1   | 5h18'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 88.557    | 78.861 | 84.992 | 78.935 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 74.607    | 71.986 | 83.042 | 77.060 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 450       | 350    | 450    | 350    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 800     | 675     | 800       | 675    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 700     | 850     | 700       | 850    |        |        |
| 2   | 6h02'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 91.556    | 80.936 | 91.592 | 82.465 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 77.531    | 73.436 | 89.567 | 79.965 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 450       | 475    | 450    | 475    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 825     | 1.025   | 825       | 1.025  |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 750     | 1.000   | 750       | 1.000  |        |        |
| 3   | 6h01'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 90.759    | 76.498 | 88.442 | 76.058 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 3.000  | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 76.359    | 69.423 | 83.042 | 73.983 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 475       | 300    | 475    | 300    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 975     | 825     | 975       | 825    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 950     | 950     | 950       | 950    |        |        |
| 4   | 5h33'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 88.722    | 79.804 | 81.229 | 80.936 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 74.847    | 72.454 | 79.354 | 78.586 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 550       | 550    | 550    | 550    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 825     | 750     | 825       | 750    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 500     | 1.050   | 500       | 1.050  |        |        |
| 5   | 5h33'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 94.198    | 80.141 | 89.970 | 77.431 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 2.000  | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 80.098    | 72.966 | 85.870 | 75.256 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 575       | 500    | 575    | 500    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 825     | 775     | 825       | 775    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 700     | 900     | 700       | 900    |        |        |
| 6   | 6h30'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 90.580    | 78.953 | 86.137 | 76.241 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 76.230    | 71.503 | 83.787 | 73.791 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 575       | 425    | 575    | 425    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 1.075   | 775     | 1.075     | 775    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 700     | 1.250   | 700       | 1.250  |        |        |
| 7   | 6h35'            | TOTAL                   |                  |                         | 141.254 | 105.089 | 91.265    | 78.655 | 92.103 | 79.295 |
|     |                  | Prof penalties          |                  | 100                     | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                     | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                     | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                    | 12.000  | 5.000   | 12.000    | 5.000  | 2.000  | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 93.589  | 77.290    | 71.730 | 88.128 | 77.370 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 750     | 450       | 425    | 450    | 425    |
|     | 6 p.m. - 8 p.m.  | 25                      | 1.875            | 1.450                   | 925     | 850     | 925       | 850    |        |        |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                   | 600     | 650     | 600       | 650    |        |        |

Table 2 – Results details for thirty computations of the model - run 1 to run 7

| Run | Running time     | Penalty function        | Details          | Penalties                | Initial |         | Optimized |        | Final  |        |
|-----|------------------|-------------------------|------------------|--------------------------|---------|---------|-----------|--------|--------|--------|
|     |                  |                         |                  |                          | Q1      | Q2      | Q1        | Q2     | Q1     | Q2     |
| 8   | 6h44'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 100.722   | 81.477 | 96.215 | 82.002 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 86.322    | 73.977 | 93.815 | 79.502 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 550       | 500    | 550    | 500    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 900       | 800    | 900    | 800    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 950     | 1.200   | 950       | 1.200  |        |        |
| 9   | 6h43'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 88.023    | 78.363 | 83.094 | 81.718 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 2.000  |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 73.673    | 71.513 | 80.744 | 77.868 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 475       | 400    | 475    | 400    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 1.025     | 850    | 1.025  | 850    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 850     | 600     | 850       | 600    |        |        |
| 10  | 8h06'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 90.275    | 79.640 | 85.204 | 80.118 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 76.550    | 72.640 | 83.479 | 78.118 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 450       | 450    | 450    | 450    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 825       | 750    | 825    | 750    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 450     | 800     | 450       | 800    |        |        |
| 11  | 8h06'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 90.106    | 81.608 | 84.066 | 79.021 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 76.081    | 74.283 | 82.041 | 76.696 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 425       | 525    | 425    | 525    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 900       | 850    | 900    | 850    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 700     | 950     | 700       | 950    |        |        |
| 12  | 6h25'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 92.767    | 78.785 | 88.735 | 79.726 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 3.000  | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 78.217    | 72.110 | 83.185 | 78.051 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 600       | 375    | 600    | 375    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 850       | 750    | 850    | 750    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 1.100   | 550     | 1.100     | 550    |        |        |
| 13  | 6h26'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 94.365    | 77.210 | 87.963 | 77.437 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 80.365    | 70.085 | 85.963 | 75.312 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 525       | 300    | 525    | 300    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 925       | 875    | 925    | 875    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 550     | 950     | 550       | 950    |        |        |
| 14  | 7h13'            | TOTAL                   |                  |                          | 141.254 | 105.089 | 90.232    | 80.700 | 85.038 | 81.539 |
|     |                  | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |                  | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |                  | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |                  | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |                  | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 76.132    | 73.725 | 82.938 | 79.564 |
|     |                  | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 500       | 475    | 500    | 475    |
|     |                  |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 650       | 700    | 650    | 700    |
|     | 8 p.m. - 10 p.m. | 50                      | 1.550            | 1.100                    | 950     | 800     | 950       | 800    |        |        |

| Run | Running time | Penalty function        | Details          | Penalties                | Initial |         | Optimized |        | Final  |        |
|-----|--------------|-------------------------|------------------|--------------------------|---------|---------|-----------|--------|--------|--------|
|     |              |                         |                  |                          | Q1      | Q2      | Q1        | Q2     | Q1     | Q2     |
| 15  | 7h13'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 87.215    | 80.054 | 80.598 | 81.717 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 73.715    | 72.504 | 79.098 | 79.167 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 625       | 475    | 625    | 475    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 625       | 925    | 625    | 925    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 250     | 1.150     | 250    | 1.150  |        |
| 16  | 6h38'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 93.289    | 80.486 | 86.698 | 79.072 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 79.364    | 72.861 | 84.773 | 76.447 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 450       | 500    | 450    | 500    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 825       | 875    | 825    | 875    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 650     | 1.250     | 650    | 1.250  |        |
| 17  | 6h38'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 92.181    | 81.942 | 88.023 | 85.115 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 78.156    | 74.742 | 85.998 | 82.915 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 500       | 450    | 500    | 450    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 775       | 850    | 775    | 850    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 750     | 900       | 750    | 900    |        |
| 18  | 6h31'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 90.759    | 79.563 | 84.653 | 79.345 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 76.584    | 72.263 | 82.478 | 77.045 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 500       | 550    | 500    | 550    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 775       | 850    | 775    | 850    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 900     | 900       | 900    | 900    |        |
| 19  | 6h36'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 93.205    | 79.560 | 89.053 | 76.911 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 79.005    | 72.310 | 86.853 | 74.661 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 600       | 450    | 600    | 450    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 950       | 850    | 950    | 850    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 650     | 950       | 650    | 950    |        |
| 20  | 7h35'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 94.690    | 80.649 | 92.136 | 80.775 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 2.000  | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 80.640    | 73.349 | 88.086 | 78.475 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 525       | 475    | 525    | 475    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 825       | 975    | 825    | 975    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 700     | 850       | 700    | 850    |        |
| 21  | 7h34'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 92.739    | 79.808 | 87.621 | 81.846 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 78.214    | 72.383 | 85.096 | 79.421 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 625       | 500    | 625    | 500    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 950       | 925    | 950    | 925    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 950     | 1.000     | 950    | 1.000  |        |
| 22  | 6h50'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 91.371    | 80.467 | 90.052 | 80.936 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 77.621    | 73.192 | 88.302 | 78.661 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 450       | 475    | 450    | 475    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 900       | 1.100  | 900    | 1.100  |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 400     | 700       | 400    | 700    |        |

| Run | Running time | Penalty function        | Details          | Penalties                | Initial |         | Optimized |        | Final  |        |
|-----|--------------|-------------------------|------------------|--------------------------|---------|---------|-----------|--------|--------|--------|
|     |              |                         |                  |                          | Q1      | Q2      | Q1        | Q2     | Q1     | Q2     |
| 23  | 6h49'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 89.410    | 80.610 | 84.994 | 80.043 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 2.000  | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 75.485    | 73.085 | 81.069 | 77.518 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 425       | 575    | 425    | 575    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 900       | 850    | 900    | 850    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 600     | 1.100     | 600    | 1.100  |        |
| 24  | 7h21'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 94.349    | 79.239 | 89.942 | 74.929 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 80.249    | 71.939 | 87.842 | 72.629 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 650       | 500    | 650    | 500    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 800       | 900    | 800    | 900    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 650     | 900       | 650    | 900    |        |
| 25  | 7h16'        | TOTAL                   |                  |                          | 141.254 | 106.489 | 93.084    | 81.288 | 87.816 | 82.292 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 1.400   | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 79.184    | 74.013 | 85.916 | 80.017 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 525       | 450    | 525    | 450    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 775       | 775    | 775    | 775    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 600     | 1.050     | 600    | 1.050  |        |
| 26  | 6h21'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 90.075    | 81.327 | 83.434 | 83.780 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 76.125    | 73.952 | 81.484 | 81.405 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 550       | 475    | 550    | 475    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 650       | 850    | 650    | 850    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 750     | 1.050     | 750    | 1.050  |        |
| 27  | 6h20'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 89.451    | 78.364 | 86.266 | 76.889 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 75.751    | 71.589 | 84.566 | 75.114 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 475       | 425    | 475    | 425    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 775       | 650    | 775    | 650    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 450     | 700       | 450    | 700    |        |
| 28  | 6h16'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 94.006    | 79.418 | 86.766 | 80.488 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 79.431    | 72.268 | 84.191 | 78.338 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 575       | 475    | 575    | 475    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 900       | 775    | 900    | 775    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 1.100   | 900       | 1.100  | 900    |        |
| 29  | 6h15'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 96.192    | 81.882 | 90.297 | 81.989 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 0      | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 82.167    | 74.482 | 88.272 | 79.589 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 475       | 425    | 475    | 425    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 800       | 975    | 800    | 975    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 750     | 1.000     | 750    | 1.000  |        |
| 30  | 5h21'        | TOTAL                   |                  |                          | 141.254 | 105.089 | 89.262    | 78.621 | 86.063 | 79.837 |
|     |              | Prof penalties          |                  | 100                      | 1.600   | 900     | 0         | 0      | 0      | 0      |
|     |              | Student Penalties       |                  | 100                      | 4.600   | 2.300   | 0         | 0      | 0      | 0      |
|     |              | Campus Penalties        |                  | 100                      | 0       | 0       | 0         | 0      | 0      | 0      |
|     |              | Required room Penalties |                  | 1000                     | 12.000  | 5.000   | 12.000    | 5.000  | 2.000  | 0      |
|     |              | Capacity Penalties      |                  | $\Delta$ or $\Delta^*10$ | 118.679 | 93.589  | 75.412    | 71.396 | 82.213 | 77.612 |
|     |              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                       | 950     | 750     | 475       | 525    | 475    | 525    |
|     |              |                         | 6 p.m. - 8 p.m.  | 25                       | 1.875   | 1.450   | 775       | 750    | 775    | 750    |
|     |              | 8 p.m. - 10 p.m.        | 50               | 1.550                    | 1.100   | 600     | 950       | 600    | 950    |        |

The results below do not mean anything if they cannot be compared to a real comparison point. That is the whole point of the following part. It was decided to use the schedules that are currently applied in the campus of Mons to do that experiment. To do so, we had to extract all the data from those schedules and transform them to be read in the room timetable array. After that operation was accomplished, the program calculated its penalty function. Obviously, we could not let this timetable without using Random Search on it. The aim of this Random Search was a bit different from the Random Search performed to design a complete timetable from a list of courses and teachers. Here, we just perform a Random Search to improve the capacity constraint while respecting the other constraints. In other words, courses are assigned to new classrooms without changing the day to which the courses are scheduled. The hours could change but not the day. This can then be compared to a room assignment optimisation. The results are depicted in *Table 3*.

| Running time | Penalty function        | Details          | Penalties                      | Initial |        | Optimized |        | Final  |        |
|--------------|-------------------------|------------------|--------------------------------|---------|--------|-----------|--------|--------|--------|
|              |                         |                  |                                | Q1      | Q2     | Q1        | Q2     | Q1     | Q2     |
| 40'          | TOTAL                   |                  |                                | 47.008  | 42.403 | 12.462    | 16.497 | 12.212 | 16.197 |
|              | Prof penalties          |                  | 100                            | 0       | 0      | 0         | 0      | 0      | 0      |
|              | Student Penalties       |                  | 100                            | 0       | 0      | 0         | 0      | 0      | 0      |
|              | Required room Penalties |                  | 1000                           | 0       | 0      | 0         | 0      | 0      | 0      |
|              | Capacity Penalties      |                  | $\Delta$ or $\Delta \times 10$ | 46.283  | 42.003 | 11.487    | 15.797 | 11.487 | 15.797 |
|              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                             | 700     | 400    | 400       | 275    | 700    | 400    |
|              |                         | 6 p.m. - 8 p.m.  | 25                             | 25      | 0      | 475       | 275    | 25     | 0      |
|              |                         | 8 p.m. - 10 p.m. | 50                             | 0       | 0      | 100       | 150    | 0      | 0      |

*Table 3* – Results for Mons current instances

Even if the schedule has changed in terms of hours, the day does not change. So, if we reschedule the courses to their initial hours, no new conflicts could appear. We can then conclude from those results that courses could better be assigned to classrooms. Indeed, we can see that the capacity penalty highly decreases after performing Random Search to try to optimise room assignment.

If we compare those results with the ones obtained by the program developed for this work (*Table 4*), we can see that results get close to each other. In the situation above, since the current timetable was already applied, it was easier to improve than a totally new timetable developed with a Random Search. However, we can see encouraging results.

| Running time | Penalty function        | Details | Penalties               | Initial |        | Optimized |        | Final  |        |
|--------------|-------------------------|---------|-------------------------|---------|--------|-----------|--------|--------|--------|
|              |                         |         |                         | Q1      | Q2     | Q1        | Q2     | Q1     | Q2     |
| 39'          | TOTAL                   |         |                         | 56.556  | 36.756 | 23.248    | 14.115 | 18.660 | 17.445 |
|              | Prof penalties          |         | 100                     | 700     | 0      | 0         | 0      | 0      | 0      |
|              | Student Penalties       |         | 100                     | 1.100   | 0      | 0         | 0      | 0      | 0      |
|              | Required room Penalties |         | 1000                    | 12.000  | 5.000  | 12.000    | 5.000  | 0      | 0      |
|              | Capacity Penalties      |         | $\Delta$ or $\Delta*10$ | 40.856  | 30.556 | 10.598    | 8.565  | 18.010 | 16.895 |
|              | Hours Penalties         |         |                         |         |        |           |        |        |        |
|              | 12 a.m. - 1 p.m.        |         | 25                      | 400     | 250    | 125       | 125    | 125    | 125    |
|              | 6 p.m. - 8 p.m.         |         | 25                      | 750     | 450    | 225       | 225    | 225    | 225    |
|              | 8 p.m. - 10 p.m.        |         | 50                      | 750     | 500    | 300       | 200    | 300    | 200    |

Table 4 – Result for one computation of Random Search to develop a timetable for Mons campus

In both cases Random Search is computed 250.000 times. Results for 1.000.000 iterations are displayed in Table 5. This table does not provide an evidence that if we increase the number of iterations of Random Search we can reach a better solution. The results are a bit better but very close. Since we increase the number of iterations we better explore the solution space. However, since we consider only 82 courses compared to 188 courses with Mons and Louvain-la-Neuve campuses, in 250.000 iterations we have more opportunities to improve the model if we consider Mons only.

| Running time | Penalty function        | Details | Penalties | Initial |        | Optimized |        | Final  |        |
|--------------|-------------------------|---------|-----------|---------|--------|-----------|--------|--------|--------|
|              |                         |         |           | Q1      | Q2     | Q1        | Q2     | Q1     | Q2     |
| 1h16'        | TOTAL                   |         |           | 56.556  | 36.756 | 24.811    | 14.658 | 16.707 | 17.182 |
|              | Prof penalties          |         | 100       | 700     | 0      | 0         | 0      | 0      | 0      |
|              | Student Penalties       |         | 100       | 1.100   | 0      | 0         | 0      | 0      | 0      |
|              | Required room Penalties |         | 1000      | 12.000  | 5.000  | 12.000    | 5.000  | 0      | 0      |
|              | Capacity Penalties      |         | Δ or Δ*10 | 40.856  | 30.556 | 12.111    | 9.083  | 16.007 | 16.607 |
|              | Hours Penalties         |         |           |         |        |           |        |        |        |
|              | 12 a.m. - 1 p.m.        |         | 25        | 400     | 250    | 100       | 100    | 100    | 100    |
|              | 6 p.m. - 8 p.m.         |         | 25        | 750     | 450    | 200       | 225    | 200    | 225    |
|              | 8 p.m. - 10 p.m.        |         | 50        | 750     | 500    | 400       | 250    | 400    | 250    |

Table 5 – 1.000.000 iterations for Random Search for Mons campus

We can finally notice that it takes much less time to compute Random Search on Mons campus: forty minutes compared to more or less six hours. As mentioned previously, it is due to the number of rooms to consider. In this case, we only have 20 rooms compared to 175 when we consider Mons and Louvain-la-Neuve campuses.

One last execution was executed on the model for both Mons and Louvain-la-Neuve campuses. We performed 1.000.000 iterations of Random Search for the first semester. We consider one semester only to save time. The results are displayed in Table 6. The goal was to see how the results could change when we increased the number of iterations. From this execution we can conclude that the number of iterations does not really matter. This is partially due to the stochastic characteristic of this method. Maybe for another computation of the code we would have provided better results, or even worse results.

It took much more time since we increased the number of iterations by four, but the results did not follow. It demonstrates that 250.000 are enough but since the method is random it is really hard to reach a better solution by increasing the number of iterations. From this we can come to the conclusion that Random Search is effective at the beginning of the process. Indeed, when starting Random Search, the penalty cost was extremely high and then it was easy to decrease it with that procedure.

| Running time | Penalty function        | Details          | Penalties               | Initial | Optimized | Final  |
|--------------|-------------------------|------------------|-------------------------|---------|-----------|--------|
|              |                         |                  |                         | Q1      | Q1        | Q1     |
| 8h39'        | TOTAL                   |                  |                         | 141.254 | 88.674    | 83.707 |
|              | Prof penalties          |                  | 100                     | 1.600   | 0         | 0      |
|              | Student Penalties       |                  | 100                     | 4.600   | 0         | 0      |
|              | Campus Penalties        |                  | 100                     | 0       | 0         | 0      |
|              | Required room Penalties |                  | 1000                    | 12.000  | 12.000    | 0      |
|              | Capacity Penalties      |                  | $\Delta$ or $\Delta*10$ | 118.679 | 74.749    | 81.782 |
|              | Hours Penalties         | 12 a.m. - 1 p.m. | 25                      | 950     | 425       | 425    |
|              |                         | 6 p.m. - 8 p.m.  | 25                      | 1.875   | 700       | 700    |
|              |                         | 8 p.m. - 10 p.m. | 50                      | 1.550   | 800       | 800    |

Table 6 – Results for 1.000.000 iterations of Random Search (Mons and Louvain-la-Neuve)

This chapter resumed the results of the performed program to design a course timetable system for LSM, at UCLouvain. There was also a little comparison of the results obtained with the current instances of Mons, before and after a room assignment optimisation was performed with Random Search, and the performed program on the campus of Mons only. In the next chapter, a criticism of the approach performed will be exposed.

## 5. Critical analysis

In this chapter, we present a critical analysis of the approximate approach to solution of the UCLouvain UCTP developed in the previous chapters. We will focus in particular on some of its strengths and weaknesses and we will indicate what we believe it could be improved in order to make the described approach usable in practice.

When we analyse the results from the previous chapter, we can easily conclude that the studied method was not the best choice in order to tackle the UCTP. This study required a lot of work and a lot of efforts to collect the needed data. Unfortunately, the latter were not exhaustive. The administrative services of both campuses (Mons and Louvain-la-Neuve) were very cooperative but could not provide very accurate information, for the sake of confidentiality.

Because of this, nobody could provide the suitable data to optimally tackle the UCTP. We had information about the typology of some classrooms but only for the campus of Mons. Thanks to a personal experience, we could determine which courses needed a specific room. The administrative service of Louvain-la-Neuve provided us information about the number of students enrolled for their courses, which was not the case for the campus of Mons. For Mons, we had to manually search for examination timetables that provided information about the number of students enrolled for all examinations on the campus, but this did not represent the real number of students enrolled for the courses. We did not have information about teachers' preferences in terms of availability and classrooms, and so on. All these real constraints did not allow us creating an optimal model at hundred percent, that allows providing an acceptable timetable starting from a list of courses and teachers.

This can surely be noticed in the results. The latter are satisfying from a theoretical point of view. Indeed, we can avoid conflicts and respect the hard constraints. However, the overall solution does not appear either realistic or satisfying. Specifically, if we look more in the details at the results, the courses are not optimally assigned to time periods: a lot of courses are still scheduled at 6 p.m., or even, still scheduled until 10 p.m. Nevertheless, the utilisation of the classrooms is correct. But the more classrooms we consider, the harder it is to optimally use them and then harder it is to assign a classroom to the course that fits it best. In particular, the more the classrooms, the more the solutions. Since the

studied method is stochastic, it is difficult to optimise 175 classrooms on two campuses in a short period of time.

On the other hand, the proceeded analyse on the UCTP instances of Mons and its results reveal that when we consider a small number of rooms, the utilisation of the Random Search is better. In the example of *Table 3*, the Random Search is used to optimise the occupancy of the classrooms. In this way, the use of Random Search is encouraging. Indeed, the current instances show that the penalties attached to the capacity of the rooms compared to the number of students enrolled in the course and assigned to this room are reduced. Before applying a Random Search of 250.000 iterations, the penalties due to the capacities were equal to 46.283 for the first semester and 42.003 for the second semester. After the execution of the code, those penalties respectively decreased to 11.487 and 15.797. These results demonstrate that the method studied brings an added value. It was not the case when we tried to develop an entire timetable from a list of courses and teachers but it is the case when we use this method as a complement to optimise the utilisation of the classroom, after defining a first timetable with another method, like an IP method, for instance.

Admittedly Random Search is not the ideal method to solve the UCTP. Actually, as the name suggests it is a stochastic method and it is also a heuristic one. Its goal is then to get closer to the optimal solution, without reaching it at any cost. To make its utilisation optimal or pertinent, a hybrid method is a good solution. In chapter two, the majority of the authors agree with this statement: hybridisation is the best way to solve the UCTP that considers a lot of different variables and elements.

In synthesis, Random Search did not prove its beneficial effects in order to create a complete timetable from a list of courses and teachers. To allow Random Search providing better results, complete data are necessary: the teachers' preferences in terms of availability and classrooms, the typology of the rooms, the set of courses provided by the whole university, and so on. However, this method demonstrated its added value when optimising the utilisation of the classrooms, after creating a first timetable with another type of approach, such as the ones described in chapter two.

## Conclusion

The objective of this work was to study a method to solve the UCTP at UCLouvain, especially for one of its faculties, the LSM. This study also aimed to determine if such a method, which is automated, could reduce the effort and the time needed by the administrative personal of both campuses to manage this task, that currently takes several weeks.

The answer provided thanks to this modest work has been acquired in a scientific way. Several methods that have already been implemented in other universities through the world are exposed in a theoretical chapter. The next chapter focuses on the studied method of this work: the Random Search. The hypotheses, the inputs and outputs, the evaluation function and the development stages of this method are introduced.

The results obtained are not very conclusive when we develop an entire timetable from a list of courses and teachers. After analysing those results, we can notice that the studied method is not made for being used as the main method to solve the UCTP. However, the other results presented in this work show that Random Search is a good secondary method to optimise the utilisation of the rooms of the university. Indeed, when courses are 'optimally' assigned to periods of time, Random Search can be used to modify the room assignment and to assign better fitting classrooms to courses. The results from *Table 3* bring evidence for that. Thanks to this method that optimises the room occupancy, we can improve the solution space exploration. Consequently, the results of this work are very satisfying.

Nevertheless, it would be interesting to test other methods to compare them and their results on this real case, to combine them to find better results. But to do so, we should have access to all the information needed to elaborate a complete and operational timetable, which was not the case in this work.

To conclude, this project is the result of a research work, which leads to a result that fits with my professional ambitions. It allowed to provide me experience, sharing and innovation and to get a better understanding in programming and operational research which are domains that interest me a lot.

## References

- ABDENNADHER, S. & MARTE, M. (1998). *University Timetabling Using Constraint Handling Rules*. Applied Artificial Intelligence, An International Journal, Volume 14, Issue 4, November 2010: pp. 311-325.
- AL-MILLI, N. (2010). *Hybrid Genetic Algorithms with Great Deluge For Course Timetabling*. IJCSNS International Journal of Computer Science and Network Security, Volume 10, No. 4: pp. 283-288.
- ALVAREZ-VALDES, R. & CRESPO, E. & TAMARIT, J. (2002). *Design and implementation of a course scheduling system using Tabu Search*. European Journal of Operational Research, Volume 137, Issue 3: pp. 512-523.
- BLAKESLEY, J. et al. (1998). *Academic Scheduling*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 223-236.
- BLUM, C. (2005). *Ant colony optimization: Introduction and recent trends*. Physics of Life Reviews, Volume 2: pp. 353-373.
- BLUM, C. et al. (2011). *Hybrid metaheuristics in combinatorial optimization: A survey*. Applied Soft Computing, The Official Journal of the World Federation on Soft Computing, Volume 11: pp. 4135-4151.
- BLUM, C. & Groß, R. (2015). *Swarm Intelligence in Optimization and Robotics*. Springer Handbook of Computational Intelligence, Kacprzyk & Pedrycz: pp. 1291-1309.
- BLUM, C. & ROLI, A. (2003). *Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison*. ACM Computing Surveys, Volume 35, No. 3: pp. 268-308.
- CARTER, M.W. & LAPORTE, G. (1998). *Recent developments in practical course timetabling*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 3-19.
- CHOWDHARY, A. et al. (2014). *Timetable Generation System*. International Journal of Computer Science and Mobile Computing, Volume 3, Issue 2: pp. 410-414.

CRUZ-CHÁVEZ, M. et al. (2016). *Solving a Real Constraint Satisfaction Model for the University Course Timetabling Problem: A Case Study*. Hindawi Publishing Corporation. Mathematical Problems in Engineering, Volume 2016: pp. 1-14.

DINKEL, J. & MOTE, J. & VENKATARAMANAN, M.A. (1989). *OR Practice—An Efficient Decision Support System for Academic Course Scheduling*. Operations Research, Volume 37, Issue 6: pp. 853-864.

ELMOHAMED, S. & CODDINGTON, P. & FOX, G. (1998). *A Comparison of Annealing Techniques for Academic Course Scheduling*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 92-112.

EURODECISION. (n.d.). *Algorithmes : Heuristiques et méta-heuristiques*. Retrieved from: <https://www.eurodecision.com/algorithmes/recherche-operationnelle-optimisation/heuristiques-meta-heuristiques> [Accessed on April 28, 2020].

FAHMY, A. (2016). *Optimization Algorithms in Project Scheduling*. Optimization Algorithms - Methods and Applications, Intech Open Science, Ozgur Baskan: pp. 171-193.

FEIZI-DERAKHSHI, M.-R. & BABEI, H. & HEIDARZADEH, J. (2012). *A Survey of Approaches for University Course Timetabling Problem*. 8th International Symposium on Intelligent and Manufacturing Systems. Sakarya University Department of Industrial Engineering, Adrasan, Antalya, Turkey: pp. 307-321.

GUNAWAN, A. & NG, K.-M. & POH, K.-L. (2007). *Solving the teacher assignment-course scheduling problem by a hybrid Algorithm*. Research Collection School Of Information Systems. International Journal of Computer, Information, and Systems Science, and Engineering, Volume 1, Issue 2: pp. 136-141.

HERTZ, A. (1990). *Finding a feasible course schedule using Tabu Search*. Discrete Applied Mathematics, Volume 35: pp. 255-270.

ILIS, M. et al. (2015). *Data mining techniques for student timetable optimization*. INFOTEH-JAHORINA, Volume 14: pp. 578-583.

KENDALL, G. & HUSSIN, N. (2005). *A Tabu Search Hyper-heuristic Approach to the Examination Timetabling Problem at the MARA University of Technology*. Practice and

Theory of Automated Timetabling V, Lecture Notes in Computer Science, Volume 3616, Burke & Trick: pp. 270–293.

KOSTUCH, P. (2005). *The University Course Timetabling Problem with a Three-Phase Approach*. Practice and Theory of Automated Timetabling V, Lecture Notes in Computer Science, Volume 3616, Burke & Trick: pp. 109-125.

NI, J. & YANG, N.-N. (2013). *Genetic Algorithm and its Application in Scheduling System*. TELKOMNIKA, Volume 11, No. 4: pp. 1934-1939.

PAECHTER, B. & RANKIN, R.C. & CUMMING, A. (1998). *Improving a Lecture Timetabling System for a University-Wide Use*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 156-165.

PAECHTER, B. & RANKIN, R.C. & CUMMING, A. (1998). *Timetabling the Classes of an Entire University with an Evolutionary Algorithm*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 865-874.

PERERA, M.T.M. & LANEL, G.H.J. (2016). *A Model to Optimize University Course Timetable Using Graph Coloring and Integer Linear Programming*. IOSR Journal Of Mathematics, Volume 12, Issue 5, Ver. III: pp. 13-18.

ROSSI-DARIA, O. et al. (2006). *A Local Search for the timetabling problem*. Proceedings of the 4th international conference on the Practice and Theory of Automated Timetabling: pp. 124–127.

SCHAERF, A. (1999). *A Survey of Automated Timetabling*. Artificial Intelligence Review, Volume 13: pp. 87–127.

SIDDHARTH, D. (2015). *Course Scheduling with Preference Optimization*. Master Thesis in Computer Science, The Pennsylvania State University.

VANDAELE, A. (2018). *Recherche Opérationnelle*. Syllabus, UCLouvain FUCaM Mons, Louvain School of Management, Mons, Belgique.

WHITE, G. & ZHANG, J. (1998). *Generating Complete University Timetables by Combining Tabu Search with Constraint Logic*. Practice and Theory of Automated Timetabling II, Lecture Notes in Computer Science, Volume 1408, Burke & Carter: pp. 187-196.

## Appendices

### Appendix 1 – Courses data

|     | A           | B         | C                                                            | D          | E                      | F                 | G        | H    | I      | J         | K                        | L             |
|-----|-------------|-----------|--------------------------------------------------------------|------------|------------------------|-------------------|----------|------|--------|-----------|--------------------------|---------------|
| 1   | Course code | Course ID | Course Name                                                  | Hrs / week | Teacher                | Students enrolled | Semestre | Year | Campus | Section   | Option                   | Required room |
| 99  | LLSMF2015   | 98        | Corporate Strategy in the Knowledge Society (Names from A to | 5          | Pégis Couderoy         | 42                | Q1       | MA2  | LLN    | INGE      | Tronc commun             |               |
| 100 | LLSMF2025   | 99        | Corporate Strategy in the Knowledge Society (Names from L to | 5          | Bernard Paque          | 34                | Q1       | MA2  | LLN    | INGE      | Tronc commun             |               |
| 101 | LLSMF2009   | 100       | Fiscalité                                                    | 5          | Marcel Gérard          | 95                | Q1       | MA2  | LLN    | INGE/GEST | Tronc commun             |               |
| 102 | LLSMF2090   | 101       | Risk Management, Internal Control and Auditing               | 5          | Loïc Decaux            | 149               | Q1       | MA2  | LLN    | INGE/GEST | Tronc commun             |               |
| 103 | LLSMS2103   | 102       | Internal Strategic Management (Names from A to K)            | 5          | Bernard Paque          | 89                | Q1       | MA2  | LLN    | GEST      | Tronc commun             |               |
| 104 | LLSMS2123   | 103       | Internal Strategic Management (Names from L to Z)            | 5          | Bernard Paque          | 129               | Q1       | MA2  | LLN    | GEST      | Tronc commun             |               |
| 105 | LLSMS2036   | 104       | Supply Chain Procurement                                     | 5          | Per Joakim Agrell      | 46                | Q1       | MA2  | LLN    | INGE/GEST | Sourcing and Procurement |               |
| 106 | LLSMS2037   | 105       | Sourcing Strategy                                            | 5          | Constantin Blome       | 46                | Q1       | MA2  | LLN    | INGE/GEST | Sourcing and Procurement |               |
| 107 | LLSMS2038   | 106       | Procurement Organisation and Scope                           | 5          | Constantin Blome       | 46                | Q1       | MA2  | LLN    | INGE/GEST | Sourcing and Procurement |               |
| 108 | MECON1104   | 107       | Economie                                                     | 3          | Nathalie Gilson        | 152               | Q1       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 109 | MQANT1110   | 108       | Mathématique de gestion 1                                    | 3          | Enrico Vitale          | 187               | Q1       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 110 | MSHUM1114   | 109       | Philosophie                                                  | 2          | Stanislas Deprez       | 253               | Q2       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 111 | MSHUM1116   | 110       | Sociologie                                                   | 2          | Géraldine André        | 279               | Q1       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 112 | MDROI1103   | 111       | Fondements du droit                                          | 3          | Jean Le Hardy de Beaul | 276               | Q1       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 113 | MGEST1105   | 112       | Comptabilité                                                 | 3          | Anne-Catherine Provos  | 197               | Q2       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 114 | MQANT1109   | 113       | Informatique de gestion                                      | 2          | François Fouss         | 239               | Q2       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 115 | MQANT1113   | 114       | Statistiques et probabilités                                 | 3          | Frédéric Vrins         | 243               | Q2       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 116 | MSHUM1115   | 115       | Psychologie                                                  | 2          | Coralie Buxant         | 254               | Q2       | BAC1 | MONS   | INGE/GEST | Bachelier                |               |
| 117 | MDROI1210   | 116       | Droit de l'entreprise                                        | 4          | Michel Dewolf          | 117               | Q1       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 118 | MECON1212   | 117       | Microéconomie                                                | 3          | Patrick Scarmure       | 207               | Q1       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 119 | MGEST1220   | 118       | Marketing                                                    | 3          | Nadia Sirinagaglia     | 168               | Q1       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 120 | MQANT1227   | 119       | Mathématique de gestion 2                                    | 3          | Nadine Meskens         | 120               | Q1       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 121 | MECON1211   | 120       | Macroéconomie                                                | 2          | Nathalie Gilson        | 102               | Q2       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 122 | MGEST1219   | 121       | Finance                                                      | 3          | Catherine D'Hondt      | 156               | Q2       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 123 | MGEST1222   | 122       | Production                                                   | 2          | Nadine Meskens         | 133               | Q2       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 124 | MQANT1221   | 123       | Inférence statistique                                        | 3          | Rudy De Winne          | 121               | Q2       | BAC2 | MONS   | INGE/GEST | Bachelier                |               |
| 125 | MINFO1201   | 124       | Informatique et algorithmique                                | 2          | François Fouss         | 96                | Q2       | BAC2 | MONS   | INGE      | Bachelier                |               |
| 126 | MTECH1201   | 125       | Technologie industrielle                                     | 3          | Laurence Baclin        | 47                | Q2       | BAC2 | MONS   | INGE      | Bachelier                |               |
| 127 | MGEST1218   | 126       | Analyse financière                                           | 3          | Anne-Catherine Provos  | 49                | Q2       | BAC2 | MONS   | GEST      | Bachelier                |               |
| 128 | MGEST1221   | 127       | Simulation de gestion : jeu d'entreprise                     | 2          | Isabelle Platten       | 46                | Q2       | BAC2 | MONS   | GEST      | Bachelier                |               |
| 129 | MGEST1325   | 128       | Comptabilité et contrôle de gestion                          | 3          | Dominique Helbois      | 107               | Q1       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 130 | MSHUM1328   | 129       | Sociologie des organisations                                 | 2          | Philippe Scieur        | 241               | Q1       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 131 | MSHUM1331   | 130       | Question de sciences religieuses                             | 1          | Stanislas Deprez       | 45                | Q1       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 132 | MECON1313   | 131       | Economie industrielle                                        | 2          | Patrick Scarmure       | 81                | Q2       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 133 | MGEST1323   | 132       | Management humain                                            | 2          | Laurent Taskin         | 145               | Q2       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 134 | MQANT1324   | 133       | Econométrie                                                  | 3          | Mikael Petitjean       | 123               | Q2       | BAC3 | MONS   | INGE/GEST | Bachelier                |               |
| 135 | MINFO1301   | 134       | Gestion de données informatisées                             | 2          | François Fouss         | 35                | Q1       | BAC3 | MONS   | INGE      | Bachelier                | Salle Info 4  |
| 136 | MQANT1329   | 135       | Optimisation                                                 | 2          | Danièle Catanzaro      | 45                | Q1       | BAC3 | MONS   | INGE      | Bachelier                |               |
| 137 | MINFO1302   | 136       | Projet de programmation                                      | 2          | Danièle Catanzaro      | 30                | Q1       | BAC3 | MONS   | INGE      | Bachelier                |               |
| 138 | MQANT1328   | 137       | Recherche opérationnelle                                     | 2          | Nadine Meskens         | 29                | Q2       | BAC3 | MONS   | INGE      | Bachelier                |               |
| 139 | MECON1312   | 138       | Economie internationale                                      | 2          | Nathalie Gilson        | 44                | Q2       | BAC3 | MONS   | GEST      | Bachelier                |               |
| 140 | MGEST1327   | 139       | Marketing financier                                          | 3          | Modèle Kervin de M...  | 37                | Q1       | BAC3 | MONS   | GEST      | Bachelier                |               |

Input data about courses are presented this way. Here we present only few data in order to show their format before being manipulated during the code execution. The course ID refers to the number used in the final output, the timetable, to schedule courses.

Here are presented all the considered courses and their features in this work.

| Code Code | Course ID | Course Name                                              | Hrs/ week | Teacher            | Students Enrolled | Semester | Year | Campus | Section | Option                | Required Room |
|-----------|-----------|----------------------------------------------------------|-----------|--------------------|-------------------|----------|------|--------|---------|-----------------------|---------------|
| LLSMS2030 | 1         | Supply Chain Management                                  | 5         | Pierre Semal       | 104               | Q1       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2031 | 2         | Tools for Supply Chain Management Decisions              | 5         | Mathieu Van Vyve   | 65                | Q1       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2035 | 3         | Supply Chain Coordination and Sourcing                   | 5         | Per Joakim Agrell  | 65                | Q1       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2032 | 4         | Advanced Operations Management : Models and Applications | 5         | Philippe Chevalier | 64                | Q2       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2033 | 5         | International Supply Chain Management                    | 5         | Philippe Chevalier | 63                | Q2       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2034 | 6         | Supply Chain Planning                                    | 5         | Marc Foret         | 61                | Q2       | MA1  | LLN    | INGE    | Supply Chain          |               |
| LLSMS2040 | 7         | Innovation Management                                    | 5         | Benoît Gailly      | 47                | Q1       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2041 | 8         | Economics of Innovation                                  | 5         | Lionel Artige      | 77                | Q1       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2042 | 9         | Developing Management                                    | 5         | Benoît Gailly      | 47                | Q1       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMF2011 | 10        | Knowledge Management                                     | 5         | Françoise de Viron | 43                | Q2       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2043 | 11        | Innovation Policy and Ecosystems                         | 5         | Benoît Gailly      | 43                | Q2       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2374 | 12        | Economics of the Internet                                | 5         | Paul Belleflamme   | 43                | Q2       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2080 | 13        | Cours bonus: International Entrepreneurship              | 5         | Frank Janssen      | 21                | Q2       | MA1  | LLN    | INGE    | Innovation Management |               |
| LLSMS2224 | 14        | Forecasting                                              | 5         | Bertrand Candelon  | 30                | Q1       | MA1  | LLN    | INGE    | Financial Engineering |               |
| LLSMS2225 | 15        | Derivatives Pricing                                      | 5         | Frédéric Vrans     | 33                | Q1       | MA1  | LLN    | INGE    | Financial Engineering |               |

|           |    |                                               |   |                        |     |    |     |     |               |                                                      |  |
|-----------|----|-----------------------------------------------|---|------------------------|-----|----|-----|-----|---------------|------------------------------------------------------|--|
| LLSMS2138 | 16 | Big data finance                              | 5 | Eric Ghysels           | 28  | Q2 | MA1 | LLN | INGE          | Financial Engineering                                |  |
| LLSMS2012 | 17 | Macro Finance                                 | 5 | Leonardo Iania         | 30  | Q2 | MA1 | LLN | INGE          | Financial Engineering                                |  |
| LLSMS2013 | 18 | Investments                                   | 5 | Renaud Beaupain        | 36  | Q1 | MA1 | LLN | INGE          | Financial Engineering                                |  |
| LLSMS2226 | 19 | Credit and interest rate risk                 | 5 | Frédéric Vrins         | 30  | Q2 | MA1 | LLN | INGE          | Financial Engineering                                |  |
| LLSMS2000 | 20 | Marketing Research                            | 5 | Nadia Steils           | 127 | Q1 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2002 | 21 | International Marketing                       | 5 | Thierry Jupsin         | 169 | Q1 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2007 | 22 | New Business Models and E-commerce            | 5 | Olivier de Broqueville | 137 | Q1 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2003 | 23 | Brand Management                              | 5 | Isabelle Schuiling     | 86  | Q2 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2004 | 24 | Big Data/Data Mining Applied to Marketing     | 5 | Ludovic Chevalier      | 82  | Q2 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2006 | 25 | On-line and Off-line Communication Strategies | 5 | Karine Charry          | 85  | Q2 | MA1 | LLN | INGE/<br>GEST | Marketing Strategy for Connected Brands              |  |
| LLSMS2280 | 26 | Business Ethics and Compliance Management     | 5 | Carlos Desmet          | 126 | Q1 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot in Corporate Sustainable Management |  |

|           |    |                                                                            |   |                  |     |    |     |     |               |                                                            |  |
|-----------|----|----------------------------------------------------------------------------|---|------------------|-----|----|-----|-----|---------------|------------------------------------------------------------|--|
| LLSMS2281 | 27 | Sustainable Management and Value Chains                                    | 5 | Carsten Reuter   | 124 | Q1 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot<br>in Corporate Sustainable Management    |  |
| LLSMS2282 | 28 | Finance and Responsible Investment Practices                               | 5 | Christel Dumas   | 95  | Q1 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot<br>in Corporate Sustainable Management    |  |
| LLSMS2283 | 29 | Sustainable Human Resource Management and Leadership                       | 5 | Ina Aust-Gronarz | 64  | Q2 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot<br>in Corporate Sustainable Management    |  |
| LLSMS2284 | 30 | Corporate Sustainability Reporting and Marketing Strategy                  | 5 | Yves De Rongé    | 65  | Q2 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot<br>in Corporate Sustainable Management    |  |
| LCPME2005 | 31 | Social and Sustainable Entrepreneurship                                    | 5 | Frédéric Dufays  | 85  | Q2 | MA1 | LLN | INGE/<br>GEST | Philippe de Woot<br>in Corporate Sustainable Management    |  |
| LCPME2001 | 32 | Théorie de l'entrepreneuriat                                               | 5 | Frank Janssen    | 34  | Q1 | MA1 | LLN | INGE/<br>GEST | Formation interdisciplinaire en création d'entreprise CPME |  |
| LCPME2002 | 33 | Aspects juridiques, économiques et managériaux de la création d'entreprise | 5 | Yves De Cordt    | 34  | Q1 | MA1 | LLN | INGE/<br>GEST | Formation interdisciplinaire en création d'entreprise CPME |  |
| LCPME2004 | 34 | Séminaire d'approfondissement en entrepreneuriat                           | 5 | Frank Janssen    | 36  | Q2 | MA1 | LLN | INGE/<br>GEST | Formation interdisciplinaire en création d'entreprise CPME |  |
| LLSMX2001 | 35 | Regenerative Economy                                                       | 5 | Yves De Rongé    | 27  | Q1 | MA1 | LLN | INGE/<br>GEST | Formation interdisciplinaire en création                   |  |

|           |    |                                                                    |   |                      |     |    |     |     |               |                                                                        |  |
|-----------|----|--------------------------------------------------------------------|---|----------------------|-----|----|-----|-----|---------------|------------------------------------------------------------------------|--|
|           |    |                                                                    |   |                      |     |    |     |     |               | d'entreprise<br>CPME                                                   |  |
| LLSMS2014 | 36 | Entrepreneurial<br>Finance                                         | 5 | Axel Funhoff         | 108 | Q1 | MA1 | LLN | INGE/<br>GEST | Formation<br>interdisciplinaire<br>en création<br>d'entreprise<br>CPME |  |
| LLSMS2080 | 37 | International<br>Entrepreneurship                                  | 5 | Frank Janssen        | 21  | Q2 | MA1 | LLN | INGE/<br>GEST | Formation<br>interdisciplinaire<br>en création<br>d'entreprise<br>CPME |  |
| LLSMS2081 | 38 | Strategic<br>Management of<br>Startups                             | 5 | Bartholomeus<br>Kamp | 15  | Q2 | MA1 | LLN | INGE/<br>GEST | Formation<br>interdisciplinaire<br>en création<br>d'entreprise<br>CPME |  |
| LCPME2005 | 39 | Social and<br>Sustainable<br>Entrepreneurship                      | 5 | Frédéric<br>Dufays   | 85  | Q2 | MA1 | LLN | INGE/<br>GEST | Formation<br>interdisciplinaire<br>en création<br>d'entreprise<br>CPME |  |
| LCPME2003 | 40 | Plans d'affaires et<br>étapes-clefs de la<br>création d'entreprise | 5 | Frank Janssen        | 34  | Q2 | MA1 | LLN | INGE/<br>GEST | Formation<br>interdisciplinaire<br>en création<br>d'entreprise<br>CPME |  |
| LLSMS2014 | 41 | Entrepreneurial<br>Finance                                         | 5 | Axel Funhoff         | 108 | Q1 | MA1 | LLN | INGE/<br>GEST | International<br>Business                                              |  |
| LLSMS2002 | 42 | International<br>Marketing                                         | 5 | Thierry Jupsin       | 169 | Q1 | MA1 | LLN | INGE/<br>GEST | International<br>Business                                              |  |
| LLSMS2030 | 43 | Supply Chain<br>Management                                         | 5 | Pierre Semal         | 104 | Q1 | MA1 | LLN | INGE/<br>GEST | International<br>Business                                              |  |
| LLSMS2114 | 44 | Entrepreneurship                                                   | 5 | Julie Hermans        | 27  | Q2 | MA1 | LLN | INGE/<br>GEST | International<br>Business                                              |  |
| LLSMS2065 | 45 | Cross Cultural<br>Competences and<br>Management                    | 5 | Ina Aust-<br>Gronarz | 26  | Q2 | MA1 | LLN | INGE/<br>GEST | International<br>Business                                              |  |

|           |    |                                                                   |      |                    |     |    |     |     |               |                        |  |
|-----------|----|-------------------------------------------------------------------|------|--------------------|-----|----|-----|-----|---------------|------------------------|--|
| LLSMS2299 | 46 | New Economic Models                                               | 5    | Yves De Rongé      | 26  | Q2 | MA1 | LLN | INGE/<br>GEST | International Business |  |
| LLSMS2397 | 47 | Maatschappelijke verantwoordelijkheid van bedrijven               | 5    | Sabine Denis       | 96  | Q1 | MA1 | LLN | INGE/<br>GEST | Tronc commun           |  |
| LLSMS2098 | 48 | Corporate Social Responsibility (Names from A to E)               | 5    | Carlos Desmet      | 189 | Q1 | MA1 | LLN | INGE/<br>GEST | Tronc commun           |  |
| LLSMS2099 | 49 | Corporate Social Responsibility (Names from F to N)               | 5    | Carlos Desmet      | 153 | Q1 | MA1 | LLN | INGE/<br>GEST | Tronc commun           |  |
| LLSMS2398 | 50 | Corporate Social Responsibility (Names from O to Z)               | 5    | Carlos Desmet      | 116 | Q1 | MA1 | LLN | INGE/<br>GEST | Tronc commun           |  |
| LLSMF2016 | 51 | Performance Management (Names A to K)                             | 5    | Kim Adamsen        | 118 | Q2 | MA1 | LLN | INGE          | Tronc commun           |  |
| LLSMF2026 | 52 | Performance Management (Names L to Z)                             | 5    | Kim Adamsen        | 121 | Q2 | MA1 | LLN | INGE          | Tronc commun           |  |
| LLSMF2014 | 53 | Data Analytics applied in Business (Names A to K)                 | 5    | Manuel Kolp        | 106 | Q2 | MA1 | LLN | INGE          | Tronc commun           |  |
| LLSMF2013 | 54 | Data Analytics applied in Business (Names L to Z)                 | 5    | Manuel Kolp        | 157 | Q2 | MA1 | LLN | INGE          | Tronc commun           |  |
| LLSMF2018 | 55 | Technological and Quantitative Project (incl. Project Management) | 12,5 | Pierre Semal       | 226 | Q1 | MA1 | LLN | INGE          | Tronc commun           |  |
| LLSMS2102 | 56 | Advanced Strategic Marketing (Names from A to K)                  | 5    | Isabelle Schuiling | 160 | Q1 | MA1 | LLN | GEST          | Tronc commun           |  |
| LLSMS2122 | 57 | Advanced Strategic Marketing (Names from L to Z)                  | 5    | Nicolas Lambert    | 175 | Q1 | MA1 | LLN | GEST          | Tronc commun           |  |
| LLSMS2100 | 58 | Corporate Finance (Names from A to K)                             | 5    | Bruno Colmant      | 175 | Q1 | MA1 | LLN | GEST          | Tronc commun           |  |

|           |    |                                                                                     |    |                         |     |    |     |     |      |                                                                   |  |
|-----------|----|-------------------------------------------------------------------------------------|----|-------------------------|-----|----|-----|-----|------|-------------------------------------------------------------------|--|
| LLSMS2120 | 59 | Corporate Finance<br>(Names from L to Z)                                            | 5  | Bruno Colmant           | 145 | Q1 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMS2121 | 60 | Management<br>Accounting (Names<br>from A to K)                                     | 5  | Kim Adamsen             | 141 | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMS2121 | 61 | Management<br>Accounting (Names<br>from L to Z)                                     | 5  | Kim Adamsen             | 141 | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMS2104 | 62 | Advanced Human<br>Resource and<br>Organisation<br>Management (Names<br>from A to K) | 5  | Ina Aust-<br>Gronarz    | 175 | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMS2124 | 63 | Advanced Human<br>Resource and<br>Organisation<br>Management (Names<br>from L to Z) | 5  | Ina Aust-<br>Gronarz    | 162 | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMA2001 | 64 | Epistemology of<br>Management                                                       | 5  | Matthieu de<br>Nanteuil | 4   | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMA2002 | 65 | Research Methods                                                                    | 10 | Philippe<br>Chevalier   | 4   | Q1 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMA2006 | 66 | Theories of<br>Organizations in<br>Management                                       | 5  | Régis<br>Coeurderoy     | 4   | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMA2007 | 67 | Communication and<br>Writing Skills                                                 | 5  | Manuel Kolp             | 5   | Q2 | MA1 | LLN | GEST | Tronc commun                                                      |  |
| LLSMS2070 | 68 | Diagnostic<br>organisationnel                                                       | 5  | Evelyne<br>Léonard      | 100 | Q1 | MA1 | LLN | GEST | Management<br>humain et<br>intervention dans<br>les organisations |  |
| LLSMS2072 | 69 | Sociologie clinique<br>du travail                                                   | 5  | Thomas<br>Périlleux     | 99  | Q1 | MA1 | LLN | GEST | Management<br>humain et<br>intervention dans<br>les organisations |  |
| LLSMF2007 | 70 | Gestion du<br>changement                                                            | 5  | Alain Vas               | 28  | Q1 | MA1 | LLN | GEST | Management<br>humain et                                           |  |

|           |    |                                                   |   |                           |     |    |     |     |      |                                                          |  |
|-----------|----|---------------------------------------------------|---|---------------------------|-----|----|-----|-----|------|----------------------------------------------------------|--|
|           |    |                                                   |   |                           |     |    |     |     |      | intervention dans les organisations                      |  |
| LLSMS2071 | 71 | Diversity Management                              | 5 | Philippe Depaepe          | 60  | Q2 | MA1 | LLN | GEST | Management humain et intervention dans les organisations |  |
| LLSMS2073 | 72 | Strategic Human Resource Management               | 5 | Laurent Taskin            | 74  | Q2 | MA1 | LLN | GEST | Management humain et intervention dans les organisations |  |
| LLSMS2074 | 73 | Contemporary Issues in Human Resources Management | 5 | Marie Antoine             | 74  | Q2 | MA1 | LLN | GEST | Management humain et intervention dans les organisations |  |
| LLSMS2018 | 74 | European Corporate Governance                     | 5 | Karine Cerrada Cristia    | 34  | Q1 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2060 | 75 | European Economic Policy                          | 5 | Eric Cornuel              | 54  | Q1 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2064 | 76 | European Public Affairs                           | 5 | Jean-Christophe Defraigne | 29  | Q1 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2061 | 77 | Corporate Strategy in Europe                      | 5 | Guilhem Bascle            | 22  | Q2 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2062 | 78 | Management of European Companies (Business Cases) | 5 | Eric Cornuel              | 24  | Q2 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2063 | 79 | Industrial Relations in Europe                    | 5 | Evelyne Léonard           | 24  | Q2 | MA1 | LLN | GEST | European Business                                        |  |
| LLSMS2014 | 80 | Entrepreneurial Finance                           | 5 | Axel Funhoff              | 108 | Q1 | MA1 | LLN | GEST | Corporate Finance                                        |  |
| LLSMS2017 | 81 | IAS/IFRS                                          | 5 | Bruno Colmant             | 56  | Q1 | MA1 | LLN | GEST | Corporate Finance                                        |  |
| LLSMS2108 | 82 | International Financial Management                | 5 | James Thewissen           | 51  | Q1 | MA1 | LLN | GEST | Corporate Finance                                        |  |
| LLSMS2009 | 83 | Risk Management of Financial Institutions         | 5 | Luc Henrard               | 44  | Q2 | MA1 | LLN | GEST | Corporate Finance                                        |  |

|           |    |                                                                 |   |                             |    |    |     |     |               |                       |  |
|-----------|----|-----------------------------------------------------------------|---|-----------------------------|----|----|-----|-----|---------------|-----------------------|--|
| LLSMS2110 | 84 | Advanced Corporate Finance                                      | 5 | Yue Zhang                   | 39 | Q2 | MA1 | LLN | GEST          | Corporate Finance     |  |
| LLSMS2118 | 85 | Project Evaluation                                              | 5 | Leonardo Iania              | 38 | Q2 | MA1 | LLN | GEST          | Corporate Finance     |  |
| LLSMS2223 | 86 | Principles of banking and finance                               | 5 | Tom Dechaene                | 91 | Q1 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2054 | 87 | International Finance                                           | 5 | Bertrand Candelon           | 87 | Q1 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2027 | 88 | Financial Innovation                                            | 5 | Bertrand Candelon           | 83 | Q1 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2128 | 89 | Central banking, financial regulation, supervision              | 5 | Luc Henrard                 | 54 | Q2 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2140 | 90 | International corporate finance                                 | 5 | Yue Zhang                   | 55 | Q2 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2029 | 91 | International financial management                              | 5 | Denis Knaepen               | 54 | Q2 | MA1 | LLN | INGE/<br>GEST | International Finance |  |
| LLSMS2001 | 92 | Consumer Behavior                                               | 5 | Nicolas Kervyn de Meerendré | 69 | Q1 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMS2021 | 93 | Managing Brand Equity                                           | 5 | Valérie Swaen               | 74 | Q1 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMS2024 | 94 | Marketing Studies                                               | 5 | Gordy Pleyers               | 64 | Q1 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMS2022 | 95 | Customer Relationship Management (CRM)                          | 5 | Nicolas Kervyn de Meerendré | 25 | Q2 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMS2023 | 96 | Ethical and Sustainable Consumption                             | 5 | Valérie Swaen               | 25 | Q2 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMS2005 | 97 | Neuromarketing, virtual reality and experimental approach [M]   | 5 | Gordy Pleyers               | 26 | Q2 | MA1 | LLN | GEST          | Consumer Insights     |  |
| LLSMF2015 | 98 | Corporate Strategy in the Knowledge Society (Names from A to K) | 5 | Régis Coeurderoy            | 42 | Q1 | MA2 | LLN | INGE          | Tronc commun          |  |

|           |     |                                                                 |   |                           |     |    |          |      |               |                          |  |
|-----------|-----|-----------------------------------------------------------------|---|---------------------------|-----|----|----------|------|---------------|--------------------------|--|
| LLSMF2025 | 99  | Corporate Strategy in the Knowledge Society (Names from L to Z) | 5 | Bernard Paque             | 34  | Q1 | MA2      | LLN  | INGE          | Tronc commun             |  |
| LLSMF2009 | 100 | Fiscalité                                                       | 5 | Marcel Gérard             | 95  | Q1 | MA2      | LLN  | INGE/<br>GEST | Tronc commun             |  |
| LLSMF2090 | 101 | Risk Management, Internal Control and Auditing                  | 5 | Loïc Decaux               | 149 | Q1 | MA2      | LLN  | INGE/<br>GEST | Tronc commun             |  |
| LLSMS2103 | 102 | Internal Strategic Management (Names from A to K)               | 5 | Bernard Paque             | 89  | Q1 | MA2      | LLN  | GEST          | Tronc commun             |  |
| LLSMS2123 | 103 | Internal Strategic Management (Names from L to Z)               | 5 | Bernard Paque             | 129 | Q1 | MA2      | LLN  | GEST          | Tronc commun             |  |
| LLSMS2036 | 104 | Supply Chain Procurement                                        | 5 | Per Joakim Agrell         | 46  | Q1 | MA2      | LLN  | INGE/<br>GEST | Sourcing and Procurement |  |
| LLSMS2037 | 105 | Sourcing Strategy                                               | 5 | Constantin Blome          | 46  | Q1 | MA2      | LLN  | INGE/<br>GEST | Sourcing and Procurement |  |
| LLSMS2038 | 106 | Procurement Organisation and Scope                              | 5 | Constantin Blome          | 46  | Q1 | MA2      | LLN  | INGE/<br>GEST | Sourcing and Procurement |  |
| MECON1104 | 107 | Economie                                                        | 3 | Nathalie Gilson           | 192 | Q1 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MQANT1110 | 108 | Mathématique de gestion 1                                       | 3 | Enrico Vitale             | 187 | Q1 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MSHUM1114 | 109 | Philosophie                                                     | 2 | Stanislas Deprez          | 253 | Q2 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MSHUM1116 | 110 | Sociologie                                                      | 2 | Géraldine André           | 279 | Q1 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MDROI1103 | 111 | Fondements du droit                                             | 3 | Jean Le Hardy de Beaulieu | 276 | Q1 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MGEST1105 | 112 | Comptabilité                                                    | 3 | Anne-Catherine Provost    | 197 | Q2 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |
| MQANT1109 | 113 | Informatique de gestion                                         | 2 | François Fouss            | 239 | Q2 | BAC<br>1 | MONS | INGE/<br>GEST | Bachelier                |  |

|           |     |                                          |   |                        |     |    |       |      |               |           |  |
|-----------|-----|------------------------------------------|---|------------------------|-----|----|-------|------|---------------|-----------|--|
| MQANT1113 | 114 | Statistiques et probabilités             | 3 | Frédéric Vrins         | 243 | Q2 | BAC 1 | MONS | INGE/<br>GEST | Bachelier |  |
| MSHUM1115 | 115 | Psychologie                              | 2 | Coralie Buxant         | 254 | Q2 | BAC 1 | MONS | INGE/<br>GEST | Bachelier |  |
| MDROI1210 | 116 | Droit de l'entreprise                    | 4 | Michel Dewolf          | 117 | Q1 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MECON1212 | 117 | Microéconomie                            | 3 | Patrick Scarmure       | 207 | Q1 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MGEST1220 | 118 | Marketing                                | 3 | Nadia Sinigaglia       | 168 | Q1 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MQANT1227 | 119 | Mathématique de gestion 2                | 3 | Nadine Meskens         | 120 | Q1 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MECON1211 | 120 | Macroéconomie                            | 2 | Nathalie Gilson        | 102 | Q2 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MGEST1219 | 121 | Finance                                  | 3 | Catherine D'Hondt      | 156 | Q2 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MGEST1222 | 122 | Production                               | 2 | Nadine Meskens         | 133 | Q2 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MQANT1221 | 123 | Inférence statistique                    | 3 | Rudy De Winne          | 121 | Q2 | BAC 2 | MONS | INGE/<br>GEST | Bachelier |  |
| MINFO1201 | 124 | Informatique et algorithmique            | 2 | François Fouss         | 96  | Q2 | BAC 2 | MONS | INGE          | Bachelier |  |
| MTECH1201 | 125 | Technologie industrielle                 | 3 | Laurence Baclin        | 47  | Q2 | BAC 2 | MONS | INGE          | Bachelier |  |
| MGEST1218 | 126 | Analyse financière                       | 3 | Anne-Catherine Provost | 49  | Q2 | BAC 2 | MONS | GEST          | Bachelier |  |
| MGEST1221 | 127 | Simulation de gestion : jeu d'entreprise | 2 | Isabelle Platten       | 46  | Q2 | BAC 2 | MONS | GEST          | Bachelier |  |
| MGEST1325 | 128 | Comptabilité et contrôle de gestion      | 3 | Dominique Helbois      | 107 | Q1 | BAC 3 | MONS | INGE/<br>GEST | Bachelier |  |
| MSHUM1328 | 129 | Sociologie des organisations             | 2 | Philippe Scieur        | 241 | Q1 | BAC 3 | MONS | INGE/<br>GEST | Bachelier |  |
| MSHUM1331 | 130 | Question de sciences religieuses         | 1 | Stanislas Deprez       | 45  | Q1 | BAC 3 | MONS | INGE/<br>GEST | Bachelier |  |
| MECON1313 | 131 | Economie industrielle                    | 2 | Patrick Scarmure       | 81  | Q2 | BAC 3 | MONS | INGE/<br>GEST | Bachelier |  |

|           |     |                                                          |      |                                |     |    |          |      |               |                       |              |
|-----------|-----|----------------------------------------------------------|------|--------------------------------|-----|----|----------|------|---------------|-----------------------|--------------|
| MGEST1323 | 132 | Management humain                                        | 2    | Laurent Taskin                 | 145 | Q2 | BAC<br>3 | MONS | INGE/<br>GEST | Bachelier             |              |
| MQANT1324 | 133 | Econométrie                                              | 3    | Mikael<br>Petitjean            | 123 | Q2 | BAC<br>3 | MONS | INGE/<br>GEST | Bachelier             |              |
| MINFO1301 | 134 | Gestion de données<br>informatisées                      | 2    | François Fouss                 | 35  | Q1 | BAC<br>3 | MONS | INGE          | Bachelier             | Salle Info 4 |
| MQANT1329 | 135 | Optimisation                                             | 2    | Daniele<br>Catanzaro           | 45  | Q1 | BAC<br>3 | MONS | INGE          | Bachelier             |              |
| MINFO1302 | 136 | Projet de<br>programmation                               | 2    | Daniele<br>Catanzaro           | 30  | Q1 | BAC<br>3 | MONS | INGE          | Bachelier             |              |
| MQANT1328 | 137 | Recherche<br>opérationnelle                              | 2    | Nadine<br>Meskens              | 29  | Q2 | BAC<br>3 | MONS | INGE          | Bachelier             |              |
| MECON1312 | 138 | Economie<br>internationale                               | 2    | Nathalie Gilson                | 44  | Q2 | BAC<br>3 | MONS | GEST          | Bachelier             |              |
| MGEST1327 | 139 | Marchés financiers                                       | 2    | Nicolas Kervyn<br>de Meerendré | 37  | Q1 | BAC<br>3 | MONS | GEST          | Bachelier             |              |
| MGEST1328 | 140 | Études marketing                                         | 2    | Catherine<br>D'Hondt           | 29  | Q1 | BAC<br>3 | MONS | GEST          | Bachelier             |              |
| MGEST1322 | 141 | Stratégie                                                | 2    | Alain Vas                      | 103 | Q1 | BAC<br>3 | MONS | GEST          | Bachelier             |              |
| MSHUM2201 | 142 | Compétences<br>relationnelles                            | 2,5  | Laurent<br>Lievens             | 110 | Q1 | MA1      | MONS | INGE/<br>GEST | Tronc commun          |              |
| MLSMM2111 | 143 | Responsabilité<br>sociétale de<br>l'entreprise           | 5    | Laurent<br>Lievens             | 18  | Q2 | MA1      | MONS | INGE/<br>GEST | Tronc commun          |              |
| MLSMM2116 | 144 | Data Analytics                                           | 5    | François Fouss                 | 23  | Q1 | MA1      | MONS | INGE          | Tronc commun          |              |
| MLSMM2117 | 145 | Projet quantitatif et<br>gestion de projet               | 12,5 | Daniele<br>Catanzaro           | 26  | Q2 | MA1      | MONS | INGE          | Tronc commun          |              |
| MLSMM2112 | 146 | Finance d'entreprise                                     | 5    | Catherine<br>D'Hondt           | 32  | Q1 | MA1      | MONS | GEST          | Tronc commun          |              |
| MLSMM2114 | 147 | Ressources humaines<br>& management des<br>organisations | 5    | Laurent Taskin                 | 31  | Q1 | MA1      | MONS | GEST          | Tronc commun          |              |
| MLSMM2113 | 148 | Strategic Marketing                                      | 5    | Caroline<br>Ducarroz           | 48  | Q2 | MA1      | MONS | GEST          | Tronc commun          |              |
| MLSMM2152 | 149 | Nouvelles<br>technologies et                             | 5    | Bart Jourquin                  | 43  | Q1 | MA1      | MONS | INGE          | Business<br>Analytics |              |

|           |     |                                            |   |                   |    |    |     |      |               |                                            |                          |
|-----------|-----|--------------------------------------------|---|-------------------|----|----|-----|------|---------------|--------------------------------------------|--------------------------|
|           |     | pratiques émergentes                       |   |                   |    |    |     |      |               |                                            |                          |
| MLSMM2151 | 150 | Data Mining                                | 5 | Nadine Meskens    | 22 | Q1 | MA1 | MONS | INGE          | Business Analytics                         |                          |
| MLSMM2153 | 151 | Web Mining                                 | 5 | François Fouss    | 18 | Q1 | MA1 | MONS | INGE          | Business Analytics                         | Salle Info 1             |
| MLSMM2154 | 152 | Machine Learning                           | 5 | Bertrand Lebichot | 14 | Q2 | MA1 | MONS | INGE          | Business Analytics                         |                          |
| MLSMM2155 | 153 | Quantitative Decision Making               | 5 | Daniele Catanzaro | 13 | Q2 | MA1 | MONS | INGE          | Business Analytics                         |                          |
| MLSMM2156 | 154 | Systèmes de recommandation                 | 5 | Felix Sommer      | 13 | Q2 | MA1 | MONS | INGE          | Business Analytics                         |                          |
| MLSMM2121 | 155 | Derivative Securities                      | 5 | Isabelle Platten  | 45 | Q1 | MA1 | MONS | INGE/<br>GEST | Financial Management                       |                          |
| MLSMM2122 | 156 | Equity Valuation                           | 5 | Mikael Petitjean  | 45 | Q1 | MA1 | MONS | INGE/<br>GEST | Financial Management                       | Salle Info 5 (Bloomberg) |
| MLSMM2123 | 157 | Fixed Income Analysis                      | 5 | Isabelle Platten  | 49 | Q1 | MA1 | MONS | INGE/<br>GEST | Financial Management                       |                          |
| MLSMM2126 | 158 | Portfolio Management                       | 5 | Catherine D'Hondt | 37 | Q2 | MA1 | MONS | INGE/<br>GEST | Financial Management                       |                          |
| MLSMM2125 | 159 | Risk Management and Financial Institutions | 5 | Isabelle Platten  | 35 | Q2 | MA1 | MONS | INGE/<br>GEST | Financial Management                       |                          |
| MLSMM2124 | 160 | Behavioural Finance                        | 5 | Rudy De Winne     | 30 | Q2 | MA1 | MONS | INGE/<br>GEST | Financial Management                       | Salle Info 5 (Bloomberg) |
| MLSMM2131 | 161 | Distribution                               | 5 | Simon Hazée       | 29 | Q1 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |                          |
| MLSMM2132 | 162 | Price Management                           | 5 | Caroline Ducarroz | 32 | Q1 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |                          |
| MLSMM2133 | 163 | Product and Innovation                     | 5 | Nadia Sinigaglia  | 30 | Q1 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |                          |

|           |     |                                                    |     |                        |    |    |     |      |               |                                            |  |
|-----------|-----|----------------------------------------------------|-----|------------------------|----|----|-----|------|---------------|--------------------------------------------|--|
| MLSMM2134 | 164 | E-comportement du consommateur                     | 5   | Karine Charry          | 41 | Q2 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |  |
| MLSMM2136 | 165 | Tendances en Digital Marketing                     | 5   | Ingrid Poncin          | 39 | Q2 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |  |
| MLSMM2135 | 166 | Méthodes et modèles en marketing                   | 5   | Caroline Ducarroz      | 42 | Q2 | MA1 | MONS | INGE/<br>GEST | Décisions marketing tactiques et digitales |  |
| MLSMM2251 | 167 | Modélisation des systèmes de transport             | 5   | Bart Jourquin          | 25 | Q1 | MA1 | MONS | INGE/<br>GEST | Transportation Management and Supply Chain |  |
| MLSMM2252 | 168 | Supply Chain Management                            | 5   | Jean-Sébastien Tancrez | 27 | Q1 | MA1 | MONS | INGE/<br>GEST | Transportation Management and Supply Chain |  |
| MLSMM2253 | 169 | Transport et mobilité                              | 5   | Bart Jourquin          | 24 | Q1 | MA1 | MONS | INGE/<br>GEST | Transportation Management and Supply Chain |  |
| MLSMM2141 | 170 | Contrôle interne et gestion des risques            | 7,5 | Bénédicte Vessié       | 29 | Q1 | MA1 | MONS | GEST          | Révisorat et expertise comptable           |  |
| MLSMM2143 | 171 | Règles d'évaluation                                | 5   | Anne-Catherine Provost | 30 | Q1 | MA1 | MONS | GEST          | Révisorat et expertise comptable           |  |
| MLSMM2142 | 172 | Droit comptable                                    | 5   | Yvan Stempniewsky      | 31 | Q1 | MA1 | MONS | GEST          | Révisorat et expertise comptable           |  |
| MLSMM2145 | 173 | IAS et IFRS                                        | 7,5 | Yvan Stempniewsky      | 30 | Q2 | MA1 | MONS | GEST          | Révisorat et expertise comptable           |  |
| MLSMM2146 | 174 | Normes comptables internationales de consolidation | 5   | Bénédicte Vessié       | 30 | Q2 | MA1 | MONS | GEST          | Révisorat et expertise comptable           |  |

|             |     |                                                           |   |                          |    |    |     |      |               |                                  |  |
|-------------|-----|-----------------------------------------------------------|---|--------------------------|----|----|-----|------|---------------|----------------------------------|--|
| MLSMM2144   | 175 | Droit des sociétés approfondi et gouvernance d'entreprise | 5 | Werner Derijcke          | 31 | Q2 | MA1 | MONS | GEST          | Révisorat et expertise comptable |  |
| MLSMM2218 G | 176 | Pilotage de la performance                                | 5 | Géraldine Danaux         | 56 | Q1 | MA2 | MONS | GEST          | Tronc commun                     |  |
| MLSMM2218 I | 177 | Pilotage de la performance                                | 5 | Géraldine Danaux         | 56 | Q1 | MA2 | MONS | INGE          | Tronc commun                     |  |
| MLSMM2211   | 178 | Gestion de la PME                                         | 5 | Karl Colin               | 29 | Q1 | MA2 | MONS | INGE/<br>GEST | Tronc commun                     |  |
| MLSMM2216   | 179 | Corporate Strategy in the Knowledge Society               | 5 | Nadia Sinigaglia         | 7  | Q1 | MA2 | MONS | INGE          | Tronc commun                     |  |
| MLSMM2217   | 180 | Integrated Information Systems                            | 5 | Bart Jourquin            | 34 | Q1 | MA2 | MONS | INGE          | Tronc commun                     |  |
| MLSMM2212   | 181 | Fiscalité                                                 | 5 | Marie Lamensch           | 38 | Q1 | MA2 | MONS | GEST          | Tronc commun                     |  |
| MLSMM2214   | 182 | International Strategic Management                        | 5 | Gilles Carlier d'Odeigne | 26 | Q1 | MA2 | MONS | GEST          | Tronc commun                     |  |
| MLSMM2263   | 183 | Savoir entreprendre                                       | 5 | Vincent Truyens          | 25 | Q1 | MA2 | MONS | INGE/<br>GEST | Entrepreneuriat                  |  |
| MHELH2262   | 184 | Entrepreneuriat technologique                             | 5 | Julie Hermans            | 25 | Q1 | MA2 | MONS | INGE/<br>GEST | Entrepreneuriat                  |  |
| MLSMM2261   | 185 | Entrepreneuriat collectif                                 | 5 | Julie Hermans            | 24 | Q1 | MA2 | MONS | INGE/<br>GEST | Entrepreneuriat                  |  |
| MCOMU2202   | 186 | Communication marketing                                   | 5 | Karine Charry            | 53 | Q1 | MA2 | MONS | GEST          | Communication marketing          |  |
| MCOMU2203   | 187 | Processus de création publicitaire                        | 5 | Marc Soumilion           | 47 | Q1 | MA2 | MONS | GEST          | Communication marketing          |  |
| MCOMU2204   | 188 | Méthodologie de la communication marketing                | 5 | Annie Courbet            | 42 | Q1 | MA2 | MONS | GEST          | Communication marketing          |  |

## Appendix 2 – Rooms data

|     | A         | B        | C      | D          |
|-----|-----------|----------|--------|------------|
| 1   | Room Name | Capacity | Campus | Building   |
| 138 | SUD3      | 70       | LLN    | Sud        |
| 139 | SUD4      | 40       | LLN    | Sud        |
| 140 | SUD5      | 40       | LLN    | Sud        |
| 141 | SUD6      | 60       | LLN    | Sud        |
| 142 | SUD7      | 40       | LLN    | Sud        |
| 143 | SUD11     | 201      | LLN    | Sud        |
| 144 | SUD12     | 40       | LLN    | Sud        |
| 145 | SUD13     | 50       | LLN    | Sud        |
| 146 | SUD14     | 40       | LLN    | Sud        |
| 147 | SUD15     | 40       | LLN    | Sud        |
| 148 | SUD16     | 50       | LLN    | Sud        |
| 149 | SUD17     | 40       | LLN    | Sud        |
| 150 | SUD8      | 294      | LLN    | Sud        |
| 151 | SUD9      | 211      | LLN    | Sud        |
| 152 | SUD18     | 316      | LLN    | Sud        |
| 153 | SUD19     | 249      | LLN    | Sud        |
| 154 | BST1      | 42       | LLN    | BST        |
| 155 | BST11     | 42       | LLN    | BST        |
| 156 | BST21     | 42       | LLN    | BST        |
| 157 | A1        | 409      | MONS   | Bâtiment D |
| 158 | A2        | 272      | MONS   | Bâtiment D |
| 159 | A3        | 182      | MONS   | Bâtiment D |
| 160 | A4        | 154      | MONS   | Bâtiment D |
| 161 | 11        | 32       | MONS   | Bâtiment D |
| 162 | 12        | 72       | MONS   | Bâtiment D |
| 163 | 13        | 46       | MONS   | Bâtiment D |

In the sheet 'Rooms' of 'DataFile' all classrooms considered in the model are listed. It provides information about the classroom name, its capacity, its campus and its building. Here are presented all the considered rooms in this work.

| Room Name | Capacity | Campus | Building Name |
|-----------|----------|--------|---------------|
| AGOR1     | 60       | LLN    | Agora         |
| AGOR2     | 60       | LLN    | Agora         |
| AGOR3     | 60       | LLN    | Agora         |
| AGOR4     | 60       | LLN    | Agora         |
| AGOR10    | 300      | LLN    | Agora         |
| AGOR11    | 299      | LLN    | Agora         |
| AGOR12    | 162      | LLN    | Agora         |
| AGOR13    | 162      | LLN    | Agora         |
| AGOR14    | 159      | LLN    | Agora         |
| BARB0     | 50       | LLN    | Sainte Barbe  |
| BARB1     | 65       | LLN    | Sainte Barbe  |
| BARB2     | 50       | LLN    | Sainte Barbe  |
| BARB3     | 50       | LLN    | Sainte Barbe  |
| BARB10    | 50       | LLN    | Sainte Barbe  |
| BARB11    | 65       | LLN    | Sainte Barbe  |
| BARB12    | 50       | LLN    | Sainte Barbe  |
| BARB13    | 50       | LLN    | Sainte Barbe  |
| BARB20    | 40       | LLN    | Sainte Barbe  |
| BARB21    | 50       | LLN    | Sainte Barbe  |
| BARB22    | 40       | LLN    | Sainte Barbe  |
| BARB23    | 40       | LLN    | Sainte Barbe  |
| BARB91    | 156      | LLN    | Sainte Barbe  |
| BARB92    | 176      | LLN    | Sainte Barbe  |

|        |     |     |              |
|--------|-----|-----|--------------|
| BARB93 | 156 | LLN | Sainte Barbe |
| BARB94 | 185 | LLN | Sainte Barbe |
| COUB1  | 79  | LLN | Coubertin    |
| COUB3  | 18  | LLN | Coubertin    |
| COUB4  | 18  | LLN | Coubertin    |
| COUB5  | 40  | LLN | Coubertin    |
| COUB6  | 18  | LLN | Coubertin    |
| COUB10 | 300 | LLN | Coubertin    |
| COUB11 | 150 | LLN | Coubertin    |
| COUB12 | 200 | LLN | Coubertin    |
| COUB14 | 18  | LLN | Coubertin    |
| COUB15 | 18  | LLN | Coubertin    |
| COUB16 | 40  | LLN | Coubertin    |
| COUB17 | 18  | LLN | Coubertin    |
| DOYE22 | 48  | LLN | Doyen        |
| DOYE31 | 227 | LLN | Doyen        |
| DOYE32 | 297 | LLN | Doyen        |
| DOYE21 | 48  | LLN | Doyen        |
| ERAS54 | 70  | LLN | Erasme       |
| ERAS55 | 70  | LLN | Erasme       |
| ERAS56 | 70  | LLN | Erasme       |
| ERAS57 | 30  | LLN | Erasme       |
| ERAS58 | 60  | LLN | Erasme       |
| LAVO51 | 100 | LLN | Lavoisier    |
| LAVO52 | 32  | LLN | Lavoisier    |
| LAVO53 | 32  | LLN | Lavoisier    |
| LECL51 | 47  | LLN | Leclercq     |
| LECL52 | 47  | LLN | Leclercq     |
| LECL53 | 47  | LLN | Leclercq     |
| LECL54 | 47  | LLN | Leclercq     |
| LECL60 | 74  | LLN | Leclercq     |
| LECL61 | 74  | LLN | Leclercq     |
| LECL62 | 74  | LLN | Leclercq     |
| LECL70 | 30  | LLN | Leclercq     |
| LECL72 | 30  | LLN | Leclercq     |
| LECL73 | 47  | LLN | Leclercq     |
| LECL80 | 24  | LLN | Leclercq     |
| LECL81 | 24  | LLN | Leclercq     |
| LECL82 | 32  | LLN | Leclercq     |
| LECL83 | 32  | LLN | Leclercq     |
| LECL84 | 32  | LLN | Leclercq     |
| LECL85 | 32  | LLN | Leclercq     |
| MERC1  | 46  | LLN | Mercator     |
| MERC2  | 54  | LLN | Mercator     |
| MERC4  | 54  | LLN | Mercator     |
| MERC11 | 46  | LLN | Mercator     |

|        |     |     |              |
|--------|-----|-----|--------------|
| MERC12 | 54  | LLN | Mercator     |
| MERC14 | 54  | LLN | Mercator     |
| MERC15 | 30  | LLN | Mercator     |
| MERC17 | 30  | LLN | Mercator     |
| MERC20 | 24  | LLN | Mercator     |
| MERC21 | 30  | LLN | Mercator     |
| MERC24 | 20  | LLN | Mercator     |
| MERC25 | 30  | LLN | Mercator     |
| MERC28 | 30  | LLN | Mercator     |
| MONT1  | 180 | LLN | Montesquieu  |
| MONT2  | 180 | LLN | Montesquieu  |
| MONT3  | 180 | LLN | Montesquieu  |
| MONT4  | 180 | LLN | Montesquieu  |
| MONT10 | 348 | LLN | Montesquieu  |
| MONT11 | 356 | LLN | Montesquieu  |
| MORE51 | 100 | LLN | More         |
| MORE52 | 100 | LLN | More         |
| MORE53 | 100 | LLN | More         |
| MORE54 | 50  | LLN | More         |
| MORE55 | 50  | LLN | More         |
| MORE56 | 50  | LLN | More         |
| MORE57 | 50  | LLN | More         |
| MORE63 | 30  | LLN | More         |
| MORE70 | 30  | LLN | More         |
| MORE71 | 30  | LLN | More         |
| MORE72 | 30  | LLN | More         |
| MORE73 | 30  | LLN | More         |
| MORE74 | 30  | LLN | More         |
| MORE76 | 30  | LLN | More         |
| MORE77 | 30  | LLN | More         |
| MORE78 | 30  | LLN | More         |
| MORE81 | 30  | LLN | More         |
| MORE82 | 30  | LLN | More         |
| MCUR5  | 30  | LLN | Marie Curie  |
| MCUR6  | 30  | LLN | Marie Curie  |
| PCUR1  | 56  | LLN | Pierre Curie |
| PCUR2  | 38  | LLN | Pierre Curie |
| PCUR3  | 38  | LLN | Pierre Curie |
| PCUR4  | 56  | LLN | Pierre Curie |
| PCUR5  | 56  | LLN | Pierre Curie |
| PCUR6  | 30  | LLN | Pierre Curie |
| PCUR8  | 32  | LLN | Pierre Curie |
| A01    | 210 | LLN | Sciences     |
| A02    | 269 | LLN | Sciences     |
| A03    | 185 | LLN | Sciences     |
| A10    | 600 | LLN | Sciences     |

|          |     |      |              |
|----------|-----|------|--------------|
| ETUSCES  | 160 | LLN  | Sciences     |
| SOCR-240 | 68  | LLN  | Socrate      |
| SOCR-242 | 100 | LLN  | Socrate      |
| SOCR10   | 675 | LLN  | Socrate      |
| SOCR11   | 299 | LLN  | Socrate      |
| SOCR20   | 30  | LLN  | Socrate      |
| SOCR21   | 70  | LLN  | Socrate      |
| SOCR23   | 40  | LLN  | Socrate      |
| SOCR24   | 24  | LLN  | Socrate      |
| SOCR25   | 40  | LLN  | Socrate      |
| SOCR26   | 24  | LLN  | Socrate      |
| SOCR27   | 30  | LLN  | Socrate      |
| SOCR28   | 30  | LLN  | Socrate      |
| SOCR40   | 79  | LLN  | Socrate      |
| SOCR41   | 30  | LLN  | Socrate      |
| SOCR42   | 30  | LLN  | Socrate      |
| SOCR43   | 59  | LLN  | Socrate      |
| STUD11   | 430 | LLN  | Studio       |
| STUD13   | 143 | LLN  | Studio       |
| SUD1     | 114 | LLN  | Croix du Sud |
| SUD2     | 40  | LLN  | Croix du Sud |
| SUD3     | 70  | LLN  | Croix du Sud |
| SUD4     | 40  | LLN  | Croix du Sud |
| SUD5     | 40  | LLN  | Croix du Sud |
| SUD6     | 60  | LLN  | Croix du Sud |
| SUD7     | 40  | LLN  | Croix du Sud |
| SUD11    | 201 | LLN  | Croix du Sud |
| SUD12    | 40  | LLN  | Croix du Sud |
| SUD13    | 50  | LLN  | Croix du Sud |
| SUD14    | 40  | LLN  | Croix du Sud |
| SUD15    | 40  | LLN  | Croix du Sud |
| SUD16    | 50  | LLN  | Croix du Sud |
| SUD17    | 40  | LLN  | Croix du Sud |
| SUD8     | 294 | LLN  | Croix du Sud |
| SUD9     | 211 | LLN  | Croix du Sud |
| SUD18    | 316 | LLN  | Croix du Sud |
| SUD19    | 249 | LLN  | Croix du Sud |
| BST1     | 42  | LLN  | BST          |
| BST11    | 42  | LLN  | BST          |
| BST21    | 42  | LLN  | BST          |
| A1       | 409 | MONS | Bâtiment D   |
| A2       | 272 | MONS | Bâtiment D   |
| A3       | 182 | MONS | Bâtiment D   |
| A4       | 154 | MONS | Bâtiment D   |
| 11       | 32  | MONS | Bâtiment D   |
| 12       | 72  | MONS | Bâtiment D   |

|                          |    |      |            |
|--------------------------|----|------|------------|
| 13                       | 46 | MONS | Bâtiment D |
| 14                       | 46 | MONS | Bâtiment D |
| 15                       | 51 | MONS | Bâtiment D |
| 17                       | 36 | MONS | Bâtiment D |
| 18                       | 22 | MONS | Bâtiment D |
| 19                       | 30 | MONS | Bâtiment D |
| 21                       | 48 | MONS | Bâtiment D |
| 22                       | 60 | MONS | Bâtiment D |
| 23                       | 60 | MONS | Bâtiment D |
| 24                       | 40 | MONS | Bâtiment D |
| 25                       | 90 | MONS | Bâtiment D |
| Salle Info 1             | 32 | MONS | Bâtiment D |
| Salle Info 4             | 32 | MONS | Bâtiment D |
| Salle Info 5 (Bloomberg) | 32 | MONS | Bâtiment D |

### Appendix 3 – Room timetable ‘Final\_Sched’

|               | Final Schedule Q1 |     |    |     |     |       |     |     |    |     |     |
|---------------|-------------------|-----|----|-----|-----|-------|-----|-----|----|-----|-----|
| Starting hour | LLN               |     |    |     |     |       |     |     |    |     |     |
| 8             | 58                | 15  | 0  | 26  | 32  | 14    | 22  | 98  | 18 | 105 | 69  |
| 9             | 58                | 15  | 0  | 26  | 32  | 14    | 22  | 98  | 18 | 105 | 69  |
| 10            | 58                | 15  | 0  | 26  | 32  | 14    | 22  | 98  | 18 | 105 | 69  |
| 11            | 92                | 0   | 35 | 255 | 0   | 42    | 87  | 68  | 9  | 36  | 0   |
| 12            | 92                | 0   | 35 | 255 | 0   | 42    | 87  | 68  | 9  | 36  | 0   |
| 13            | 92                | 0   | 35 | 255 | 0   | 42    | 87  | 68  | 9  | 36  | 0   |
| 14            | 59                | 43  | 99 | 93  | 102 | 21    | 244 | 101 | 48 | 0   | 2   |
| 15            | 59                | 43  | 99 | 93  | 102 | 21    | 244 | 101 | 48 | 0   | 2   |
| 16            | 59                | 43  | 99 | 93  | 102 | 21    | 244 | 101 | 48 | 0   | 2   |
| 17            | 20                | 100 | 47 | 74  | 86  | 0     | 244 | 65  | 41 | 57  | 0   |
| 18            | 20                | 100 | 47 | 74  | 86  | 0     | 103 | 65  | 41 | 57  | 0   |
| 19            | 20                | 100 | 47 | 74  | 86  | 0     | 103 | 65  | 41 | 57  | 0   |
| 20            | 0                 | 0   | 0  | 216 | 0   | 0     | 103 | 65  | 0  | 0   | 0   |
| 21            | 0                 | 0   | 0  | 216 | 0   | 0     | 0   | 0   | 0  | 0   | 0   |
|               | LU                | MA  | ME | JE  | VE  | LU    | MA  | ME  | JE | VE  | LU  |
|               | AGOR1             |     |    |     |     | AGOR2 |     |     |    |     |     |
|               | Final Schedule Q2 |     |    |     |     |       |     |     |    |     |     |
| Starting hour | LLN               |     |    |     |     |       |     |     |    |     |     |
| 8             | 78                | 40  | 25 | 90  | 13  | 0     | 0   | 0   | 62 | 0   | 52  |
| 9             | 78                | 40  | 25 | 90  | 13  | 0     | 273 | 0   | 62 | 0   | 52  |
| 10            | 78                | 40  | 25 | 90  | 13  | 34    | 273 | 0   | 62 | 0   | 52  |
| 11            | 77                | 19  | 10 | 30  | 29  | 34    | 63  | 0   | 45 | 89  | 285 |
| 12            | 77                | 19  | 10 | 30  | 29  | 34    | 63  | 0   | 45 | 89  | 285 |
| 13            | 77                | 19  | 10 | 30  | 29  | 0     | 63  | 0   | 45 | 89  | 0   |
| 14            | 60                | 72  | 0  | 38  | 97  | 66    | 91  | 0   | 73 | 61  | 95  |
| 15            | 60                | 72  | 0  | 38  | 97  | 66    | 91  | 0   | 73 | 61  | 95  |
| 16            | 60                | 72  | 0  | 38  | 97  | 66    | 91  | 0   | 73 | 61  | 95  |
| 17            | 37                | 85  | 12 | 0   | 0   | 0     | 23  | 0   | 6  | 0   | 0   |
| 18            | 37                | 85  | 12 | 64  | 0   | 241   | 23  | 228 | 6  | 0   | 0   |
| 19            | 37                | 85  | 12 | 64  | 0   | 241   | 23  | 228 | 6  | 84  | 0   |
| 20            | 0                 | 0   | 0  | 64  | 0   | 0     | 280 | 0   | 0  | 84  | 0   |
| 21            | 0                 | 0   | 0  | 0   | 0   | 0     | 280 | 0   | 0  | 84  | 0   |
|               | LU                | MA  | ME | JE  | VE  | LU    | MA  | ME  | JE | VE  | LU  |
|               | AGOR1             |     |    |     |     | AGOR2 |     |     |    |     |     |

This is a representation of the 3D timetable in 2D. Numbers above zero correspond to course IDs and zero means that the classroom is empty at that time. Course IDs appears as many times as the lecture course lasts, since one time slot equals one hour. Here only

the timetable for AGOR1 and AGOR2 in Louvain-la-Neuve are presented. The sheet contains 875 columns, which represents the 175 rooms multiplied by 5 days per week. It is then complicated to show them all.

#### **Appendix 4 – Final timetables representation ‘Sched\_by\_year\_Mons’**

| Final Schedule by year MONS |                                                                   |                                                   |                                                             |    |  |
|-----------------------------|-------------------------------------------------------------------|---------------------------------------------------|-------------------------------------------------------------|----|--|
| Q1                          |                                                                   |                                                   |                                                             |    |  |
| BAC1                        | INGE                                                              |                                                   |                                                             |    |  |
| LU                          | MA                                                                | ME                                                | JE                                                          | VE |  |
| 8h                          |                                                                   | Sociologie - Local: 25<br>- Prof: Géraldine André |                                                             |    |  |
| 9h                          |                                                                   | Sociologie - Local: 25<br>- Prof: Géraldine André | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 10h                         | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 11h                         | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 12h                         | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   |                                                             |    |  |
| 13h                         |                                                                   |                                                   |                                                             |    |  |
| 14h                         |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 15h                         |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 16h                         |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 17h                         |                                                                   |                                                   |                                                             |    |  |
| 18h                         |                                                                   |                                                   |                                                             |    |  |
| 19h                         |                                                                   |                                                   |                                                             |    |  |
| 20h                         |                                                                   |                                                   |                                                             |    |  |
| 21h                         |                                                                   |                                                   |                                                             |    |  |
| 22h                         |                                                                   |                                                   |                                                             |    |  |

| BAC1 | GEST                                                              |                                                   |                                                             |    |  |
|------|-------------------------------------------------------------------|---------------------------------------------------|-------------------------------------------------------------|----|--|
| LU   | MA                                                                | ME                                                | JE                                                          | VE |  |
| 8h   |                                                                   | Sociologie - Local: 25<br>- Prof: Géraldine André |                                                             |    |  |
| 9h   |                                                                   | Sociologie - Local: 25<br>- Prof: Géraldine André | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 10h  | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 11h  | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   | Mathématique de gestion 1 - Local: 17 - Prof: Enrico Vitale |    |  |
| 12h  | Fondements du droit - Local: 17 - Prof: Jean Le Hardj de Beaulieu |                                                   |                                                             |    |  |
| 13h  |                                                                   |                                                   |                                                             |    |  |
| 14h  |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 15h  |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 16h  |                                                                   | Economie - Local: A2<br>- Prof: Nathalie Gilson   |                                                             |    |  |
| 17h  |                                                                   |                                                   |                                                             |    |  |
| 18h  |                                                                   |                                                   |                                                             |    |  |
| 19h  |                                                                   |                                                   |                                                             |    |  |
| 20h  |                                                                   |                                                   |                                                             |    |  |
| 21h  |                                                                   |                                                   |                                                             |    |  |
| 22h  |                                                                   |                                                   |                                                             |    |  |

This is the final representation of the room timetable. We create a schedule for each year, section and option in order to provide an easy way to read a timetable for each student. Here are presented only two output schedules in order to show how they all are displayed in the final Excel file.

### Appendix 5 – Penalty costs

| Constraints         | Details                           | Costs              |
|---------------------|-----------------------------------|--------------------|
| Professor conflicts |                                   | 100                |
| Student conflicts   |                                   | 100                |
| Campus conflicts    |                                   | 100                |
| Required room       |                                   | 1.000              |
| Capacity constraint | Capacity < Enrolled students      | $\Delta \times 10$ |
|                     | Capacity $\geq$ Enrolled students | $\Delta$           |
| Hours constraint    | 12 a.m. – 1 p.m.                  | 25                 |
|                     | 6 p.m. – 8 p.m.                   | 25                 |
|                     | 8 p.m. – 10 p.m.                  | 50                 |

In this table, we list the cost penalty associated to each constraint considered in the model. for Capacity constraint, the symbol delta,  $\Delta$ , represents the difference between the room capacity and the number enrolled to the course that is taught in that room.

For the other penalty constraints, we count the number of times we have a conflict, or we face the predefined situation. Then, we multiply the number generated by its cost. Finally, we sum all these penalty constraints to have the final result for the penalty function.

### Appendix 6 – Initial timetable creation Mosel code

```

procedure SeveralBlocksofCoursePerWeek
  CourseToAdd:=0;
  forall(i in Courses | Data(i,3)="5" OR Data(i,3)="8" OR Data(i,3)="10" OR Data(i,3)="12") do
    if Data(i,3)="5" OR Data(i,3)="8" then CourseToAdd+=1;
    else CourseToAdd+=2;
    end-if
  end-do
end-procedure
SeveralBlocksofCoursePerWeek

```

The first operation has for aim to separate courses of more than four hours a week into two or three lecture classes. So, with this procedure we count the number of courses that will be added. They will be added to data course with a new course ID to be clearly recognisable in the final timetable. Then we must copy information about these courses for the new course ID and adapt those from the original data array. To do so, we use an intermediary data array. Finally, in a new data array the old data array information is copied, and we add the one from the intermediary data array.

```

NewCourse:=MaxCourses+1;
cours:=1;
forall(i in Courses | Data(i,3)="5" OR Data(i,3)="8" OR Data(i,3)="10" OR Data(i,3)="12") do
  if Data(i,3)="5" then
    Data(i,3):="3";
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="2";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
  elif Data(i,3)="8" then
    Data(i,3):="4";
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="4";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
  elif Data(i,3)="10" then
    Data(i,3):="4";
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="3";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
    NewCourse+=1;
    cours+=1;
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="3";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
  elif Data(i,3)="12" then
    Data(i,3):="4";
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="4";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
    NewCourse+=1;
    cours+=1;
    IntermediaryData(cours,1):=string(NewCourse);
    IntermediaryData(cours,2):=Data(i,2);
    IntermediaryData(cours,3):="4";
    IntermediaryData(cours,4):=Data(i,4);
    IntermediaryData(cours,5):=Data(i,5);
    IntermediaryData(cours,6):=Data(i,6);
    IntermediaryData(cours,7):=Data(i,7);
    IntermediaryData(cours,8):=Data(i,8);
    IntermediaryData(cours,9):=Data(i,9);
    IntermediaryData(cours,10):=Data(i,10);
    IntermediaryData(cours,11):=Data(i,11);
  end-if
  NewCourse+=1;
  cours+=1;
end-do

```

```

declarations
  NewMaxCourses:=MaxCourses+CourseToAdd;
  NewCourses=1..NewMaxCourses;
  row: integer;

  NewData: array(NewCourses,1..11) of string;
end-declarations

forall(i in Courses) do
  forall(j in 1..11) do
    NewData(i,j):=Data(i,j);
  end-do
end-do

row:=1;
forall(i in (MaxCourses+1)..(MaxCourses+CourseToAdd)) do
  forall(j in 1..11) do
    NewData(i,j):=IntermediaryData(row,j);
  end-do
  row+=1;
end-do

```

Now we have all data needed to start creating an initial timetable. The next procedure aims at vectorising the data array. The vectorisation transforms each course of the data array in a serie of one hour course. For instance, if course ID 4 is taught three hours a week, the procedure transforms this information in a serie of three course IDs 4 of one hour. The procedure already assigns Mons' courses to Mons campus and Louvain-la-Neuve's courses to Louvain-la-Neuve campus.

```

procedure VectorizingNewData(a:array(1..(MaxRooms*MaxDays*MaxHours)) of integer, b:array(1..(MaxRooms*MaxDays*MaxHours)) of integer)
  declarations
    p,q,r,s,m: integer;
    nbRoom: integer;
    changeRoom: integer;
  end-declarations

  nbRoom:=1;
  changeRoom:=1;
  p:=1;
  q:=1;
  r:=1;
  s:=1;
  forall(j in NewCourses) do
    if NewData(j,6)="Q1" then
      m:=integer(NewData(j,3));

      if NewData(j,8)="LLN" then
        forall(k in 1..m) do
          a(p):=integer(NewData(j,1));
          p+=1;
        end-do
      elif NewData(j,8)="MONS" then
        forall(k in 1..m) do
          a((MaxRooms*LLN*MaxDays*MaxHours)+r):=integer(NewData(j,1));
          r+=1;
        end-do
      end-if
    elif NewData(j,6)="Q2" then
      m:=integer(NewData(j,3));

      if NewData(j,8)="LLN" then
        forall(k in 1..m) do
          b(q):=integer(NewData(j,1));
          q+=1;
        end-do
      elif NewData(j,8)="MONS" then
        forall(k in 1..m) do
          b((MaxRooms*LLN*MaxDays*MaxHours)+s):=integer(NewData(j,1));
          s+=1;
        end-do
      end-if
    end-if
    changeRoom+=1;
    if changeRoom > MaxDays*MaxHours then
      nbRoom+=1;
      changeRoom:=1;
    end-if
  end-do
end-procedure
VectorizingNewData(HoursPerCourseQ1,HoursPerCourseQ2)

```

The next procedure consists in creating the initial timetable. This initial timetable is a 3-dimensional table which contains course IDs or zero if the classroom is free at that period. So the vectorised array is copied in the timetable. However, the procedure avoids that lecture courses are spread out on several days because there are not enough hours left in the day. So when it happens, we put zeros for the rest of the day and switch to the next day.

```

procedure InitialSchedule(a:array(1..(MaxRooms*MaxDays*MaxHours)) of integer,b:array(Rooms,Days,Hours) of integer)
  declarations
    p,m,r,LeftHours: integer;
  end-declarations

  p:=1;
  forall(i in Rooms) do
    forall(j in Days) do
      LeftHours:=MaxHours;
      r:=0;
      forall(n in Hours) do
        if a(p)<>0 then
          m:=integer(NewData(a(p),3));
          m-=r;

          if m<=LeftHours then
            b(i,j,n):=a(p);
            p+=1;

            if a(p)<>a(p-1) AND a(p)<>0 then r:=0
            elif a(p)=a(p-1) AND a(p)<>0 then r+=1;
            end-if
          else
            b(i,j,n):=0;
          end-if
        elif a(p)=0 then
          b(i,j,n):=0;
          p+=1;
        end-if
        LeftHours-=1;
      end-do
    end-do
  end-do
end-procedure
InitialSchedule(HoursPerCourseQ1,TimetableQ1)
InitialSchedule(HoursPerCourseQ2,TimetableQ2)

```

Since we created a 3-dimensional array, it is not possible to display it in an Excel file. The goal of the next procedures is to allow that this timetable can be displayed as a 2-dimensional array. First we have to vectorise again the timetable to copy it in a 2-dimension array. The latter is then initialised to an Excel sheet called "Init\_Sched".

```

procedure VectorizingTimetable(a:array(1..(MaxRooms*MaxDays*MaxHours)) of integer, b:array(Rooms,Days,Hours) of integer)
  declarations
    q: integer;
  end-declarations

  q:=1;
  forall(i in Rooms) do
    forall(j in Days) do
      forall(n in Hours) do
        a(q):=b(i,j,n);
        q+=1;
      end-do
    end-do
  end-do
end-procedure
VectorizingTimetable(NewHoursPerCourseQ1, TimetableQ1)
VectorizingTimetable(NewHoursPerCourseQ2, TimetableQ2)

declarations
  DisplayedTimeTableQ1: array(Hours,1..MaxRooms*MaxDays) of integer;
  DisplayedTimeTableQ2: array(Hours,1..MaxRooms*MaxDays) of integer;
end-declarations

procedure DisplayTimetable(a:array(Hours,1..MaxRooms*MaxDays) of integer, b:array(1..(MaxRooms*MaxDays*MaxHours)) of integer)
  declarations
    c,l,t: integer;
  end-declarations

  c:=1;
  l:=1;
  while(l<(MaxRooms*MaxDays*MaxHours)) do
    t:=1;
    while(t<=MaxHours) do
      a(t,c):=b(l);

      t+=1;
      l+=1;
    end-do
    c+=1;
  end-do
end-procedure
DisplayTimetable(DisplayedTimeTableQ1, NewHoursPerCourseQ1)
DisplayTimetable(DisplayedTimeTableQ2, NewHoursPerCourseQ2)

```

## Appendix 7 – Penalties calculation Mosel code

The penalties calculation is divided into four procedures.

```

declarations
  prof: integer;
  CProf=100;
  PProf: integer;

  stds: integer;
  CStudent=100;
  PStud: integer;

  camp: integer;
  CCampus=100;
  PCampus: integer;
end-declarations

procedure Campus_Penalties(a:array(Rooms,Days,Hours) of integer)
  declarations
    l,m: integer;
  end-declarations

  camp:=0;
  forall(g in 1..MaxRoomsLLN) do
    forall(i in (MaxRoomsLLN+1)..MaxRooms) do
      forall(j in Days) do
        forall(n in 1..MaxHours-1) do
          forall(o in Hours[o>n]) do
            l:=a(g,j,n);
            m:=a(i,j,o);

            if m=0 OR l=0 then break;
            elif NewData(1,4)=NewData(m,4) then
              if (NewData(1,7)="MA1" AND NewData(m,7)="MA1") OR (NewData(1,7)="MA2" AND NewData(m,7)="MA2")
              OR (NewData(1,7)="MA1" AND NewData(m,7)="MA2") OR (NewData(1,7)="MA2" AND NewData(m,7)="MA1") then
                if (NewData(1,10)="Tronc commun" AND NewData(m,10)<>"Tronc commun") OR (NewData(m,10)="Tronc commun"
                AND NewData(1,10)<>"Tronc commun") then
                  camp+=1;
                  camp-=1;
                else camp+=1;
                end-if
              end-if
            end-if
          end-do
        end-do
      end-do
    end-do
  end-do
  PCampus:=CCampus*camp;
end-procedure

```

The first procedure counts the number of hours for which a professor is on both campuses the same day. The second procedure counts the number of hours teachers and students are in conflict, which means that a teacher should teach several different courses at the same time or students cannot attend to their courses because they are taught at the same time. For these procedures, we ensure that we do not consider part 1 master courses with part 2 master courses in conflict.

```

procedure Prof_Student_Penalties(a:array(Rooms,Days,Hours) of integer)
  declarations
    l,m: integer;
  end-declarations

  prof:=0;
  stds:=0;
  forall(g in 1..MaxRooms-1) do
    forall(i in Rooms[i]>g) do
      forall(j in Days) do
        forall(n in Hours) do
          l:=a(g,j,n);
          m:=a(i,j,n);

          if m=0 OR l=0 then break;
          elif NewData(l,4)=NewData(m,4) then
            ! même prof
            if (NewData(l,7)="MA1" AND NewData(m,7)="MA1") OR (NewData(l,7)="MA2" AND NewData(m,7)="MA2")
              OR (NewData(l,7)="MA1" AND NewData(m,7)="MA2") OR (NewData(l,7)="MA2" AND NewData(m,7)="MA1") then
              if (NewData(l,10)="Tronc commun" AND NewData(m,10)<>"Tronc commun") OR (NewData(m,10)="Tronc commun"
                AND NewData(l,10)<>"Tronc commun") then
                prof+=1;
                prof-=1;
              else prof+=1;
              end-if
            else prof+=1;
            end-if
          end-if

          if m=0 OR l=0 then break;
          elif (NewData(l,7)=NewData(m,7)) AND (NewData(l,7)="BAC1" OR NewData(l,7)="BAC2" OR NewData(l,7)="BAC3") then
            if NewData(l,9)=NewData(m,9) then stds+=1;
            elif NewData(l,9)="INGE" AND NewData(m,9)="INGE/GEST" then stds+=1;
            elif NewData(l,9)="GEST" AND NewData(m,9)="INGE/GEST" then stds+=1;
            elif NewData(m,9)="INGE" AND NewData(l,9)="INGE/GEST" then stds+=1;
            elif NewData(m,9)="GEST" AND NewData(l,9)="INGE/GEST" then stds+=1;
            end-if
          elif NewData(l,7)=NewData(m,7) AND (NewData(l,7)="MA1" OR NewData(l,7)="MA2") then
            if NewData(l,10)=NewData(m,10) then
              if NewData(l,10)="Tronc commun" then
                if NewData(l,9)=NewData(m,9) then stds+=1;
                elif NewData(l,9)="INGE" AND NewData(m,9)="INGE/GEST" then stds+=1;
                elif NewData(l,9)="GEST" AND NewData(m,9)="INGE/GEST" then stds+=1;
                elif NewData(m,9)="INGE" AND NewData(l,9)="INGE/GEST" then stds+=1;
                elif NewData(m,9)="GEST" AND NewData(l,9)="INGE/GEST" then stds+=1;
                end-if
              else stds+=1;
              end-if
            elif (NewData(l,10)="Tronc commun" AND NewData(m,10)<>"Tronc commun") OR (NewData(m,10)="Tronc commun"
              AND NewData(l,10)<>"Tronc commun") then
              stds+=1;
              stds-=1;
            end-if
          end-if
        end-do
      end-do
    end-do
  end-do
  PProf:=prof*CProf;
  PStud:=stds*CStudent;
end-procedure

```

The next procedure counts the number of hours for which a course requiring a specific room is not assigned to this required room and for which a course is taught between 12 a.m. and 1 p.m., 6 p.m. and 8 p.m., and 8 p.m. and 10 p.m.

```

declarations
  rroom, PRoom: integer;
  CRoom=1000;

  o,p,q,PHours: integer;
  CHours13_14=25;
  CHours18_20=25;
  CHours20_22=50;
end-declarations

procedure RequiredRoom_Hours_Penalties(a:array(Rooms,Days,Hours) of integer,b:array(Rooms,1..4) of string)
  rroom:=0;
  o:=0;
  p:=0;
  q:=0;
  forall(i in Rooms) do
    forall(j in Days) do
      forall(n in Hours) do
        if a(i,j,n)<>0 AND NewData(a(i,j,n),11)<>"" AND NewData(a(i,j,n),11)<>b(i,1) then rroom+=1;
        end-if

        if (n=6 AND a(i,j,n)<>0) then o+=1;
        elif (n=11 AND a(i,j,n)<>0) OR (n=12 AND a(i,j,n)<>0) then p+=1;
        elif (n=13 AND a(i,j,n)<>0) OR (n=14 AND a(i,j,n)<>0) then q+=1;
        end-if
      end-do
    end-do
  end-do
  PRoom:=rroom*CRoom;
  PHours:=o*CHours13_14+p*CHours18_20+q*CHours20_22;
end-procedure

```

The last procedure calculates the capacity constraint of each course assigned to a classroom. To do so, a new array is created. This array gathers information about each hour of course, its location and its capacity, and its number of students enrolled. Finally, the procedure calculates the delta between the capacity of the room and the number of students enrolled. If the room is too small, the delta is multiplied by 10 in order to have a higher penalty associated with this situation.

```

declarations
  st,room,audit,class,Cap: integer;
  RoomAssignedToCourseQ1: array(1..(MaxRooms*MaxDays*MaxHours),1..5) of string;
  RoomAssignedToCourseQ2: array(1..(MaxRooms*MaxDays*MaxHours),1..5) of string;
end-declarations

procedure CapacityConstraint(a:array(1..(MaxRooms*MaxDays*MaxHours),1..5) of string, b:array(1..(MaxRooms*MaxDays*MaxHours)) of integer,
  c:array(Rooms,1..4) of string)
  class:=1;
  forall(j in Rooms) do
    audit:=1;
    room:=1;
    while(audit<=(MaxDays*MaxHours)) do
      if room<=MaxDays*MaxHours then
        a(class,3):=c(j,1);
        a(class,4):=c(j,2);
      else
        a(class,3):="None";
        a(class,4):="0";
      end-if
      class+=1;
      room+=1;
      audit+=1;
    end-do
  end-do

  st:=1;
  forall(i in 1..(MaxRooms*MaxDays*MaxHours)) do
    a(i,1):=string(b(i));

    if b(i)<>0 then
      a(i,2):=NewData(st,5);

      if i<(MaxRooms*MaxDays*MaxHours) AND b(i+1)<>b(i) then st+=1;
      end-if
    elif b(i)=0 then a(i,2):="0";
    end-if

    if b(i)<>0 AND integer(a(i,2))<=integer(a(i,4)) then
      a(i,5):=string(-(integer(a(i,2))-integer(a(i,4))));
    elif b(i)<>0 AND integer(a(i,2))>integer(a(i,4)) then
      a(i,5):=string((integer(a(i,2))-integer(a(i,4)))*10);
    elif b(i)=0 then a(i,5):="0";
    end-if
  end-do

  Cap:=sum(i in 1..(MaxRooms*MaxDays*MaxHours))integer(a(i,5));
end-procedure

```

## Appendix 8 - Random Search Mosel code

This procedure executes a Random Search. The first operation is to select an origin (x, y and z). This origin must be a not null cell. After that, the course length is calculated. Then the destination course of the same length is selected (r, s and t). If the origin and the destination do not appear in their tabu list, we update them and swap the cells. We calculate the penalty function and if the result is better, we save the swap. Otherwise we swap again to go back to the initial situation. We can notice that for the destination we avoid the three last rooms: they are computer rooms in Mons. By doing this we avoid that a normal course is assigned to a computer room.

```

procedure RandomSearch(TT:array(Rooms,Days,Hours) of integer,RATC:array(1..(MaxRooms*MaxDays*MaxHours),1..5) of string,
                      HPC:array(1..(MaxRooms*MaxDays*MaxHours)) of integer, CR:array(Rooms,1..4) of string, quad:string)
  declarations
    ct: integer;
    w_size,u_size: integer;
    a,b,c,d: integer;
    temp: integer;
    w_campus,u_campus: string;
    LIMIT_RS=250000;
  end-declarations

  ct:=0;
  tbc:=0;

  repeat
    ct+=1;
    if tbc=tb then tbc:=1;
    else tbc+=1;
    end-if

    check:=0;
    while(check=0) do
      w:=0;
      while(w=0) do
        x:=integer(round(0.5+(random*MaxRooms)));
        y:=integer(round(0.5+(random*MaxDays)));
        z:=integer(round(0.5+(random*MaxHours)));
        w:=TT(x,y,z);
      end-do

      w_size:=1;
      w_campus:=NewData(w,8);
      a:=0;
      if z>1 then
        while(z-a-1>=1 AND TT(x,y,z-a-1)=w) do
          w_size+=1;
          a+=1;
        end-do
      end-if
      b:=0;
      if z<14 then
        while(z+b+1<=MaxHours AND TT(x,y,z+b+1)=w) do
          w_size+=1;
          b+=1;
        end-do
      end-if

      u_size:=0;
      u_campus:="";
      while (u_size<>w_size OR w_campus<>u_campus) do
        r:=integer(round(0.5+(random*(MaxRooms-3))));
        s:=integer(round(0.5+(random*MaxDays)));
        t:=integer(round(0.5+(random*MaxHours)));
        u:=TT(r,s,t);

        if u<>0 then u_campus:=NewData(u,8);
        elif u=0 AND r<=MaxRoomsLLN then u_campus:="LLN";
        elif u=0 AND r>MaxRoomsLLN then u_campus:="MONS";
        end-if
      end-while
    end-while
  end-repeat

```

```

u_size:=1;
d:=0;
c:=0;
if (u=0 AND (MaxHours-t)<w_size) then
  forall(k in 1..(MaxHours-t) | TT(r,s,t+k)=u) do      ! On regarde au-dessus
    u_size+=1;
    d+=1;
  end-do
  if u_size<w_size then
    forall(k in 1..(w_size-u_size) | TT(r,s,t-k)=u) do  ! On regarde en-dessous
      u_size+=1;
      c+=1;
    end-do
  end-if
elif (u=0 AND (MaxHours-t)>=w_size) then
  forall(k in 1..(w_size-1) | TT(r,s,t+k)=u) do        ! On regarde au-dessus
    u_size+=1;
    d+=1;
  end-do
else
  if t>1 then
    while(t-c-1>=1 AND TT(r,s,t-c-1)=u) do
      u_size+=1;
      c+=1;
    end-do
  end-if
  if t<MaxHours then
    while(t+d+1<=MaxHours AND TT(r,s,t+d+1)=u) do
      u_size+=1;
      d+=1;
    end-do
  end-if
end-if
end-do
TabuCheck(x,y,z,r,s,t)
end-do

TabuORIGIN(tbc,1):=x;
TabuORIGIN(tbc,2):=y;
TabuORIGIN(tbc,3):=z;
TabuDESTINATION(tbc,1):=r;
TabuDESTINATION(tbc,2):=s;
TabuDESTINATION(tbc,3):=t;

forall(i in 0..w_size-1) do
  temp:=TT(r,s,t-c+i);
  TT(r,s,t-c+i):=TT(x,y,z-a+i);
  TT(x,y,z-a+i):=temp;
end-do

VectorizingTimetable(HPC,TT)
Prof_Student_Penalties(TT)
RequiredRoom_Hours_Penalties(TT,CR)
CapacityConstraint(RATC,HPC,CR)

if quad="Q1" then
  penalties_swapQ1:=PProf+Cap+PHours+PStud+PRoom+PCampus;

  if penalties_swapQ1<=penalties_finalQ1 then
    penalties_finalQ1:=penalties_swapQ1;
  else
    forall(i in 0..w_size-1) do
      temp:=TT(x,y,z-a+i);
      TT(x,y,z-a+i):=TT(r,s,t-c+i);
      TT(r,s,t-c+i):=temp;
    end-do
  end-if
elif quad="Q2" then
  penalties_swapQ2:=PProf+Cap+PHours+PStud+PRoom+PCampus;

  if penalties_swapQ2<=penalties_finalQ2 then
    penalties_finalQ2:=penalties_swapQ2;
  else
    forall(i in 0..w_size-1) do
      temp:=TT(x,y,z-a+i);
      TT(x,y,z-a+i):=TT(r,s,t-c+i);
      TT(r,s,t-c+i):=temp;
    end-do
  end-if
end-if
until(ct=LIMIT_RS)
end-procedure

```

## Appendix 9 – Tabu check Mosel code

```

declarations
  tb:=300;
  tbc: integer;
  TabuORIGIN: array(1..tb,1..3) of integer;
  TabuDESTINATION: array(1..tb,1..3) of integer;
  check: integer;

  w,x,y,z: integer;
  r,s,t,u: integer;
end-declarations

procedure TabuCheck(Origin_room:integer,Origin_day:integer,Origin_hour:integer,Dest_room:integer,Dest_day:integer,Dest_hour:integer)
  forall(i in 1..tb) do
    if (TabuORIGIN(i,1)=Origin_room AND TabuORIGIN(i,2)=Origin_day AND TabuORIGIN(i,3)=Origin_hour) OR (TabuDESTINATION(i,1)=Dest_room
      AND TabuDESTINATION(i,2)=Dest_day AND TabuDESTINATION(i,3)=Dest_hour) then
      check:=0;
      break;
    else
      check:=1;
    end-if
  end-do
end-procedure

```

This procedure is the one that checks if the two selected points are in the tabu lists. Two tabu lists are used: one for the origin points and one for the destination points. The size of this list was hard to define. Since the procedure has to run for both semesters the size should be adapted to both situations. It was decided to put it at 300, but it could go until 377 since in the second semester the number of hours to teach equals 378. Since it is the lowest number of hours in a semester, we had to adapt to this semester.

## Appendix 10 – Room assignment Mosel code

This procedure assigns the courses requiring a specific room to this room after executing Random Search.

```

procedure AssignRequiredRoom(a: array(Rooms,Days,Hours) of integer, b:array(Rooms,1..4) of string)
  declarations
    temp: integer;
    l,m: integer;
  end-declarations

  forall(i in Rooms) do
    forall(j in Days) do
      forall(n in Hours) do
        if a(i,j,n)<>0 AND NewData(a(i,j,n),11)<>"" AND NewData(a(i,j,n),11)<>b(i,1) then
          m:=integer(NewData(a(i,j,n),3));

          forall(k in Rooms) do
            if b(k,1)=NewData(a(i,j,n),11) then
              if a(k,j,n)=0 then
                if m=1 then
                  temp:=a(k,j,n);
                  a(k,j,n):=a(i,j,n);
                  a(i,j,n):=temp;
                elif m=2 then
                  if n<MaxHours AND a(k,j,n+1)=0 then
                    forall(rr in 0..m-1) do
                      temp:=a(k,j,n+rr);
                      a(k,j,n+rr):=a(i,j,n+rr);
                      a(i,j,n+rr):=temp;
                    end-do
                  end-if
                elif m=3 then
                  if n<MaxHours-1 AND a(k,j,n+1)=0 AND a(k,j,n+2)=0 then
                    forall(rr in 0..m-1) do
                      temp:=a(k,j,n+rr);
                      a(k,j,n+rr):=a(i,j,n+rr);
                      a(i,j,n+rr):=temp;
                    end-do
                  end-if
                elif m=4 then
                  if n<MaxHours-2 AND a(k,j,n+1)=0 AND a(k,j,n+2)=0 AND a(k,j,n+3)=0 then
                    forall(rr in 0..m-1) do
                      temp:=a(k,j,n+rr);
                      a(k,j,n+rr):=a(i,j,n+rr);
                      a(i,j,n+rr):=temp;
                    end-do
                  end-if
                end-if
              elif a(k,j,n)<>0 AND NewData(a(k,j,n),11)<>b(k,1) then
                l:=integer(NewData(a(k,j,n),3));

                if l=m then
                  if n>1 AND a(k,j,n-1)<>a(k,j,n) then
                    forall(rr in 0..m-1) do
                      temp:=a(k,j,n+rr);
                      a(k,j,n+rr):=a(i,j,n+rr);
                      a(i,j,n+rr):=temp;
                    end-do
                  end-if
                end-if
              end-if
            end-if
          end-do

          if a(i,j,n)=0 then break;
        end-if
      end-do
    end-do
  end-do
end-procedure

```

## **Appendix 11** – Display timetable by year Mosel code

```

procedure Timetable_by_YEAR(a:string, b:string, c:string, d:array(Hours,Days) of string, e:array(Rooms,Days,Hours) of integer,
    f:array(Rooms,1..4) of string, g:string)
    forall(i in Rooms) do
        forall(j in Days) do
            forall(n in Hours | e(i,j,n)<>0 AND NewData(e(i,j,n),7)=a AND (NewData(e(i,j,n),9)=b OR NewData(e(i,j,n),9)="INGE/GEST")
                AND NewData(e(i,j,n),10)=c AND NewData(e(i,j,n),8)=g) do
                d(n,j):=NewData(e(i,j,n),2)+" - Local: "+Classrooms(i,1)+" - Prof: "+NewData(e(i,j,n),4);
            end-do
        end-do
    end-do
end-procedure

```

This last procedure is used to display the room timetable as little timetables for each year/section/option. It provides information about the course name, its classroom and the teacher teaching it.