Course Scheduling Optimization at HEC: Application of Mathematical Models and Scheduling Algorithm to Minimize Course Conflicts

Auteur : Delsaux, Julie
Promoteur(s) : Paquay, Célia
Faculté : HEC-Ecole de gestion de l'Université de Liège
Diplôme : Master en ingénieur de gestion, à finalité spécialisée en Supply Chain Management and Business Analytics
Année académique : 2023-2024
URI/URL : http://hdl.handle.net/2268.2/20229

COURSE SCHEDULING OPTIMIZATION AT HEC: APPLICATION OF MATHEMATICAL MODELS AND SCHEDULING ALGORITHM TO MINIMIZE COURSE CONFLICTS

Master thesis by
Julie DELSAUX
For a Master in Business Engineering with a specialization in Supply Chain Management and Business Analytics
Academic year 2023/2024

Jury :
Supervisor :
Célia PAQUAY
Reader:
Marie BARATTO

1. Acknowledgments

I would like to express my gratitude to the following people who played a major role in this thesis-making process:

First, I would like to thank my supervisor, **Ms. Célia Paquay** for her help and guidance. She was always available to answer my questions and to give me feedback on my advancements. Her support was especially appreciated during the creation of the mathematical models as she helped me whenever I was facing obstacles to create correct constraints. Moreover, she shared with me a useful book that helped me create an algorithm.

Ms. Marie Baratto deserves my gratitude. Not only she taught me how to code in Julia and solve problems by using Gurobi when I was in my first year of my Master's, but also helped me with the implementation of my mathematical models into coding, took the time to revise my code with me to detect the potential issues and explained to me how to renew my Gurobi License.

I would like to acknowledge **Mr. Yves Crama** for teaching me how to design algorithms during my first year of my master's at HEC and sharing with me his knowledge to have a more reflective view of my work to continuously improve my algorithms.

I am grateful that **Ms. Marie-Gabrielle Boxus** answered all my questions concerning the current class scheduling process at HEC. She took the time to explain everything in detail and provided useful databases I used to create the instances tested on my models and algorithm.

I also would like to thank **Ms. Fabienne Fontaine** who shared with me an Excel File containing the course information for bachelor's degree.

My acknowledgments extend to **Véronique, Alain, and Gilles** for giving ideas about how to improve my thesis and to **Pauline and Yoli** for motivation and support.

Table of contents

2. List of abbreviations / Glossary

HEC: Hautes Études de Commerce – School of Management of the University of Liège

ILP: Integer Linear Programming

NP: Nondeterministic, Polynomial time

RAM: Random-access memory

PhD: Doctor of Philosophy

3. Introduction

Every year, the Faculty of HEC Liège, an educational institution specialized in business and management which is part of the University of Liège, faces the challenge of designing the schedules of the courses over the academic year as well as three exam periods. The first course period is given from September to December and the second one from February to May while the first exam period is given in January, the second one is in June and the last one is given from the end of August until the beginning of September. Course planning is essential to ensure students can navigate through their studies smoothly and to optimize the overall efficiency of university operations.

A course represents a specific subject taught by one or multiple instructor(s) throughout a single semester or the academic year. It is attended by one or more students who are individuals pursuing one or multiple courses.

Before the educational reform called "Décret Paysage" that was implemented in 2014, course planning was quite easy to organize. Anyone who completed all the courses in the courses program could progress to the next bloc of study. As a result, every student from a given bloc had the same curriculum. The curriculum of a student refers to the courses this student must complete and be evaluated at the end of the semester.

In 2015, the reform led to some changes for the Faculty. It brought greater flexibility to students' schedules. They do not need to succeed in all their courses to follow courses from the next bloc. The system is based on the concept of prerequisites and credits. When a course has another course as a prerequisite, it means that the student first has to complete the prerequisite course before having access to the course that has the other course as a prerequisite. The number of credits assigned to a course represents the workload of this course. Generally, it is considered that one credit corresponds to thirty hours of learning. As a result, students can follow courses from different blocs.

Currently, the Faculty designs its schedules by hand by following a specific procedure. First, it contacts all the teachers to know their availabilities and requirements for this year. Once this information is received, it creates a schedule in Excel based on the constraints given by the instructors. After that, it will ask to a program called CELCAT that will automatically associate each course to a specific classroom based on the schedule and the number of students that registered for this course last year. Once it has a final schedule, the planning will be sent to the respective teachers, and they will have the possibility to reach out to the person responsible for the schedule design to expose potential problems. The schedule will be redefined until the best possible solution for a defined period, there is a continuous improvement of the course planning until reaching a solution that satisfies the most each party.

However, the implementation of the "Décret Paysage" made the class scheduling process more difficult. Indeed, as students can now attend courses from different blocs, the number of possible course conflicts increased for students in this case as they could have courses, they are enrolled in, given at the same time. Schedules are still designed per blocs of students which means that they are designed for the students having only courses from one bloc so they have only one course at a time, while the students having courses from many blocs can face course conflicts.

HEC has to face different challenges to create their schedules, and this is a complex problem that takes time, adaptation, and rigor. In this regard, an attempt is made to assign teachers, classrooms, student groups, and time slots to courses.

Teachers are often dedicated to other activities than teaching. They sometimes work in other organizations, do research, are doing a PhD, or teach in other schools. They have their own preferences and constraints that must be carefully taken into account in the planning process. Understanding the value of those constraints and including them in the process is the key to having a schedule that satisfies them. C. Day, A. Kington, G. Stobart and, P. Sammons (2006) affirm that the instructors' satisfaction at work is intimately linked to job motivation, task perception, and future perspective. The constraint of teaching during less preferred hours can affect their work effectiveness and enthusiasm, which will impact the overall educational quality of the concerned institution. This is the same for the way students are willing to learn which is linked to their satisfaction of having chosen the right studies with a schedule that takes their needs into account. Inefficient schedules can slow down students' progress in their journey to degree completion.

Students and their constraints also represent a big part of the scheduling process. The University of Liège and more specifically the Faculty of HEC regroups a large number of students that have their own preferences and educational needs. Some students are following their studies only at HEC, which means that they start studying in the faculty in their first year of bachelor's and finish with a diploma provided by the same faculty while other students transition from one educational entity to another and will maybe spend only a few years at HEC. Most students also have the opportunity to move abroad for a semester and gain an Erasmus experience in another University, which means that HEC will also receive students from other schools that will have specific schedules based on their school of origin. In addition, students' course preferences and classroom preferences may vary.

Not all classrooms are available at any time and once a course is assigned to a specific classroom, this one cannot be used anymore. Other constraints like students' group availability and classroom capacity must be considered for scheduling. When classroom capacity does not align with the number of enrolled students, students suffer from the consequences and may not be able to follow a class due to unavailable chairs where to sit.

The challenge of designing class schedules lies in the vast amount of data. In the case of HEC, there are a total of 21 classrooms within the faculty building and 78 classrooms distributed among the Opéra, 20 Août, and Sart Tilman Quartier Agora buildings. Approximately 213 students are registered for courses in the first bloc of the Business Engineering program, and around 520 students are enrolled in courses for the first bloc of the Economics and Management program. Currently, HEC has approximately 3,500 students, 200 teachers, and 90 courses for the bachelor programs.

Given these challenges, it is crucial to explore how mathematical models and an algorithm, more precisely, Integer Linear Programming models and a First Fit heuristic could improve the scheduling process.

This research aims to address the question: "How can mathematical models and a scheduling algorithm be effectively applied to design course schedules that minimize conflicts while respecting constraints in the Faculty of HEC?"

The purpose is to analyse the use of Integer Linear Programming models and a First Fit heuristic in the class scheduling process and more generally to simplify the lives of class schedulers by reducing the amount of time necessary to design course planning and propose a decision-aiding tool they could use in the future academic years to create the most satisfying class schedule respecting the greatest number of constraints.

The theoretical justification for this research question lies in the significant body of literature related to educational institutions encountering this problem and employing scheduling algorithms and optimization techniques to solve it.

Since 1960, researchers have studied the use of computers to more effective and less time-consuming production school timetable approaches. This was the subject of the research of Appleby, Blake and, Newman (1961) and Murphy and Robert (1964) who stated the necessary data to obtain a class schedule by using a computer. In the most recent years, advanced algorithmic solutions have been designed using integer linear programming or other complex method to solve the timetable problem such as Particle Swarn Optimization employed by Chen and Shih (2013) by focusing on teacher and class preferences.

The focus of this research is to, first, analyse the existing literature and the solution methods that have already been used. Then, the objective will be to find a solution to this class scheduling optimization problem that will be adapted to the Faculty of HEC context. It will involve an interview with the person responsible for class scheduling at HEC to understand the current process.

In this thesis, the purpose is not to provide a solution that will be correct for every educational entity but one that is correct and relevant to the HEC context. Based on the insights obtained during the interview, three Integer Linear Programming (ILP) models will be built, coded in Julia, and then, solved thanks to the informatic solver called “Gurobi” and a First Fit heuristic will be designed and coded in Julia to find solutions for tests made on instances. To code, I used the Visual Studio Code editor.

The first ILP model is the most basic one whose objective is to assign time slots, days, classrooms, the appropriate instructor, and the correct student group to each course while respecting hard constraints to find a solution.

The second ILP model introduces the instructors’ availabilities which were not taken into account in the first ILP model. The number of working days per instructor is also constrained and the requirement that certain courses cannot be taught on the same day is considered. The students are still considered as blocs and the objective of the model is to find a feasible solution.

The third ILP model is the most complete one. It considers students as individuals rather than groups and includes all courses without simplification. This model is the closest to reality and aims to minimize course conflicts for each student by using an objective function.

The First Fit heuristic’s purpose is to assign courses to time slots, rooms, and instructors while satisfying constraints by following several repeated tasks. This algorithm treats courses sorted by their respective instructor’s availabilities, from the lowest to the highest, and then by student groups to place them in the schedule.

After defining and developing the mathematical models and the heuristic, they are tested one by one based on instances. This research focuses on the first semester of the academic year, for the bachelor’s programs in business engineering. It utilizes data from the 2023-2024 academic year, such as the number of students enrolled in the courses, to develop and test these scheduling solutions.

Each test will be decomposed into three main instances:

- Instance considering only the first bloc of business engineering.
- Instance considering the first and the second blocs of business engineering.
- Instance considering from the first to the third blocs of business engineering.

The models are tested to observe the impact of changes within the instances, for example, the decrease in availabilities per instructor. Then the results of the tests are compared to each other and discussed.

Finally, the limitations of each ILP model and the algorithm will be described and used for comparison.

4. Literature Review

Educational institutions have faced the challenge of designing effective timetabling while respecting various constraints and optimizing students' and teachers' preferences. If a schedule is not well-structured and efficient, it can result in difficulties in room utilization, conflicts in the time slots, and dissatisfaction from both students and instructors. The design of timetables is not only responsible for the organization of classes but also for the educational experience lived by both parties.

The main objective of this literature review is to go through the various methods and algorithms that have been proposed to solve the timetabling problem within educational institutions and evaluate them depending on how well they optimize schedules effectively.

In the pursuit of a comprehensive understanding, this review will encompass the different methodologies in chronological order. This will include linear programming approaches that provide mathematically optimized solutions, heuristic methods that offer guidelines for quick scheduling, metaheuristic algorithms that are more adaptable and robust, and innovative hybrid models that combine multiple techniques.

This literature review aims to provide researchers with a large view of what has been already developed in the domain of class scheduling optimization so they can choose more easily what kind of method to use in their case.

4.1. Early Contributions

Appleby, Blake and, Newman (1961) are among the pioneers to describe techniques for producing school timetables on a computer and to compare them to the ones made by human beings. They stated that humans leverage experience to simplify the task and keep an overview of the problem while computers can handle detailed data processing without taking a wider view of the problem. In addition, when humans face challenges to create a feasible timetable by hand, they interchange courses or classrooms which leads to a time-consuming process that may reach 100 hours or more. However, by computer, the person responsible for schedule-making would need only several hours to prepare the data and on average, two hours to produce the timetable.

Among the possible methods of solution proposed by the researchers, one caught my attention: a heuristic approach that would start from a blank timetable which would be filled based on various criteria. The entry of courses inside the timetable could be made randomly, however, this would increase the processing time. This gave me the idea that the heuristic I want to develop should not be made randomly but should follow a strategy and be based on data sorted based on a criterion.

Moreover, the problem is stated with the following data:

- Number of days per cycle.
- Number of periods per day.
- Number and distribution of classes.
- The subjects for each class.
- The number of necessary hours for each subject.
- The allocation of teachers to subjects and classes.

The way the problem stated provides fundamental data necessary to create instances for testing the models and the algorithm. In fact, in the HEC case, the number of days per week is 5, the number of periods per day is 24, each time slot being 30 minutes, from 8 am to 8 pm. Classrooms are assigned based on student enrolment which is determined per course rather than per class. Each class has specific subjects which are given by their respective teacher during a certain amount of time.

Murphy and Robert (1964) not only compared schedules made by hand and by computer but also provided insights about how to implement what they call “Generalized academic simulation programs” (GASP), which is school scheduling by computer, in other schools. In addition to the data Appleby, Blake, and Newman (1961) already treated, Murphy and Robert added other lists necessary to have a more complete model:

- Rooms along with their capacity.
- Instructors’ availabilities and preferences.

As already mentioned earlier, courses have different numbers of students enrolled in them, which means that a room assigned to a course given to a class could not be adapted to another course given to the same class.

By including instructors’ availabilities and preferences, the model also enhances job satisfaction in addition to the prevention of course conflicts.

Barraclough (1965), unlike her previous peers, considered new requirements in her paper describing a method for building school schedules by computer:

- The breaks during the day that should be included in the schedule.
- Courses can be optional, and classes must choose which courses they want to take.

Indeed, in the HEC case, students must be free for lunch as eating is a fundamental need and the more students advance in their curriculum, the more choices they can make in terms of courses, which increases the complexity of the class scheduling process.

In her method, she assigns periods based on binary patterns representing instructors and class availabilities by comparing them and selecting the first period in the week where both instructor and class are available. Then, the entry is inserted into the timetable and the period of availability is reduced for the teacher and student groups.

When there is no common period of availability for both parties, an interchange is operated within the instructor’s timetable to make him available at the same time as the class. If still no matching availabilities are found, an interchange is operated within the class schedule to find periods it is free at the same time as the instructor.

4.2. Complexity and constraints

Even, Itai and Shamir (1976) were among the first to demonstrate the complexity of the class scheduling problem by demonstrating that even a simplified version of the timetable problem is NP-complete, which means that they can have their solutions verified in polynomial time. They started by defining a finite set of hours per week, a collection of hours during which the teachers are available, a collection of hours during which classes are available, and a matrix presenting the number of hours a teacher is required to teach to a class.

In their mathematical model, they set four constraints to define a timetable problem :

- A course can be given only if both the teacher and the class are available.
- The number of courses given during the week between the teacher and the class is the correct one.
- A class cannot have more than one teacher at a time.

- A teacher can teach only one class at a time.

Then, they also defined a restricted timetable problem by fixing the number of hours per week to 3, the classes are always available, teachers are either available for 2 or 3 hours and each teacher is required to teach specific classes for one hour. Their objective was to demonstrate that the timetable and the restricted timetable were NP class.

What I found particularly interesting were the four constraints included in the timetable problem. These constraints are fundamentals to ensure a feasible schedule without any conflicts and provide a basis to apply in my mathematical models and algorithm.

4.3. Optimization Methods

Tripathy (1984) developed a solution method for timetable problems based on Lagrangean relaxation to solve small grouping problems. The Lagrangean relaxation is a mathematical method for solving complex optimization problems by relaxing some of the constraints. The groups were created in two phases to reduce the complexity of the scheduling problem. The courses and rooms are selected in the first phase while the time slots are assigned in the second phase. The method also incorporates branch and bound procedures which benefit from the special ordered set of variables. By using the branch and bound method, the problem is split into smaller subproblems, and the algorithm solves them while keeping track of the best solution obtained so far. The bounds are used to cancel the subproblems that are guaranteed not to contain the optimal solution.

By using a two-phase approach, the complexity of the problem is reduced as the assignation of classrooms and time slots are not made at the same time. In addition, this could have also been made in the other sense, by assigning first time slots and then classrooms. However, in both cases, the students, instructors, and classrooms' availabilities must be taken into account.

Mulvey (1982) proposed a network-based optimizing method to deal with the challenge of underutilized university classroom space. The objective is to make more efficient use of classroom resources and schedules.

If the only condition concerning the assignation of the classroom to a student group is that the classroom must have enough capacity to accommodate the number of students present in the group, there is a possibility of assigning big rooms to small groups of students while these rooms could be used by bigger classes, which would result in underutilization of classroom space. To address this problem in an algorithm, this would be interesting to sort the list of classrooms by capacity so that, when the algorithm tries to assign a classroom to a group of students, this would be the first available classroom having a capacity slightly higher than the number of students of the class.

Dinkel, Mote, and Vekataramaman (1989) also used a network-based approach to solve the academic course scheduling problem based on Mulvey's model, but they integrated a penalty function within a network optimization framework. The penalty function aims at quantifying the penalty associated with deviations from the constraints and preferences.

Aubin and Ferland (1989) also used penalty terms to manage conflicts and excessive classroom use in their research. To do so, they divided the timetabling problem into two subproblems. The first part consists of creating a timetable considering student registrations, teachers' availability, and classroom resources while the second part consists of grouping students for large courses requiring multiple weekly sessions. The researchers took into consideration the link between both subproblems to provide a solution that respects as much as possible the constraints by including penalty terms in their objective function.

While designing a mathematical model presenting a lot of soft constraints, a coherent objective function is to minimize the number of penalties linked to non-respected constraints. For example, if the objective is to have no course conflicts, a constraint could be created to count the number of times there are course conflicts, and this count should be minimized. In the case there are no course conflicts, the objective value will be 0, otherwise, it will be strictly positive and mean that there are course conflicts.

4.4. Heuristic and Metaheuristic approaches

Minimizing course conflicts was the objective of Hertz (1991) when he developed two heuristics based on the Tabu search method to handle grouping and timetabling subproblems. The Tabu search technique was created by Glover (1986), it was designed to use memory structures called tabu lists to keep track of previous solutions to avoid reconsidering them in the future and the solution being trapped in a local optimum. This method includes solutions found by exploring the neighbourhood of the previous solutions and keeping the best ones. Its primary aim is to minimize conflicts between courses that have the same teachers, students, or classrooms. Hertz introduced its two heuristics to handle both timetabling and grouping problems. Then, he added more constraints such as the variability of the course's length.

While designing a heuristic, it is important to find a way of not considering many times the same schedules structures, otherwise, the algorithm would process the data indefinitely even if there are no possible solutions.

Mooney, Darden and, Parameter (1996) introduced a new approach called "CHRONOS" to solve timetabling problems. This algorithm creates initial schedules based on teachers' and students' preferences while avoiding scheduling conflicts based on the "what-if" modelling. The "what-if" modelling is a method used in decision-making and planning processes to analyse hypothetical outcomes and consequences of different choices that can be made. The technique continuously improved the quality of the planning while respecting the constraints. This system was implemented at Purdue University and provided a solution for scheduling 500 courses across 31 large classrooms.

The "what-if" modelling could be a great tool when not all courses have feasible time slots in the case of an algorithm. For example, "what if a course X has other time slots?", "what if this course is given in another classroom?".

Landa Silva (2003) focused on the space allocation problem while satisfying constraints. To do so, iterative improvement, simulated annealing, tabu search, as Hertz (1991), and genetic algorithms were used to provide qualitative solutions in a faster way than manual methods. Iterative Improvement is a method that starts with an initial solution and then, explore the neighbouring solutions to find slightly better solutions until no improvements can be found. Simulated Annealing is a probabilistic optimization method that starts with an initial solution and explores a close solution, but, in contrast to Iterative Improvement, it can accept worse solutions to avoid being trapped in local optima. The more the algorithm works, the lower is the probability of accepting worse solutions. Genetic Algorithms are based on the principle of natural selection. It uses a population of hypothetical solutions and applies genetic operators to create new generations. The algorithm evolves until reaching the termination conditions that are defined. Yazdani, Naderi, and Zeinali (2017) only used three metaheuristics to solve the mathematical model using linear integer programming that they developed. The metaheuristics were artificial immune, genetic, and simulated annealing algorithms. Artificial Immune is a system that creates modifications of its initial solutions through iterations in response to the constraints.

Pinedo (2009) proposed a First Fit heuristic to treat exam scheduling. The algorithm starts are the beginning of the activity lists and checks if the activity fits the first time slot. If this is the case, it is added to the schedule, otherwise, the procedure checks if it fits in the second time slot and so on until finding a time slot in which the activity can fit. Once the activity fits a time slot, the next activity is considered.

I found this heuristic particularly interesting as it treats the courses and the time slots in a specific order. Moreover, it does not consider many times the same association between a course and a time slot. If there are no possible associations, the algorithm moves to the next time slot or course once every time slot has been considered. Additionally, it handles conflicts by checking at each iteration if the time slot is feasible or not for the course.

4.5. Advanced Algorithmic Solutions

Wasfy and Aloul (2007) discussed the idea of using integer linear programming (ILP) to solve the University Class Scheduling Problem. This involved assigning courses to classrooms while considering the classrooms' capacities and university regulations. To solve this problem, they used ILP solvers that included genetic algorithms and Boolean Satisfiability techniques. Boolean Satisfiability problem involves determining whether or not there exists an assignment of truth values (true or false) to set variables that satisfy a given Boolean formula. They also used CPLEX to optimize space utilization without violating the most important constraints. CPLEX is a high-performance commercial software tool developed by IBM to solve complex optimization problems.

Samiuddin and Haq (2019) formulated a mathematical model via Binary Integer Linear Programming to which they applied the data collected. They optimized the model by using the Simplex method to optimize classroom utilization and stakeholders' preferences while respecting constraints. Chen, Bayanati, Ebrahimi, and Khalijian (2022) developed an integer model in which the stakeholders' satisfaction is what should be maximized. They used a genetic algorithm to solve this model which improved the performance of the algorithm.

Chen and Shih (2013) employed Particle Swarm Optimization to solve the timetable problem. In this application, they did not consider any constraints concerning room capacities however, they preferred focusing on teacher and class preferences. Particle Swarm Optimization is an algorithm that maintains a population of potential solutions, called particles, that move to find optimal configurations. In the case of the University Class Scheduling Problem, the particles are potential timetable solutions.

This comprehensive literature review has explored a wide array of methods and algorithms to address the challenge of class scheduling optimization within educational institutions. By examining the different approaches, it is obvious that effective timetabling is important for the proper functioning of schools and universities and to provide the best educational experience possible.

From the early computed methods by Appleby, Blake, and Newman (1961) to advanced algorithms, the class scheduling problem faces numerous constraints and there is a lot of different ways to face them.

The foundation has been placed by pioneers like Appleby and al. (1961) and Murphy and Robert (1964) who demonstrated the effectiveness of computer-generated solutions compared to hand-made schedules. They also provided essential data such as instructors' availabilities, and the allocation of instructors to courses and classes.

Over the years, even more researchers considered other constraints such as Barraclough (1965) who highlighted the need for breaks and the apparition of optional courses, or Even, Itai and, Shamir (1976) that proposed the implementation of penalty functions.

Heuristic and metaheuristic approaches have made their apparition by providing flexible solutions. For example, Hertz (1991) utilized the Taby Search, while Mooney et al. (1996) used the “what-if” modelling approach or Pinedo (2009) developed the First Fit heuristic in the case of exam scheduling.

The integration of advanced algorithmic solutions like Integer Linear Programming or metaheuristic like Particle Swarm Optimization and Genetic algorithm shows the ongoing innovation in the treatment of the problem.

Overall, this literature review provides a comprehensive overview of the diverse approaches to class scheduling and is useful to develop other solutions that could be more adapted to HEC case.

5. Methodology

5.1. Methodological Approach

This research will use a mixed-methods approach, employing both qualitative and quantitative design. The objective is to gain an in-depth understanding of the current scheduling processes at HEC and to test mathematical models and algorithms to design class schedules.

The qualitative aspect involves understanding the current scheduling process at HEC, while the quantitative aspect includes testing the mathematical models and algorithms using real data.

5.2. Methods of data collection

Qualitative data has been gathered thanks to a semi-structured interview with Ms. Boxus, the program manager at HEC. Before conducting the interview, I prepared a few questions I wanted to address with her to collect as much information as possible and to have a thread line. This interview aimed to collect insights about the current scheduling process and understand better the challenges faced, the constraints encountered, and the main objectives while designing course timetable.

Concerning the quantitative data, it has been obtained from the academic personnel who shared with me the following documentation :

- A list of the classrooms available for the whole university “*Liste des salles*”, sorted by building provided by Ms. Boxus.
- The 2023-2024 list of courses given to bachelor students “*Course bac HEC 2023-2024*”, along with the course’s code, its title, the semester the course should be given, the bloc of students it is taught to, and the number of students enrolled in it, provided by Ms. Fontaine, data architect and data analyst of the operational excellence program, she also works at HEC.
- The course program “*Bachelier en ingénieur de gestion*” available on HEC Liège Website. This program provides the instructors related to the courses, the necessary hours per course, and the choices students have to make among courses.

However, I did not have access to two missing documents. The first one is the one providing instructors’ availabilities and the second one is the precise course enrolment for each student, for confidentiality reasons, Ms. Boxus was not able to share it with me.

To face this challenge, I implemented instructors’ availabilities based on the other results given by instances with full availabilities by removing setting the time slots assigned, in the previous results, to their respective course from their availabilities.

Concerning the precise course enrolment for each student, this document was only useful for the third ILP model that considers students as individuals, I then decided to create students that would be enrolled in some courses and not in other courses.

5.3. Methods of analysis

To analyse the qualitative data collected from the interview, I used a thematic analysis which identified themes and patterns relating to scheduling challenges, constraints, and objectives to include in my mathematical model and algorithm.

Quantitative data will be analysed using Integer Linear Programming (ILP) models and a First Fit heuristic which will be coded in Julia on Visual Studio Code.

An incremental approach was adopted in testing designing the ILP models and in testing the instances on the mathematical models and the algorithm.

The first step of designing an ILP model is to define necessary indices and variables and after analysing the qualitative data, state the constraints and objective function that should be also mathematically created. In other words, each constraint stated should have its mathematical correspondence in the model description.

To create a new ILP model, I added new constraints and new variables in to make even more complete but complex mathematical models. The idea was to incorporate the necessary constraints until reaching the third model which is the one closest to reality and which takes into account the "Décret Paysage".

Once the model or the algorithm has been defined, they were tested using quantitative data representing different scenarios. Data from each student's bloc of business engineering were incorporated incrementally to create instances for testing. Additionally, variations in instructor availabilities were introduced to analyse their impact on the outcoming schedules.

Not only I analysed the results obtained for each instance within the same model, but also, I analysed the results obtained by the different models and the algorithm.

Moreover, I analysed the required time for mathematical models to be processed and their objective value.

5.4. Justification of methodological choices

The chosen methodology aligns with the research objectives and desire to understand the complexity of the scheduling problem at HEC. By using a mixed-methods approach, I was able to combine insights from qualitative interviews with quantitative analysis using mathematical models and algorithms.

The use of interview ensures keeping a human point of view over the computation of the class scheduling process, which means that students' needs, such as having a break for lunch, or instructors' preferences such as having a free day per week, are taken into account.

6. Problem Description

To make a fully comprehensive and complete description of the problem of class Scheduling, I contacted Ms. Boxus, the person responsible for the Class Schedule at HEC and she agreed to help me and be interviewed so I could gather all the information I needed to treat the problem as close to the reality as possible.

6.1. Interview

On the 5th of November 2023, I interviewed Ms. Boxus, to understand how she proceeds to create the schedules.

She starts the class scheduling process for the next school year in March. First, she looks at the course program to know if a course is supposed to be given on the first or the second semester, or both, how many theoretical and practical hours are needed, and which instructor(s) is in charge of the course.

Then, she contacts the instructors to ask them about their unavoidable constraints. For example, if an instructor also teaches at another school or works in a company on certain days, their schedule should be arranged to leave free days for these additional activities. She also asks instructors how many courses per week are necessary, in how many blocs, and if practical activities need to be planned, if yes, it is sometimes necessary to divide the classroom into many student groups and give the course many times a week. Even though the number of hours given per course often remains the same from one year to another, it happens that some instructors require changes, or some instructors are replaced by others that have another methodology.

Once she gathered the information needed from the instructors, she encodes holidays and Sundays as unavailable in an Excel file. Then, she starts placing the courses for the first bloc of students. To do so, her objective is to prevent students from travelling from one building to another on the same day to reduce their ecological impact. She also plans lunch breaks of at least 45 minutes to 1 hour and 15-minute breaks between each course to leave some time for students to move from one classroom to another. She also tries to match the sub-courses with each other. For example, if two courses are split into two groups of students, when the first group receives the first course, the second group receives the second course, and vice-versa. She does the same process manually for every bloc of students.

Once the courses are fixed in the schedule, she requests in the program “Celcat”, which finds classrooms from a database, that can be assigned to courses depending on their capacity and availability. However, there are no sufficient classrooms in the HEC building to accommodate every student at the same time. In this case, she makes some requests to other centers such as the Sart-Tilman, the Opera, the Aquarium, or the 20 Août or she changes the schedule, so the course is given at a moment the classroom is available. To know how many students are enrolled in a course and what is the capacity necessary to accommodate them, she takes back the data from the previous year that gives an estimation of the number of students that will be enrolled in the course this year.

If, inadvertently, she encoded many courses at the same time for the same instructor, Celcat signals it.

Once the schedules are made and the classrooms are assigned to courses, she sends them to the teachers so they can notify if the schedule does not correspond to their expectations. If needed, she operates the necessary changes until finding a solution that satisfies the maximum number of instructors while respecting the objective of minimizing the number of trips between buildings for each group of students.

Generally speaking, she avoids planning any courses on the weekend so that students that have a student apartment can go back to their parents' house for the weekend.

She also tries to avoid scheduling courses too early, which means before 9 am, and too late, which means after 6 pm.

For teachers' well-being, she tries to plan breaks of at least 15 minutes between each of their respective courses and at least a free day per week so that they can have time to prepare for their next week's courses or to do some research.

HEC faculty presents a large choice of courses, which why it some courses may be given at the same time and generate frustration among students.

The faculty also proposes transversal programs in collaboration with other schools such as Gramm or with other faculties such as law. These programs are designed last.

Finally, there is the specific case of the course "Ateliers de compétences". This course is split into 74 workshops that students can choose from. To avoid course conflicts between the workshops and the other courses, she plans fixed time slots dedicated only to the workshop. For example, in bachelor's, the Thursday afternoon is dedicated to the workshops' activities while it is the Tuesday for the master students. In addition, some students use the free Tuesday in case they already passed their workshop to work as interns in a company.

6.2. General Description

Every year, HEC Liège faces the challenge of designing class schedules for its diverse range of programs, from the bachelor to the master. While designing students' timetables, the program manager assigns each course to a group of time slots and a classroom while ensuring that the instructor responsible for this course, the students' group following this course, and the classroom are available during these time slots.

Designing students' timetables is a long and robust process that takes many months to complete because the following constraints need to be adhered to:

- An instructor can teach only one course at a time.
- A student group can be given only one course at a time.
- A course should be given for its entire duration during consecutive time slots.
- Only the instructor responsible for a course can teach this course.
- Students should follow only the courses they are enrolled in.
- The classroom assigned to a course must have a capacity higher than the number of students registered for the course.
- Only one classroom should be assigned to its course and for its entire duration.
- Courses should start after 9 am and stop before 6 pm.
- Lunch breaks should be planned.
- A course can be given only at moments its instructor is available.
- An instructor should not work more than four days per week.
- Two sub-courses from the same theoretical bloc or the same practical bloc to the same student group should be given on different days.

Due to the reform "Décret Paysage", students can follow courses from different blocs, and this made the class schedule process even more complicated as it increases the risk of course conflicts per student. The objective is then to reduce the number of course conflicts.

Finally, some courses are split into sub-groups, which means that this course must be given many times throughout the week and to different groups of students, sometimes by the same instructor, sometimes by different ones.

7. The mathematical models

To solve the Class Scheduling Optimization problem, I designed an Integer Linear Programming (ILP) model. This model was designed based on the description of the problem and the data provided by both Ms. Boxus and Ms. Fontaine. The model has been formulated in three phases.

The first phase was to make the most basic model. In this model, classrooms, days of the week, time slots, instructors, student groups, and courses are considered. The objective is to attribute time slots, a day, a classroom, the right instructor, and the right student group to each course while respecting hard constraints, and then, finding a feasible solution.

The second model goes further by taking into account new constraints such as the instructors' availability, the maximum number of working days per instructor, and the fact that some courses cannot be given within the same day. The objective is the same as for the first model which means to find a feasible solution that satisfies every constraint.

Lastly, the third model is the most complex and complete one as students are considered as individuals and not as a group anymore and all courses are considered and not simplified anymore. This model's objective is to minimize the number of course conflicts for every student and then, to find an optimal solution due to the existence of an objective function. In fact, due to the "Décret Paysage" students can be enrolled in courses from different student blocs. However, this possibility can generate course conflicts.

Each model is composed of various indices, parameters, decision variables, and a set of constraints aimed at finding a feasible scheduling solution.

7.1. First model

To design the first model, I considered the most basic information I gathered to generate a sufficiently coherent solution that respects the reality of the HEC class scheduling case. This means that, for this model, I had to take into account the following constraints :

1. Each course has a specific duration attributed. Whenever a course has a starting timeslot attributed, it means that this course must be given from the starting time slot until the end of its duration. In other words, a course with a specific duration must be conducted in consecutive time slots. For example, if a course must be given during 4 time slots, knowing that each time slot lasts 30 minutes, this course cannot be given from 9 am to 10 am and from 1 pm to 2 pm, otherwise, it would be separated into non-consecutive time slots.

2. A course can only be given by the instructor that is responsible for this course, otherwise, every teacher could teach any course. However, this is not the case, the instructors have their own competencies and courses to give, and they cannot exchange courses between themselves.

3. At HEC, courses are attributed to a specific group of students. When students enter the first year at the university, they must pass some courses before having access to courses in the second year. The model should ensure that the student groups are assigned only courses designated to their enrolment.

4. A course must be given exactly once a week. In the case of a course given many times a week, it will be considered under two different names to make the distinction between the different

blocs of this course. For example, if a course consists of theoretical, practical, and remediation blocs, those three blocs must be considered as three different courses even though they have the same instructor.

5. Each course has a specific number of students registered to their course. The model must ensure that, when a classroom is assigned to a course, this classroom has enough capacity to accommodate all the students registered for the course.

6. An instructor can only give one course at a time. Otherwise, course conflicts for the instructor could happen.

7. A group of students can only be given one course at a time, otherwise, course conflicts for the group of students could happen. However, this constraint applies only when considering students as a group. In fact, due to the Décret Paysage, it is allowed for a student to have courses from different student groups which means that a student could have courses from different student blocs. This reform increases the risks of scheduling conflicts for students, which is why the objective of the third model is to minimize the number of course conflicts for every student. I will get back to this while introducing the third model.

8. A course cannot start so late that it would not have the time to last for its whole duration. In other words, a course must start early enough to ensure that consecutive time slots are available for its entire duration. Otherwise, the course could be interrupted before being completed.

9. Only one classroom can be attributed to a course. Otherwise, a course could have different classrooms from one time slot to another, which would mean that students would need to move from one class to another for the same course.

Based on those constraints I designed the first model.

Indices

- r = 1,…,R for the classrooms
- d = 1,…,D for the days of the week (where D=5)
- t = 1,…,T for time slot (where T=24 as 1 is from 8h to 8h30, 2 is from 8h30 to 9h, 3 is from 9h to 9h30, …24 is for 19h30 to 20h).
- i = 1,…,I for instructors
- s = 1,…,S for student groups
- c = 1,…,C for course blocs that must be given

Parameters

- $student\_group\_size(c, s)$: $size\ of\ student\ group\ s\ for\ course\ c$
- $room\_capacity(r)$: $capacity\ of\ classroom\ r$
- $Duration(c)$: $duration\ of\ the\ course\ c\ in\ number\ of\ time\ slots$ *(for example, if the course c lasts 1 hour and 30 minutes, the duration will be 3.)*
- *Assoc(s,c): binary parameter that takes value 1 if the group of students s follows the course c, 0 otherwise.*
- *Comp(i,c): binary parameter that takes value 1 if the course c is given by instructor I, 0 otherwise.*

Decision Variables

- $X(c,i,t,d,r,s)$: binary variable that takes the value 1 if the course c is provided by the ith instructor at the time slot t of day d in the classroom r and involves sth student group.
- Y(c, t, d): binary variable that takes the value 1 if the course c starts at time slot t of day d.
- Z(c,r): binary variable that takes the value 1 if the course c is given in classroom r.

Objective function

No objective function for this ILP model version as the objective is to find a feasible solution.

Constraints

- $Y(c,t,d) \leq \sum_{i=1}^{I}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t',d,r,s) \quad \forall t' \in \{t, \dots, t + duration(c) - 1\}, \quad \forall c,d,t$ ensures that, if course c starts at time slot t of day d, the course must be scheduled for its entire duration starting from its starting time slot t. (C1)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq Comp(i,c) \times duration(c) \;\; \forall c,i$ ensures that course c can only be taught by the corresponding instructor i to one student's group in one room on the same day. This constraint ensures that teachers can only provide courses for which they are competent. (C2)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{i=1}^{I} X(c,i,t,d,r,s) \leq Assoc(s,c) \times duration(c) \;\; \forall c,s$ ensures that course c can only be given to the corresponding students' group s by one instructor I in one room on the same day. This constraint ensures that students can only follow classes for which they registered. (C3)

- $\sum_{t=1}^{T}\sum_{d=1}^{D} Y(c,t,d) = 1 \;\; \forall c$ ensures that the course c must start once and only once a week. In other words, this constraint ensures that a course is given exactly once a week. (C4)

- $\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) \times student_{group_{size}}(c,s) \leq room_{capacity}(r) \;\; \forall r,t,d,c$ ensures a room r reserved at time slot t during day d has enough capacity to welcome students' group s enrolled in course c in that classroom at that time. (C5)

- $\sum_{c=1}^{C}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq 1 \;\; \forall i,t,d$ ensures that an instructor i can give only one course c in one room r to one student group s at a time. (C6)

- $\sum_{c=1}^{C}\sum_{i=1}^{I}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq 1 \;\; \forall t,d,s$ ensures that a student group can only be given one course c by one instructor i in one room r at a time. (C7)

- $\sum_{t=T-duration(c)+1}^{T} Y(c,t,d) = 0 \;\; \forall c,d$ ensures that a course cannot start at a timeslot for which the course cannot be scheduled for its entire duration. (C8)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times Z(c,r) \;\; \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes. (C9)

- $\sum_{r=1}^{R} Z(c,r) = 1 \;\; \forall c$ ensures that only one room can be attributed to a course. In other words, each course is assigned to only one room. (C9)

7.2. Second model

To design the second model, I integrated two new parameters, one new decision variable, and four new constraints to the previous version of the model. However, there is still no objective function, as the objective of this version of the ILP model is to find a feasible solution.

The two new parameters I integrated are:

- The availability of instructors; this parameter takes the value 1 whenever the instructor is available and 0 when not.
- The “type” of two courses, which represents wether two courses should be given on two distinct days or not.

The new decision variable is the one that represents whether an instructor works on a day or not.

Here are the new constraints I integrated into the previous version to build the second model:

1. No courses can be given before 9 am, between 12 am and 1 pm, and after 6 pm. The HEC scheduler tries to avoid planning courses at those times so that students and instructors do not need to be at school too early or too late, and they can have a lunch break. This constraint was already taken into account from the second test of the first model but is now fully integrated into the mathematical model.

2. No courses can be taught by an instructor if this one is not available. In other words, instructors can get courses assigned at some time slots only if they are available at those time slots.

3. An instructor cannot work more than four days per week. The HEC scheduler always tries to leave at least one free day during the week for each instructor. This way, the teachers can have a full day to prepare for their next lessons or, in the case they have other projects or other jobs aside from their teacher job, they can work on that.

4. If two courses are considered as the same “type”, it means that they should not be given within the same day. For example, the courses “Mathématiques: Analyse infinitésimale – 1” and “Mathématiques: Analyse infinitésimale – 2” are considered as the same type as they are both from the same bloc of courses, which is “Mathématiques: Analyse infinitésimale”. Courses of the same type should be given on different days so that students can have the time to assimilate what they have learned from the previous course and not have too many hours of the same course within the same day.

Based on those constraints I designed the second model.

Indices

- $r = 1,\ldots,R$ for the classrooms
- $d = 1,\ldots,D$ for the days of the week (we do not consider the weekend so, we could write $d = 1,\ldots,5$)
- $t = 1,\ldots,T$ for time slot (1 is from 8h to 8h30, 2 is from 8h30 to 9h, 3 is from 9h to 9h30, …24 is for 19h30 to 20h00)).
- $i = 1,\ldots,I$ for instructors
- $s = 1,\ldots,S$ for student groups

- c = 1,…,C for courses blocs that must be given (for example, if the mathematic course must be given 2 times a week, we will have 2 distinct indexes for those 2 blocs)

Parameters

- $student\_group\_size(c, s)$: $size\ of\ student\ group\ s\ for\ course\ c$
- $room\_capacity(r)$: $capacity\ of\ a\ specific\ room\ r$
- $Duration(c)$: $duration\ of\ the\ course\ c\ in\ number\ of\ time\ slots$ *(for example, if the course c lasts 1h30, the duration will be 3.)*
- *Assoc(s,c): binary parameter that takes value 1 if the group of students s follows the course c, 0 otherwise.*
- *Comp(i,c): binary parameter that takes value 1 if course c is given by instructor I, 0 otherwise.*

New parameters :

- *A(i,t,d): binary parameter that takes value 1 if an instructor i is available at timeslot t of day d.*
- *Type(c,c'): binary parameter that takes value 1 if a course c is linked to a course c'.*

Decision Variables

- $X(c, i, t, d, r, s)$: binary variable that takes value 1 if the course c is provided by the ith instructor at time slot t of day d in the room r and the sth student group participate in it.
- Y(c, t, d): binary variable that takes value 1 if the course c starts at time slot t of day d.
- Z(c,r): binary variable that takes value 1 if the course c is given in classroom r.

New decision variable :

- $W(i, d)$: binary variable that takes value 1 if instructor i works on day d.

Objective function

No objective function for this ILP model version as the objective is to find a feasible solution.

Constraints

- $Y(c, t, d) \leq \sum_{i=1}^{I} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c, i, t', d, r, s) \quad \forall t' \in \{t, \dots, t + duration(c) - 1\}, \quad \forall c, d, t$ ensures that, if course c starts at time slot t of day d, the course must be scheduled for its entire duration starting from its starting time slot t.

- $\sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c, i, t, d, r, s) \leq Comp(i, c) \times duration(c) \ \forall c, i$ ensures that course c can only be taught by the corresponding instructor i to one student's group in one room on the same day. This constraint ensures that teachers can only provide courses for which they are competent.

- $\sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} \sum_{i=1}^{I} X(c, i, t, d, r, s) \leq Assoc(s, c) \times duration(c) \ \forall c, s$ ensures that course c can only be given to the corresponding students' group s by one instructor I in one room on the same day. This constraint ensures that students can only follow classes for which they registered.

- $\sum_{t=1}^{T} \sum_{d=1}^{D} Y(c, t, d) = 1 \ \forall c$ ensures that the course c must start once and only once a week. In other words, this constraint ensures that a course is given exactly once a week.

- $\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) \times student_{group_{size}}(c,s) \ \leq room_{capacity}(r) \ \ \forall r,t,d,c$ ensures a room r reserved at time slot t during day d has enough capacity to welcome students' group s enrolled in course c in that classroom at that time.

- $\sum_{c=1}^{C}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq 1 \ \ \forall i,t,d$ ensures that an instructor i can give only one course c in one room r to one student group s at a time.

- $\sum_{c=1}^{C}\sum_{i=1}^{I}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq 1 \ \ \forall t,d,s$ ensures that a student group can only be given one course c by one instructor i in one room r at a time.

- $\sum_{t=T-duration(c)+1}^{T} Y(c,t,d) = 0 \ \ \ \forall c,d$ ensures that a course cannot start at a timeslot for which the course cannot be scheduled for its entire duration.

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times Z(c,r) \ \ \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes.

- $\sum_{r=1}^{R} Z(c,r) = 1 \ \ \ \forall c$ ensures that only one room can be attributed to a course. In other words, each course is assigned to only one room.

New constraints :

- $X(c,i,t,d,r,s) = 0 \ \ \ \forall t \in \{1,2,9,10,21,22,23,24\}, \forall c,i,d,r,s$ ensures that no courses can be given before 9 am, between 12 am and 1 pm, and after 6 pm. (C1)

- $\sum_{c=1}^{C}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq A(i,t,d) \ \ \forall i,t,d$ ensures that an instructor I can only get a course assigned at timeslot t of day d if they are available at that moment. (C2)

- $\sum_{t=1}^{T}\sum_{c=1}^{C} Y(c,t,d) \times comp(i,c) \leq W(i,d) \times T \ \ \ \forall i,d$ ensures that if the instructor i gives at least one course on day d, then, this instructor will be considered as working on that day and the variable W(i,d) will take the value 1. We added T on the right to ensure that an instructor can give more than one course a day. (C3)

- $\sum_{d=1}^{D} W(i,d) \leq 4 \ \ \ \forall i$ ensures that an instructor i cannot work more than 4 days per week. (C3)

- $\sum_{t=1}^{T} Y(c,t,d) \times type(c,c') \ + \ \sum_{t=1}^{T} Y(c',t,d) \times type(c,c') \leq 1 \ \ \ \forall d,c,c'$ ensures that if 2 courses are linked, for example, two courses of maths given by the same instructor, then, the courses cannot be given within the same day. (C4)

7.3. Third model

To design the third model, I changed the indice "s" so that this does not represent student groups anymore, but each s is for a unique student.

In addition, an integer positive variable is added to the model. This decision variable represents the number of course conflicts for a student s, at a timeslot t, on a day d. The objective function is to minimize the sum of this decision variable over every time slot, day, and student.

To define the value of the decision variable I just defined; I integrated a new constraint that ensures that the decision variable takes the value of the number of courses that are conflicting whenever there are scheduling conflicts.

Based on these new elements I designed the third model of the ILP model.

Indices

- r = 1,…,R for the classrooms
- d = 1,…,D for the days of the week (we do not consider the weekend so, we could write d =1,…,5)
- t = 1,…,T for time slot (1 is from 8h to 8h30, 2 is from 8h30 to 9h, 3 is from 9h to 9h30, …24 is for 19h30 to 20h00)).
- i = 1,…,I for instructors
- s = 1,…,S for students
- c = 1,…,C for courses blocs that must be given (for example, if the mathematic course must be given 2 times a week, we will have 2 distinct indexes for those 2 blocs)

Parameters

- $student\_group\_size(c)$: $size\ of\ student\ group\ for\ course\ c$
- $room\_capacity(r)$: $capacity\ of\ a\ specific\ room\ r$
- $Duration(c)$: $duration\ of\ the\ course\ c\ in\ number\ of\ time\ slots$ *(for example, if the course c lasts 1'30", the duration will be 3.)*
- *Assoc(s,c): binary parameter that takes value 1 if the student s follows the course c, 0 otherwise.*
- *Comp(i,c): binary parameter that takes value 1 if course c is given by instructor I, 0 otherwise.*
- *A(i,t,d): binary parameter that takes value 1 if an instructor i is available at timeslot t of day d.*
- *Type(c,c'): binary parameter that takes value 1 if a course c is linked to a course c' and they cannot be given within the same day.*

Decision Variables

- $X(c,i,t,d,r,s)$: binary variable that takes value 1 if the course c is provided by the ith instructor at time slot t of day d in the room r and the sth student participates in it.
- Y(c, t, d): binary variable that takes value 1 if the course c starts at time slot t of day d.
- Z(c,r): binary variable that takes value 1 if the course c is given in classroom r.
- $W(i,d)$: binary variable that takes value 1 if instructor i works on day d.

<u>New decision variable :</u>

- $V(t,d,s)$: integer positive variable that represents the number of course conflicts for student s, at timeslot t on day d.

Objective function

$$Min \sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{s=1}^{S} V(t,d,s)$$

Constraints

- $Y(c,t,d) \times Assoc(s,c) \leq \sum_{i=1}^{I}\sum_{r=1}^{R} X(c,i,t',d,r,s) \quad \forall t' \in \{t, \ldots, t + duration(c) - 1\}, \quad \forall c,d,t,s$ ensures that, if course c starts at time slot t of day d, the course has to be given during the following time slots until reaching the total duration of the course. (C1)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq Comp(i,c) \times duration(c) \quad \forall c,i,s$ ensures that course c can only be given by the corresponding instructor i in one room on the same day. This constraint ensures that teachers can only provide courses for which they are competent. (C2)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{r=1}^{R}\sum_{i=1}^{I} X(c,i,t,d,r,s) = Assoc(s,c) \times duration(c) \ \forall c,s$ ensures that course c can only be given to the student s who registered for this course by one instructor i in one room on the same day. This constraint ensures that students can only follow classes for which they registered. (C3)

- $\sum_{t=1}^{T}\sum_{d=1}^{D} Y(c,t,d) = 1 \ \forall c$ ensures that the course c must start exactly once a week. In other words, this constraint ensures that a course is given exactly once a week. (C4)

- $\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) \times student\_groupe\_size(c) \leq room_{capacity}(r) \ \forall r,t,d,c$ ensures a room r reserved at time slot t during day d has enough capacity to welcome the students that registered to this course with instructor i in that classroom at that time. (C5)

- $\sum_{c=1}^{C}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq \sum_{c=1}^{C}\sum_{s=1}^{S} Assoc(s,c) \quad \forall i,t,d$ ensures that an instructor i can give only one course c in one room r at a time. (C6)

- $\sum_{t=T-duration(c)+1}^{T} Y(c,t,d) = 0 \quad \forall c,d$ ensures that a course cannot start at a timeslot for which it would not be given for all its duration. (C7)

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times \sum_{s=1}^{S} Assoc(s,c) \times Z(c,r) \quad \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes. (C8)

- $\sum_{r=1}^{R} Z(c,r) = 1 \quad \forall c$ ensures that only one room can be attributed to a course. In other words, each course is assigned to only one room. (C9)

- $X(c,i,t,d,r,s) = 0 \quad \forall t \in \{1,29,10,21,22,23,24\}, \forall c,i,d,r,s$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes. (C10)

- $\sum_{c=1}^{C}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq A(i,t,d) \quad \forall i,t,d,s$ ensures that an instructor I can only get a course assigned at timeslot t of day d if they are available at that moment. (C11)

- $\sum_{t=1}^{T}\sum_{c=1}^{C} Y(c,t,d) \times comp(i,c) \leq W(i,d) \times T \quad \forall i,d$ ensures that if the instructor i gives at least one course on day d, then, this instructor will be considered as working on that day and the variable W(i,d) will take the value 1. We added T on the right to ensure that an instructor can give more than one course a day. (C12)

- $\sum_{d=1}^{D} W(i,d) \leq 4 \quad \forall i$ ensures that an instructor i cannot work more than 4 days per week. (C13)

- $\sum_{t=1}^{T} Y(c,t,d) \times type(c,c') + \sum_{t=1}^{T} Y(c',t,d) \times type(c,c') \leq 1 \quad \forall d,c,c'$ ensures that if 2 courses are linked, for example, two courses of maths given by the same instructor, then, the courses cannot be given within the same day. (C14)

New constraint :

- $\sum_{c=1}^{C} \sum_{i=1}^{I} \sum_{r=1}^{R} X(c,i,t,d,r,s) \leq V(t,d,s) + 1 \ \forall t,d,s$ ensures that if there are scheduling conflicts for a specific student, the binary variable V(t,d,s) takes the value of the number of courses that are conflicting at a specific timeslot on a day d. (C15)

In this third model, I had to readapt some constraints so that the model would be accurate as we changed s from student groups to students as individuals.

In constraint C1, I had to add "Assoc(s,c)" so that X could not be equal to 1 if a student is not registered for a course. Indeed, I had to change the first constraint from this :

- $Y(c,t,d) \leq \sum_{i=1}^{I} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c,i,t',d,r,s) \quad \forall t' \in \{t, \ldots, t + duration(c) - 1\}, \quad \forall c,d,t$ ensures that, if course c starts at time slot t of day d, the course must be scheduled for its entire duration starting from its starting time slot t.

Into this :

- $Y(c,t,d) \times \boldsymbol{Assoc(s,c)} \leq \sum_{i=1}^{I} \sum_{r=1}^{R} X(c,i,t',d,r,s) \quad \forall t' \in \{t, \ldots, t + duration(c) - 1\}, \quad \forall c,d,t,\boldsymbol{s}$ ensures that, if course c starts at time slot t of day d, the course has to be given during the following time slots until reaching the total duration of the course.

In constraint C2, I had to remove the indice s from the sum as we are considering students as individuals and not as groups anymore. I had to change the second constraint from this :

- $\sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} \boldsymbol{\sum_{s=1}^{S}} X(c,i,t,d,r,s) \leq Comp(i,c) \times duration(c) \ \forall c,i$ ensures that course c can only be taught by the corresponding instructor i to one student's group in one room on the same day. This constraint ensures that teachers can only provide courses for which they are competent.

Into this :

- $\sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} X(c,i,t,d,r,s) \leq Comp(i,c) \times duration(c) \quad \forall c,i,\boldsymbol{s}$ ensures that course c can only be given by the corresponding instructor i in one room on the same day. This constraint ensures that teachers can only provide courses for which they are competent.

I also had to change the constraint C6 so that the instructor can give only one course at a time but to many students. The sixth constraint changed from this:

- $\sum_{c=1}^{C} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c,i,t,d,r,s) \leq 1 \ \forall i,t,d$ ensures that an instructor i can give only one course c in one room r to one student group s at a time.

Into this :

- $\sum_{c=1}^{C} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c,i,t,d,r,s) \leq \boldsymbol{\sum_{c=1}^{C} \sum_{s=1}^{S} Assoc(s,c)} \ \forall i,t,d$ ensures that an instructor i can give only one course c in one room r at a time. (C6)

I also removed the constraint that ensured that a student group could only have one course at a time. I consider students as individuals now and the objective function is to calculate how many course conflicts there are for every student. Then, I cancelled this constraint for the third model :

- $\sum_{c=1}^{C}\sum_{i=1}^{I}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq 1 \quad \forall t,d,s$ ensures that a student group can only be given one course c by one instructor i in one room r at a time.

In the constraint C8, I had to add the parameter "Assoc(s,c)" so that the second member considers every student concerned by the course too. Then, I changed the eighth constraint from this :

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times Z(c,r) \quad \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes.

Into this :

- $\sum_{t=1}^{T}\sum_{d=1}^{D}\sum_{i=1}^{I}\sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times \sum_{\boldsymbol{s=1}}^{\boldsymbol{S}} \boldsymbol{Assoc(s,c)} \times Z(c,r) \quad \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes. (C8)

Finally, I changed the constraint C11 to adapt it to the fact that I now consider students as individuals and not as groups anymore. I changed the eleventh constraint from this :

- $\sum_{c=1}^{C}\sum_{r=1}^{R}\sum_{s=1}^{S} X(c,i,t,d,r,s) \leq A(i,t,d) \quad \forall i,t,d$ ensures that an instructor I can only get a course assigned at timeslot t of day d if they are available at that moment.

Into this:

$\sum_{c=1}^{C}\sum_{r=1}^{R} X(c,i,t,d,r,s) \leq A(i,t,d) \quad \forall i,t,d,\boldsymbol{s}$ ensures that an instructor I can only get a course assigned at timeslot t of day d if they are available at that moment. (C11)

8. Algorithm

To solve the Class Scheduling Optimization problem, I also designed a First Fit heuristic that I implemented in Julia.

8.1. Initialization

To initialize this algorithm, data structures had to be set up and populated with the necessary information to have a complete model and later, to generate the class schedules.

- **Instructor Availability Matrix**: this matrix whose dimensions correspond to the number of instructors, the number of time slots, and the number of days, stores the availability of instructors. In other words, if an instructor I is available at time slot t on day d, the value in the matrix corresponding to those indices is 1; otherwise, the value is 0.

- **Instructor Availability Dictionary**: this dictionary is used to store the total number of available time slots for each instructor based on the Instructor Availability Matrix.

- **Course Instructor Dictionary**: this dictionary associates each course to its respective instructor.

- **Sorted Courses Vector**: this vector is a list of courses sorted firstly by availability, based on the Instructor Availability Dictionary and the Course Instructor linked Dictionary, so that the courses with the instructor responsible for this course having the least availability are the first ones of the list, and then, the vector is sorted by students' group, from the first bloc of students to the last one.

- **Course Availability Matrix**: this matrix whose dimensions correspond to the number of courses, the number of time slots, and the number of days, stores the availability of courses, based on the availability of their respective instructor. In other words, it is based on the Instructor Availability Matrix but instead of being the availability of the instructors, it is the availability of the courses. The order of the courses in the matrix is the same as in the Sorted Courses Vector. To keep coherence with the mathematical model previously defined, the time slots before 9 am, between 12 am and 1pm, and after 6 pm are considered as unavailable, so their value in the matrix is 0. However, these time slots will potentially be set back to the value 1 in the algorithm but I will get back to this in the "Scheduling" part of the Algorithm description.

- **Sorted Classrooms DataFrame**: This table presents the classrooms along with their respective building and capacity, sorted by capacity, from the lowest to the highest.

- **Room Availability Matrix**: this matrix whose dimensions correspond to the number of classrooms, the number of timeslots, and the number of days, stores the availability of each classroom. In other words, if the classroom r is available at time slot t on day d, the value in the matrix corresponding to those indices is 1; otherwise, the value is 0. At the beginning of the code, all classrooms are available all the time as they have not been assigned to any courses yet, so all the values in the matrix are 1. Within the matrix, the classrooms are ordered from the classroom having the least capacity to the one having the highest capacity.

- **Unscheduled Courses List**: this empty list will be used to keep track of the courses that have not been scheduled due to lack of free time slots.

- **Scheduled Courses Dictionary**: this dictionary stores the courses that have been scheduled along with their assigned day, their starting time slot, their ending time slot, and the classroom associated with it.

- **Type Table**: this table regroups the courses that need to be given on different days. For example, the courses "Physique générale : partim 1 - théorie 1" and "Physique générale : partim 1 - théorie 2" should be given on different days so that students do not have two times the same course on the same day as those two courses are part of the same theoretical bloc.

8.2. Scheduling

To schedule the different courses through the empty timetable, the algorithm proceeds the following way:

1. The algorithm iterates over each course in the Sorted Course Vector and searches for the first consecutive available time slots for the whole duration of the course in the Course Availability Matrix.

2. If there is a group of feasible consecutive time slots, the algorithm iterates over each classroom by order of their capacity, from the lowest to the highest, and checks if the classroom is available at the same time slots as the course and if the capacity of the classroom is higher than the number of students registered to the course. If this is the case, the Room Availability Matrix is updated to set the values linked to the classroom, the feasible time slots, and the day at 0 so that the classroom is not available at these moments anymore. If no feasible consecutive time slots are found, the course is put in the Unscheduled Courses Vector.

3. The course is added to the Scheduled Courses Dictionary along with its starting time slot, its ending time slot, its assigned day, and classroom.

4. The Course Availability Matrix is updated three times :

    a. It is updated for the other courses having the same student group. The values of the courses given to the same student groups as the course that has just been planned are set to 0 for the same time slots and day as this course so that the other courses having the same student group cannot be planned at the same time. This way, student groups cannot be taught many courses at the same time.

    b. It is secondly updated for the courses that have the same instructor as the course that has just been added to the Scheduled Courses Dictionary. The values in the matrix for the courses that have the same instructor are set to 0 for the same time slots and day as the course that has just been planned so that an instructor cannot give many courses at the same time.

c. The Course Availability Matrix is also updated based on the Type Table. If the course that has just been planned is supposed to be given on another day of another specific course, the algorithm updates the availability of the other courses so that the value of the other courses for the day assigned to the main course planned is equal to 0. This way, the two courses cannot be given on the same day.

5. In the case there are courses in the Unscheduled Courses Vector, the algorithm resets the time slots 1, 2, 21, 22, 23, and 24 as available for every course and every day except for the courses that cannot be given on the same day as another course that has already been scheduled.

6. Then, the algorithm iterates through each course that is in the Unscheduled Courses Vector and searches for the first consecutive time slots for the whole duration of the course in the Course Availability Matrix.

7. If there is a group of feasible consecutive time slots, the algorithm iterates over each classroom by order of their capacity, from the lowest to the highest, and checks if the classroom is available at the same time slots as the course and if the capacity of the classroom is higher than the number of students registered to the course. If this is the case, the Room Availability Matrix is updated to set the values linked to the classroom, the feasible time slots, and the day at 0 so that the classroom is not available at these moments anymore. This seventh step is the same as the second step except that this time, we treat the courses that are in the Unscheduled Courses Vector and not in the Sorted Courses Vector. Steps 3 and 4 are made again so that the course is in the Scheduled Courses Dictionary and the Availability Matrix is correctly updated.

8. If no still feasible time slots have been found, the algorithm iterates over each day until finding consecutive time slots that are available for the course even though there are no classrooms available for these time slots.
   Then, it iterates through each classroom until finding a classroom that has a capacity high enough to accommodate the number of students enrolled in the course. The Occupied Courses Vector is created and gathers the courses that occupy the classroom during the consecutive time slots for which the course from Unscheduled Courses is available. Then, it searches for alternative consecutive time slots for which the courses and the classroom as well are available.
   If alternative timeslots are found, the Room Availability Matrix is updated and its value for the alternative time slots is set to 0 while it is set to 1 for the consecutive time slots for which the course from Unscheduled Courses is available.
   Moreover, the Course Availability Matrix is updated so that the courses that have alternative time slots are considered unavailable during their newly assigned time slots and as available during their previously assigned time slots. The Course Availability Matrix of other courses that as the same instructor as the courses that have been moved to new time slots is also updated so that instructors cannot give many courses at the same time and the previously assigned time slots are now considered as available. Finally, it is updated based on the Type Table so that, if the courses that have been moved cannot be given on the same day as another course, the Course Availability Matrix of this other course is set to the value 0 for the day the course has been moved and to 1 for the previous day, the courses were assigned to.

9. If the courses could be moved, the Room Availability Matrix will be updated so that the room is not available anymore during the time slots attributed to the course from Unscheduled Courses and this course will be added to the Scheduled Courses dictionary along with assigned day, starting time slot, ending time slot, and assigned classroom. Course Availability Matrix is also updated the same way it was during step 4 except that the time slots concerned are the ones assigned to the course that has just been added to the Scheduled Courses Dictionary.

At the end of the process of the algorithm, the Scheduled Courses Dictionary groups all the courses that have been scheduled along with their day, starting time slot, ending time slot, and classroom, and the Unscheduled Courses Vector lists the courses that had no feasible consecutive time slots.

9. Results and Discussion

9.1. Tests on the first model

9.1.1. First test – 1 student group

I coded the mathematical model in Julia with the solver Gurobi to solve it and find a feasible solution. Since in the code it is necessary to put an objective function, I integrated this one :

$$Min \sum_{c=1}^{C} \sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c,i,t,d,r,s)$$

In addition, while coding the model, I set a time limit of 40 seconds.

Initially, I conducted tests without considering the two last constraints as I thought that the other constraints were enough to ensure that only one classroom would be attributed to one course for its whole duration :

- $\sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{i=1}^{I} \sum_{s=1}^{S} X(c,i,t,d,r,s) = duration(c) \times Z(c,r) \quad \forall c,r$ ensures that the same classroom is assigned to the course for its entire duration, preventing unnecessary classroom changes.

- $\sum_{r=1}^{R} Z(c,r) = 1 \quad \forall c$ ensures that only one room can be attributed to a course In other words, each course is assigned to only one room.

However, I noticed that some courses did not have the same classroom for their whole duration. The classrooms were switched from one time slot to another for the same course. This is why I introduced the next general constraint :

Only one classroom can be attributed to a course. Otherwise, a course could have different classrooms from one time slot to another, which would mean that students would need to move from one class to another for the same course.

For this first test, I considered an instance "*Information_needed – 1.A*" composed of 21 classrooms all located in the Building of Rue Louvrex so that students would not have to move from one building to another one. The aim of putting so many classrooms even though only one bloc of students is considered for this instance is to analyse how the model reacts when adding additional blocs of students and more specifically if there are classroom changes. In addition, the classrooms attributed to courses could change from one course to another as the number of students enrolled in the course is different for each course.

I considered only the first bloc of business engineering and the 14 courses they are enrolled in during the first semester with the 9 instructors responsible for those courses.

Some complexities were simplified for this first model :

- The "Anglais 1" course has been kept into one unique course when in reality, this course is divided into 5 different courses given to five different student sub-groups, taught by five different instructors.

- The “Maitrise de l’outil informatique course” has also been kept into one unique course while in reality, it is given three times a week to three different student sub-groups by the same instructor.

- The “Economie politique – Microéconomie – TP1” course is in reality given twice a week to two different student sub-groups.

- The “Finance et comptabilité – TP”, “Physique Générale : partim 1 – TP 1” and “Physique Générale : partim 1 – TP 2 ” courses are in reality given three times a week to three different student sub-groups.

- The “Cours de langue” course reflects the “Espagnol 1” in reality as the instructor is Alexis Alvarez, the teacher responsible for this course in the real-life case. However, in reality, students have the opportunity to choose between 8 different language courses: “Allemand 1”, “Allemand 1+”, “Chinois 1 (anglais)”, “Espagnol 1”, “Espagnol 1+”, “Italien 1”, “Néerlandais débutant”, and “Néerlandais 1”. In this instance, it was easier to consider only one language course for this model. Moreover, some of the language courses are even given multiple times a week to different student sub-groups, but here, I only consider the language course as a whole and given to one unique student group. I also decided that this student group is composed of 248 students as it was the number of students enrolled in this course last year. So, this course language is considered to be given to one unique student group and only once a week.

Those courses will be considered as they should be in the last version of the ILP model, which means that they would be split when they need to be split between different student groups.

I ran the code and got the values of the different variables that I interpreted by designing a corresponding schedule (*fig.1*) in Excel. I did that for every result I had during the tests, here is the schedule obtained.

	Monday	Tuesday	Wednesday	Thursday	Friday
8h00-8h30	Analyse sociale de l'économie et de l'entreprise - Annie Cornet - N1a 30 (0/30)	Maitrise de l'outil informatique - Eddy Heins - N1a 30 (0/30)	Mathématiques: Analyse infinitésimale - 1 - Valérie Henry - N1a 50 (0/50)	-	-
8h30-9h00				-	-
9h00-09h30				-	-
09h30-10h00				-	-
10h00-10h30		Physique générale : partim 1 - TP 2 - Amandine Collignon - N1a 50 (0/50)	Finance et comptabilité - TP - Wilfried Niessen - N1a 50 (0/50)	-	-
10h30-11h00				-	-
11h00-11h30	Anglais 1 - Céline Leroy - N1a 30 (0/30)	Physique générale : partim 1 - TP 1 - Amandine Collignon - N1a 50 (0/50)		-	-
11h30-12h00				-	-
12h00-12h30		Physique générale : partim 1 - théorie 2 - Pierre Deneye - N1a 50 (0/50)	Finance et comptabilité - Wilfried Niessen - N1a 50 (0/50)	-	-
12h30-13h00				-	-
13h00-13h30				-	-
13h30-14h00				-	-
14h00-14h30	Economie politique - Microéconomie - Alain Jousten - N1a 30 (0/30)	Physique générale : partim 1 - théorie 1 - Pierre Deneye - N1a 50 (0/50)		-	-
14h30-15h00				-	-
15h00-15h30			-	-	-
15h30-16h00			-	-	-
16h00-16h30		Mathématiques: Analyse infinitésimale - 2 - Valérie Henry - N1a 50 (0/50)	-	-	-
16h30-17h00			-	-	-
17h00-17h30	Economie politique - Microéconomie - TP 1 - Alain Jousten - N1a 30 (0/30)		-	-	-
17h30-18h00			-	-	-
18h00-18h30	Cours de langue - Alexis Alvarez - N1a 50 (0/50)		-	-	-
18h30-19h00		-	-	-	-
19h00-19h30	-	-	-	-	-
19h30-20h00	-	-	-	-	-

Figure 1. Schedule – 1.A. – Bloc 1 of Business Engineer – *Results V1*

It took 2.15 seconds to have for the ILP to be optimized, and the objective value is 56.

While analysing the results, I noticed that the order of the course followed the order of the list for the first four courses which are “Analyse sociale de l’économie et de l’entreprise”, “Anglais 1”, “Economie politique – Microéconomie” and “Economie politique – Microéconomie – TP 1”. The “Physique Générale” courses are also consecutive and given within the same day which, in reality, is never the case because the class scheduler tries to diversify the courses given on a day.

No lunch break is planned which is also a problem because, in reality, students need to have a break to eat lunch as it is an essential need.

Every course is given at the beginning of the week, no courses are planned on Thursday or Friday which makes really busy days, especially on Monday and Tuesday.

It is always the two same classrooms that are used :

- N1a 30 (0/30) which has a capacity of 228 students.
- N1a 50 (0/50) which has a capacity of 513 students.

This could be explained by the fact that those are the classrooms having the highest capacity at HEC and that the course having the lowest number of students enrolled in is the "Analyse sociale de l'économie et de l'entreprise" with 171 students enrolled and no other classrooms than the two ones cited above can be used for that many students.

I observed that some courses that should be given within different days in the real-life case are given within the same day in this schedule. This is the case for the "Physique Générale : partim 1 – TP1" and "Physique Générale : partim 1 – TP 2" courses that are given on Tuesday and for the "Physique Générale : partim 1 – théorie 1" and "Physique générale : partim 1 – théorie 2" courses that are also given on Tuesday.

Some courses are planned quite early and other quite late. For example, the courses "Analyse sociale de l'économie et de l'entreprise", "Maitrise de l'outil informatique", "Mathématique : Analyse infinitésimale – 1 » start at 8 Am while "Cours de langue" and "Mathématique : Analyse infinitésimale - 2" finish after 6pm. In reality, the class scheduler tries to avoid putting courses before 9 am and after 6 pm.

9.1.1.1. Improvement for the first test

To get closer to reality, I decided to introduce a new constraint to the model and in the code so that no courses could be given before 9 am, between 12 am and 1 pm, and after 6 pm. I wanted to see if it was possible to have a solution with this supplementary constraint :

- $X(c, i, t, d, r, s) = 0 \quad \forall t \in \{1,2,9,10,21,22,23,24\}, \forall c, i, d, r, s$ ensures that no courses can be given before 9 am, between 12 am and 1 pm, and after 6 pm.

As a reminder, the time slots 1 and 2 correspond to the time slots 8h-8h30 and 8h30-9h00, the time slots 9 and 10 correspond to the time slots 12h-12h30 and 12h30-13h00, and the time slots from 21 to 24 correspond to the time slots from 18h until 20h.

While interviewing Ms. Boxus, she told me that she tried to avoid scheduling courses during those time slots to reduce students' time at school. However, it is important to keep them so that, if I notice that some courses cannot be scheduled, I can remove this constraint. Moreover, when I will split all the courses depending on sub-groups, for example, the "Cours de langue" course will be split between the different language courses, it is interesting to keep the time slots before 9 am and after 6 pm so that if no other time slots are free, those can be considered.

I tested this new instance “*Information – 1.A’.*” and got the following results (*fig. 2*) :

	Monday	Tuesday	Wednesday	Thursday	Friday
8h00-8h30	-	-	-	-	-
8h30-9h00	-	-	-	-	-
9h00-09h30	Analyse sociale de l'économie et de l'entreprise - Annie Cornet - N1a 30 (0/30)	nomie politique - Microéconomie - Alain Jousten - N1a 30 (0/	ique générale : partim 1 - théorie 2 - Pierre Deneye - N1a 50 ((	ématiques: Analyse infinitésimale - 1 - Valérie Henry - N1a 50	-
09h30-10h00					-
10h00-10h30					-
10h30-11h00					-
11h00-11h30					-
11h30-12h00					-
12h00-12h30	-	-	-	-	-
12h30-13h00	-	-	-	-	-
13h00-13h30	Anglais 1 - Céline Leroy - N1a 30 (0/30)	Maitrise de l'outil informatique - Eddy Heins - N1a 30 (0/30)	ique générale : partim 1 - théorie 1 - Pierre Deneye - N1a 50 ((	Finance et comptabilité - TP - Wilfred Niessen - N1a 50 (0/50)	-
13h30-14h00					-
14h00-14h30					-
14h30-15h00					-
15h00-15h30		que générale : partim 1 - TP 2 - Amandine Collignon - N1a 50 (	ématiques: Analyse infinitésimale - 2 - Valérie Henry - N1a 50	Finance et comptabilité - Wilfred Niessen - N1a 50 (0/50)	-
15h30-16h00					-
16h00-16h30	Economie politique - Microéconomie - TP 1 - Alain Jousten - N1a 30 (0/30)	que générale : partim 1 - TP 1 - Amandine Collignon - N1a 50 (			-
16h30-17h00					-
17h00-17h30	Cours de langue - Alexis Alvarez - N1a 50 (0/50)	-	-	-	-
17h30-18h00		-	-	-	-
18h00-18h30	-	-	-	-	-
18h30-19h00	-	-	-	-	-
19h00-19h30	-	-	-	-	-
19h30-20h00	-	-	-	-	-

<u>Figure 2. Schedule – 1.A’. – Bloc 1 of Business Engineer – *Results V1*</u>

It took 1.95 seconds to have for the ILP to be optimized, and the objective value is still 56.

The new constraint is respected as no courses are starting before 9 am or finishing after 6 pm nor given between 12 am and 1 pm.

As a consequence of this new condition, there are now courses given on Thursday and we can observe fewer busy days at the beginning of the week compared to what we had in the previous tests as there are fewer courses given per day for Monday and Tuesday.

The classrooms assigned to the courses are the same for each course as we had previously. The order of the courses given throughout the week remains almost the same as for the previous test except for the “Economie Politique – Microéconomie” course that is given later in the week than previously.

However, the two “Physique Générale” practical courses, the two “Physique Générale” theoretical courses, and the two “Finance et comptabilité” courses are still respectively given within the same day. To improve this, in the second model, I will introduce a new parameter and a constraint that prevent the concerned courses from being given within the same day.

9.1.2. Second test – 2 student groups

For this new instance, I added the second bloc of business engineering and the 11 courses they are enrolled in the first semester with the 7 new instructors responsible for those courses in addition to the courses assigned to the first bloc of business engineering.

Again, the “Cours de langue 2” course reflects the “Espagnol 2” in reality as the instructor is Véronique Peiffer, the teacher responsible for this course in the real-life case. However, in reality, students have the opportunity to choose between 7 different language courses: “Allemand 2”, “Allemand avancé 2”, “Chinois 2 (anglais)”, “Espagnol 2”, “Espagnol avancé 2”, “Italien 2”, and “Néerlandais 2”. In this instance, it was easier to consider only one language course for this model.

The “Anglais 2” course has been simplified. In reality, this course is given by three different instructors, four times a week to four different student sub-groups.

I kept the latest constraint added that prevents courses from being given before 9 am, after 6 pm, and between 12 am and 1 pm.

I tested this new instance “*Information – 1.B.*” and got the following results:

For the first bloc of business engineering (*fig.3*):

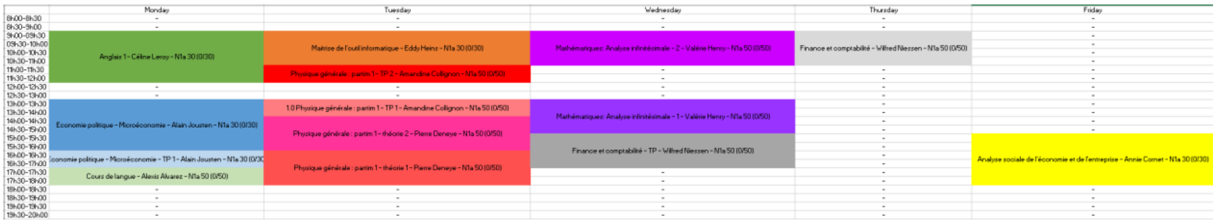


Figure 3. Schedule – 1.B. – Bloc 1 of Business Engineering – *Results V1*

For the second bloc of business engineering (*fig.4*):

Figure 4. Schedule – 1.B. – Bloc 2 of Business Engineering – *Results V1*

It took 14.75 seconds in total for the ILP to be optimized, and the objective value is 102.

By comparing the results with what we had during the previous test, I noticed that the classrooms assigned per course have not changed for the first bloc of business engineering. However, for the first bloc of business engineering, there are now courses each day of the week.

Only the “Economie politique – Microéconomie – TP1” and “Cours de langue” courses are given at the same time slots and days as for the previous results.

For the second bloc of business engineering, there are no courses given on Friday.

If we analyse the classrooms assigned to the courses for the second bloc of business engineering, we can say that other classrooms than the one assigned to the courses dedicated to the first bloc of business engineer are considered except for the “Cours de langue 2” that has been assigned classroom N1a 30 (0/30).

The new classrooms considered are :

- N1d 0/86 that has a capacity of 108 students.
- N1d 1/82 that has a capacity of 80 students.
- N1a 138 (1/38) that has a capacity of 120 students.

The lowest number of students enrolled in a course is 78, for the “Anglais 2” course, which explains why this course is the only one with the “N1d 1/82” classroom that has a capacity of 80 students, and the highest number of students enrolled to a course is 167 to the “Cours de langue 2” course.

By observing these results and the previous ones, I made the observation that the diversity of classrooms used increases with the number of courses. This can be explained by the fact that the same classrooms cannot be available for many courses at the same time so other ones need to be used.

For the second bloc of business engineering, as well as for the first bloc, we observe that some courses that should be given within different days in the real-life case are given within the same day.

This is the case for the “Comptabilité analytique et contrôle de gestion – 1” and “Comptabilité analytique et contrôle de gestion – 2” courses that are given on Tuesday and for the “Mathématiques pour ingénieurs de gestion – théorie 1” and “Mathématiques pour ingénieurs de gestion – théorie 2” courses that are given on Wednesday.

9.1.3. Third test – 3 student groups

For this new instance “*Information_needed – 1.C.*”, I added the third bloc of business engineer and the 11 courses they are enrolled in the first semester with the 6 new instructors responsible for those courses in addition to the courses assigned to the first and second blocs of business engineer and to the instructors already considered for the previous blocs. I also had to increase the time limit of the optimizer from 40 seconds to 120 seconds.

Again, the “Cours de langue 3” reflects the “Espagnol 3” in reality as the instructor is Véronique Peiffer, the teacher responsible for this course in the real-life case. However, in reality, students have the opportunity to choose between 7 different language courses: “Allemand 3”, “Allemand avancé 3”, “Chinois 3 (anglais)”, “Espagnol 3”, “Espagnol avancé 3 (espagnol)”, “Italien 3 (italien)”, and “Néerlandais 3”. In this instance, it was easier to consider only one language course for this model.

The “Anglais 3” course has been simplified. In reality, this course is taught by two different instructors and is given four times a week, to four different student sub-groups.

The “Ateliers de compétences” course is one of the most complex courses to treat. Students can choose between more than fifty courses that are considered as “Ateliers de compétences”. Those courses aimed at developing soft skills such as negotiation, oral presentation, and audacity. Generally, one course of this type is given 4 times during the semester, for four hours to a sub-group of 20 students. However, 454 students registered for this course. Then, in my instance, I consider a group of 454 students and the course given during the whole semester. This way, every student would have their “Atelier de compétences” course at the same time. Only the choice of classroom would have to be reviewed afterward to make it match the real number of students for each sub-course.

For the “Gestion stratégique des ressources humaines”, students have, in reality, the choice between the English version and the French version. Depending on the choice, the instructor that gives the course is not the same. Then, whenever the French version is given, the English version could be given too but in another classroom.

Concerning the “Introduction à l’électronique” courses, those are optional courses. In fact, students from the third bloc of business engineer must choose between “Introduction à l’électronique”, “Marketing and Innovation (anglais)”, “Programmation orientée gestion” and “Introuction à la programmation – [10h Laboratoire]. I have chosen to consider the “Introduction à l’électronique” course as it is one of them that is given during the first semester and in my instance, I only consider courses taught during the first semester.

As for the previous tests, I kept the latest constraint added that prevents courses from being given before 9 am, after 6 pm, and between 12 am and 1 pm.

I ran the code and got the following results:

For the first bloc of business engineering (*fig. 5*):

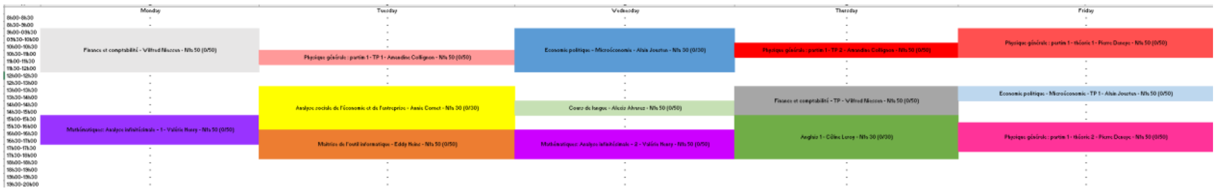


Figure 5. Schedule – 1.C. – Bloc 1 of Business Engineering – *Results V1*

For the second bloc of business engineering (*fig. 6*):

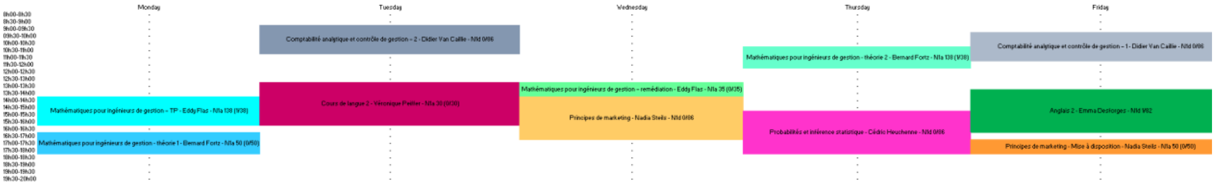


Figure 6. Schedule – 1.C. – Bloc 2 of Business Engineering – *Results V1*

For the third bloc of business engineer (*fig. 7*) :

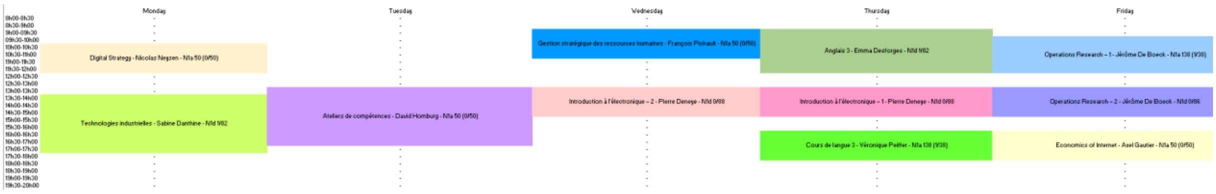


Figure 7. Schedule – 1.C. – Bloc 3 of Business Engineer – *Results V1*

It took 56.46 seconds in total for the ILP to be optimized, and the objective value is 157.

By comparing this test with the two previous ones, I can state that, for this mathematical model, the bigger the instance is, the higher the time required for the ILP to be optimized is. Moreover, the higher is the number of courses, the higher is the objective.

For the first bloc of business engineer, I noticed some classroom changes per course compared to the previous test, this is the case for the "Economie politique – Microéconomie – TP 1" and "Maitrise de l'outil informatique" that switched from the "N1a 30 (0/30)" to the "N1a 50 (0/50)" that has a higher capacity.

Still, for the first bloc of business engineering, the courses that must be taught on different days are, in fact, given on different days. This is new compared to the previous tests as before, "Physique Générale" practical courses and theoretical courses were respectively given within the same day, this is not the case anymore.

For the second bloc of business engineering, there are also classroom changes per course compared to the previous results. "Principe marketing – Mise à disposition" course moved from "N1d 0/86" to "N1a 050 (0/50)" classroom which has a higher capacity. "Mathématiques pour ingénieurs de gestion – remédiation" course moved from "N1a 138 (1/38) " to "N1a 35 (0/35)" classroom which as exactly the same capacity as the previous classroom. "Mathématiques pour ingénieurs de gestion – théorie 1" course moved from "N1a 138 (1/38) " to "N1a 50 (0/50) " classroom which has capacity.

Moreover, now, some courses dedicated to the second bloc of business engineering are also given on Friday.

Concerning the results for the third bloc of business engineering, there are courses given every day of the week. The “Operations Research” courses should not be given on the same day, in reality; however, this is the case here.

Generally speaking, no courses are scheduled the same way they were for the previous test. All courses have been rescheduled.

9.1.4. Limitations of this first model

The main limitation of this first model is the fact that some courses that should be given on different days are given on the same days.

In addition, some courses, such as “Anglais 1”, “Cours de langue 2” or “Ateliers de compétences” have been simplified to facilitate the management of those courses. Student sub-groups have not been taken into account, because students were considered as groups and not as individuals in this model.

Moreover, the HEC scheduler generally pays attention to the fact that instructors should not work for more than four days per week. Even though I did not encounter this case in those instances as no instructor has more than four courses assigned, this should be taken into consideration too.

Finally, some courses should start at the same time. For example, when the HEC scheduler designs the class schedule, she tries to make matches between practical courses given to different student groups. In other words, we know that “Finance et comptabilité – TP”, “Physique Générale : partim 1 – TP 1” and “Physique Générale : partim 1 – TP 2 ” courses are in reality given three times a week to three different student sub-groups, then, an arrangement should be made so that when a student sub-group has one of those courses, the two other sub-groups have the two other courses so they would have similar schedules.

9.2. Tests on the second model

9.2.1. First test – 1 student group

I coded the mathematical model in Julia with the solver Gurobi to solve it and find a feasible solution. Since it is necessary in the code to put an objective function, I integrated this one :

$$Min \sum_{c=1}^{C} \sum_{i=1}^{I} \sum_{t=1}^{T} \sum_{d=1}^{D} \sum_{r=1}^{R} \sum_{s=1}^{S} X(c,i,t,d,r,s)$$

In addition, while coding the model, I set a time limit of 120 seconds, so 2 minutes.

For this first test, I considered an instance “*Information_needed – 2.A.”* composed of 21 classrooms all located in the Building of Rue Louvrex, so that students would not have to move from one building to another.

I considered only the first bloc of business engineering and the 14 courses they are enrolled in during the first semester, along with the 9 instructors responsible for those courses. The same complexities that were simplified for the first instance used for the first test of the first model are present in this current instance. In other words, the basis of the instance is the same as the one used for the first test I made for the first model.

In addition, I added the “Type” table in the Excel file that gathers all the data. This file groups the courses that should not be given on the same day. This is the case for the course “Mathématiques: Analyse infinitésimale – 1” and “Mathématiques: Analyse infinitésimale – 2”, “Physique générale :

partim 1 – théorie 1” and “Physique générale : partim 1 – théorie 2”, and finally, “Physique générale : partim 1 – TP 1” and “Physique générale : partim 1 – TP 2” .

I also added the availability table to the Excel file that gathers all the data and started with an instance where instructors are always available. I wanted to begin with instructors available all the time so I could have an initial result showing how the model reacted to the new constraints except the availability constraint. This result will also serve as a basis to remove availabilities for each teacher.

I ran the code and got these results *(fig.8)* :

<u>Figure 8. Schedule – 2.A. – Bloc 1 of Business Engineering – *Results V2*</u>

It took 1.85 seconds to have for the ILP to be optimized, and the objective value is 56, which is the same value as what I obtained with the first instance of the first model.

Compared to the results of the first test of the first model, after implementing the constraint that prevents courses from being given before 9 am, after 6 pm, and between 12 am and 1 pm, there are courses given every day of the week. The courses “Physique Générale : partim 1 – TP 1” and “Physique Générale : partim 1 – TP 2”, as well as the courses “Physique Générale : partim 1 – théorie 1” and “Physique Générale : partim 1 – théorie 2” are no longer taught on the same day anymore. Moreover, the only course that has a classroom change is “Maitrise de l’outil informatique”, which moves from classroom “N1a 30 (0/30)” to classroom “N1a 50 (0/50)” However, it is still the two classrooms that are used:

- N1a 30 (0/30), which has a capacity of 228 students.
- N1a 50 (0/50), which has a capacity of 513 students.

9.2.1.1. Less availability

To test the model with a more restrictive instance “*Information_needed – 2.A’.*”, I removed the time slots that were associated with a course for each teacher. In other words, whenever a teacher has a course to teach in the results generated by the previous test, I removed the time slots during which they have to give that course. This way, they are no longer available during those time slots, and I can create a new instance to test my model when there are fewer availabilities for the teachers.

For example, the teacher Annie Cornet has a course called “Analyse sociale de l’économie et de l’entreprise » on Monday from 9am to 12 am. I changed those time slots for that day as unavailable in the new instance, for this instructor.

I ran the code with the new instance and got these results *(fig.9)*:

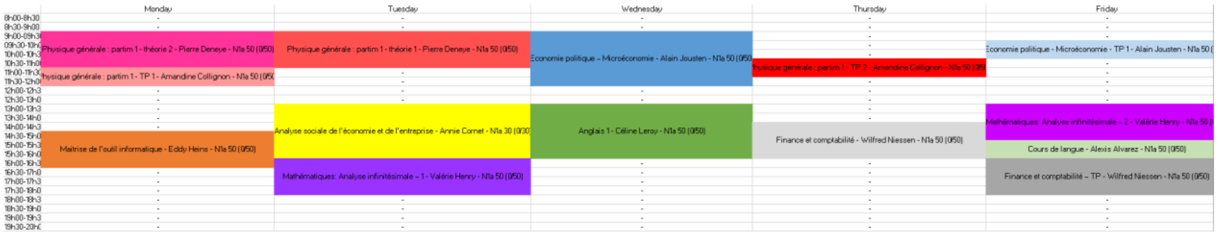


Figure 9. Schedule – 2.A'. – Bloc 1 of Business Engineering – *Results V2*

It took 2.17 seconds for the ILP to be optimized, and the objective value is 56, which is the same value as what I obtained with the previous instance.

Except for the course "Finance et comptabilité – TP", all courses are given another day than the day they were taught during the previous test.

The courses "Economie politique – Macroéconomie", "Anglais 1" and "Economie politique – TP 1" changed classrooms from « N1a 30 (0/30) » to "N1a 50 (0/50)".

I continued removing the time slots during which instructors have to give that course. This way they are no longer available during those time slots, and I created new instances to test my model. In total, I had six new instances with time slots of no availabilities for the instructors, from the one with the most availabilities to the one with the least availabilities for instructors. I was not able to get a result for a seventh instance as there were so few availabilities for each teacher that the model was not able to find any solution.

The results I got are available in the Appendices section.

We can conclude those tests by saying that the fewer availabilities there are for instructors, the less the model is likely to find a feasible solution.

While comparing the different results I had from the different instances, I noticed that the courses with a number of students enrolled to the course lower than the capacity of the classroom "N1a 30 (0/30)", which is 228 students, often have a switch of the classroom. In fact, for the courses "Analyse sociale de l'économie et de l'entreprise", "Anglais 1", "Economie politique - Microéconomie", "Economie politique – Microéconomie – TP 1", and "Maitrise de l'outil informatique", that have a number of students enrolled lower than the capacity of the classroom "N1a 30 (0/30)" sometimes have this classroom associated and sometimes the classroom "N1a 50 (0/50)", that has a capacity of 513 students. This can be explained by the fact that, as I am only considering the 1st bloc of business engineering in this instance, the classrooms are not used by other blocs of students.

9.2.2. Second test – 2 student groups

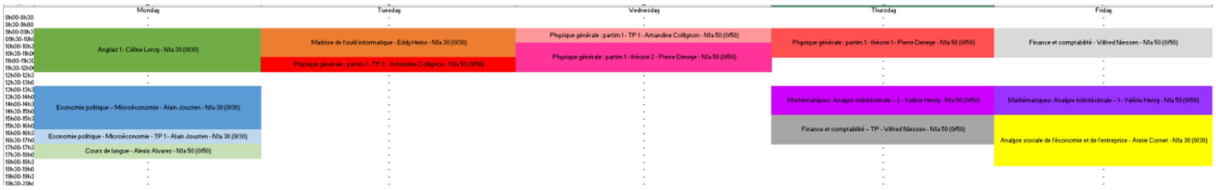
For this new instance "*Information_needed – 2.B.*", I added the second bloc of business engineering and the 11 courses they are enrolled in the first semester with the 7 new instructors responsible for those courses in addition to the courses assigned to the first bloc of business engineer, as I did for the second test of the first ILP model.

In the "Type" sheet of the Excel file gathering the data, I added the courses "Comptabilité analytique et contrôle de gestion – 1" and "Comptabilité analytique et contrôle de gestion – 2", and the courses "Mathématiques pour ingénieurs de gestion - théorie 1" and "Mathématiques pour ingénieurs de gestion - théorie 2" as they should not be given within the same day.

For the first instance of those second tests, I set the teachers as available all the time.

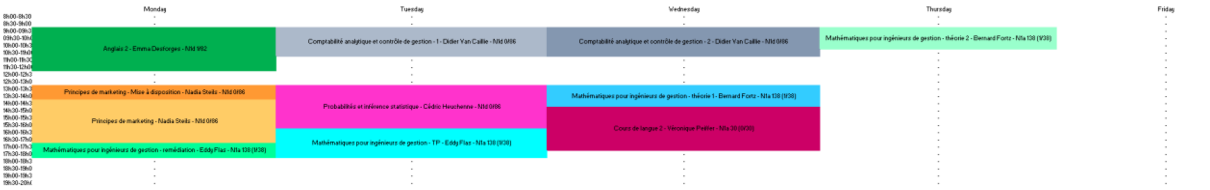
I ran the code and got the following results:

<u>For the first bloc of business engineering *(fig.10):*</u>



<u>Figure 10. Schedule – 2.B. – Bloc 1 of Business Engineering – *Results V2*</u>

<u>For the second bloc of business engineering *(fig.11)*:</u>



<u>Figure 11. Schedule – 2.B. – Bloc 2 of Business Engineering – *Results V2*</u>

It took 17.88 seconds in total for the ILP to be optimized, and the objective value is 102, which is the same value as what I obtained with the second instance of the first model.

By comparing the results for the first bloc of business engineering with the results I got with the first instance of the first tests of the ILP model, I can say that 42.86% of the courses kept the same schedule. In addition, except the course “Maitrise de l’outil informatique”, all courses from the first bloc of business engineering kept the same classroom assigned as the one they had in the results of the first instance of the first tests.

By focusing on the second bloc of business engineering, I noticed that 54.54% of courses kept the same schedule as what they had for the second test made on the first model. And as for this test, there are no courses given on Friday.

Exactly as what I analysed for the second test made on the first model, the same new classrooms are considered. 100% of the courses of the second bloc of business engineering have the same classroom assigned as the one they had assigned during that test.

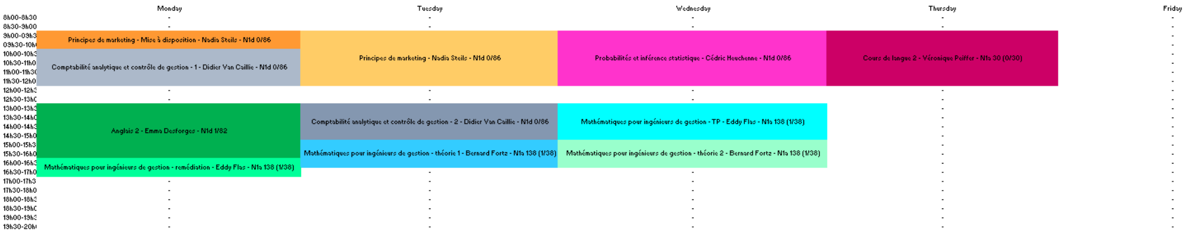
9.2.2.1. Less availability

As I did for the first tests of this ILP model, I removed the time slots that were associated with a course for each teacher to create a new instance “*Information_needed – 2.B’.”* and here is the result I got :

<u>For the first bloc of business engineering *(fig.12)*:</u>

<u>Figure 12. Schedule – 2.B’. – Bloc 1 of Business Engineering – *Results V2*</u>

<u>For the second bloc of business engineering *(fig.13)*:</u>



<u>Figure 13. Schedule – 2.B'. – Bloc 2 of Business Engineering – *Results V2*</u>

It took 19.75 seconds in total for the ILP to be optimized, and the objective value is 102, which is the same value as what I obtained with the previous instance.

By comparing the results, I got with the results I had for the first instance of the second tests, I observed that 96% of courses kept the same classroom assigned even though they do not have the same time slots attributed anymore.

I continued removing the time slots during which instructors have to give that course to create new instances to test my model. In total, I had four new instances with time slots of no full availability for the instructors, from the one with the most availability to the one with the least availability for instructors. I did not test a fifth instance as I already knew from the previous tests that the fewer time slots available there are, the less the model is likely to find a feasible solution.

The results I got are available in the Appendices section.

9.2.3. Third test – 3 student groups

For this new instance "*Information_needed – 2.C.*", I added the third bloc of business engineering and the 11 courses they are enrolled in the first semester with the 6 new instructors responsible for those courses in addition to the courses assigned to the first and second blocs of business engineer and to the instructors already considered for the previous blocs as I did for the third test of the first model.

The new courses in the "Type" sheet of the Excel file that gathers the data are the courses "Operations Research – 1" and "Operations Research – 2" as they should not be given within the same day.

I set the instructors as fully available for all the time slots.

When I first ran the code, I had no results. I wondered if it was due to the lack of classrooms, so I decided to add 5 new classrooms :

- O2 Bovy (3/7b) that has a capacity of 140 students.
- O2 Lejeune (1/3a) that has a capacity of 267 students.
- O2 Noppius (2/3a) that has a capacity of 500 students.
- O2 Pousseur (3/7a) that has a capacity of 98 students.
- O2 Thiry (2/8a) that has a capacity of 240 students.

They are all located in the building "Liège_Centre_Ville - Site Opéra".

However, I still had no feasible solution, so I added another classroom called "A1 Salle Gothot (1/36)" which is located in the building "Liège Centre Ville - Site 20-Août". At the same time, I also increased the time limit of execution of the model from 40 seconds to 120 seconds and finally got the following results :

For the first bloc of business engineering (*fig.14*):

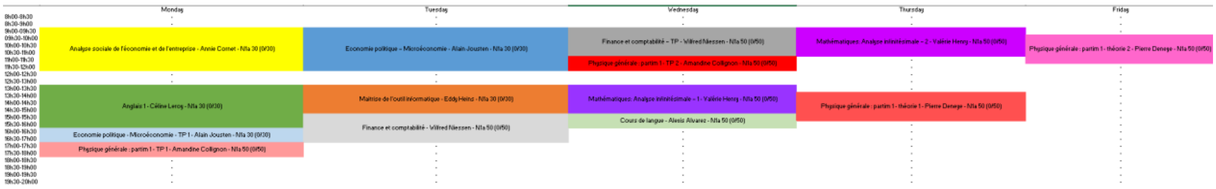


Figure 14. Schedule – 2.C. – Bloc 1 of Business Engineering – *Results V2*

For the second bloc of business engineering *(fig.15)* :

Figure 15. Schedule – 2.C. – Bloc 2 of Business Engineering – *Results V2*

For the third bloc of business engineering *(fig.16):*

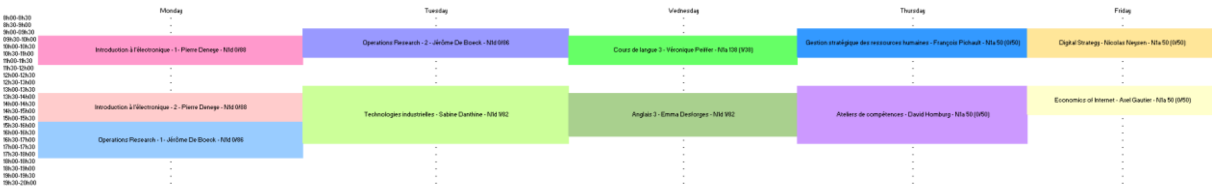


Figure 16. Schedule – 2.C. – Bloc 3 of Business Engineering – *Results V2*

It took 71.83 seconds in total for the ILP to be optimized, and the objective value is 157, which is the same value as what I obtained with the third instance of the first model.

I can state the same observation as I did for the tests made on the first ILP model which is that the bigger the instance is, the higher is the time required for the ILP to be optimized and the higher is the number of courses, the higher is the objective value.

Moreover, I noticed that the higher is the number of constraints, the higher is the time required for the ILP to be optimized.

Considering only the results for the first and the second blocs of business engineer, I noticed that only 12% of courses have the same time slots assigned as the first instance of the second tests.

However, 100% of them kept the same classroom assigned even though they do not have the same time slots attributed.

Focusing on the results for the third bloc of business engineering now, I noticed that a new classroom that has not been considered for the other blocs has been attributed to the courses of the third bloc. This new classroom is “N1d 0/88” and has a capacity of 36 students. However, I noticed that none of the new classrooms I introduced before running this instance with the code has been used. This means that the initial problem of no feasible solutions found was only linked to the time limit that was too low and not the lack of classrooms. I also noticed that the bigger the instance is, the longer the time required for the model to find a solution is.

I also noticed that the courses “Introduction à l’électronique – 1” and “Introduction à l’électronique – 2” should have been put in the sheet “Type” as courses that cannot be given within the same day.

9.2.3.1. Less availability

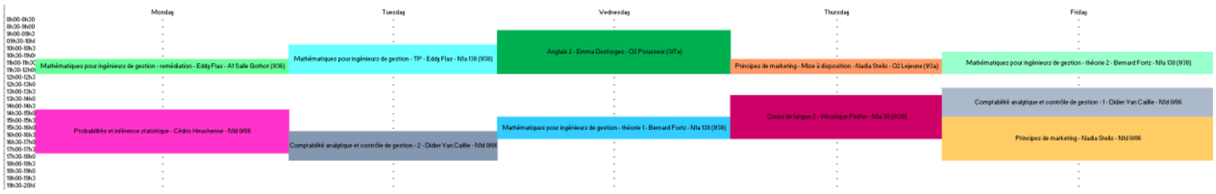
To build the new instance “*Information_needed – 2.C’.*” to test the model, I removed time slots that were associated with a course for each teacher in the previous result and I added the missing courses to the “Type” sheet so that the model would not put them on the same day in the schedule.

I ran the code and got the following results:

<u>For the first bloc of business engineering (*fig.17*):</u>

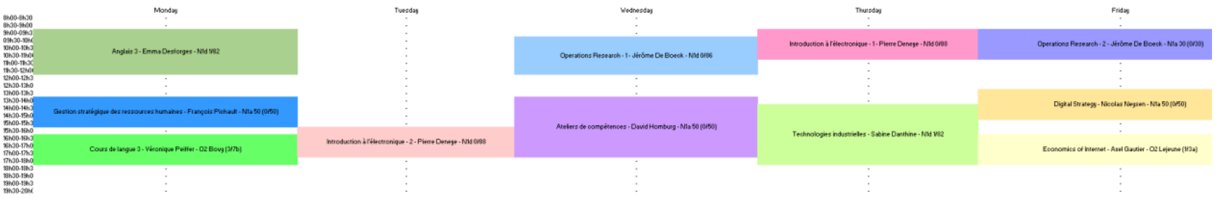
<u>Figure 17. Schedule – 2.C’. – Bloc 1 of Business Engineering – *Results V2*</u>

<u>For the second bloc of business engineering (*fig.18*):</u>



<u>Figure 18. Schedule – 2.C’. – Bloc 2 of Business Engineering – *Results V2*</u>

<u>For the third bloc of business engineering (*fig.19*):</u>



<u>Figure 19. Schedule – 2.C’. – Bloc 3 of Business Engineering – *Results V2*</u>

It took 85.89 seconds in total for the ILP to be optimized, and the objective value is 157, which is the same value as what I obtained with the previous instance.

Even though the objective value is the same as when the instructors were fully available, each time I removed availabilities from an instance during the tests made on the second ILP model, the time required for the model to be optimized increased.

Compared to the results of the previous instance for those third tests, I noticed that 69.44% of courses kept the same classrooms even though they did not have the same time slots assigned. However, new classrooms that were considered in the previous results have been attributed to some courses. The new classrooms considered are the ones I added at the beginning of the third test :

- A1 Salle Gothot (1/36) that has a capacity of 360 students.
- O2 Thiry (2/8a) that has a capacity of 240 students.

- O2 Noppius (2/3a) that has a capacity of 500 students.
- O2 Lejeune (1/3a) that has a capacity of 267 students.
- O2 Pousseur (3/7a) that has a capacity of 98 students.

9.2.3.2. Fewer classrooms

I wondered if the use of the new classrooms was linked with the decrease of availabilities of the instructors, so I created a new instance “*Information_needed – 2.C’’.*” that had the same data as the previous instance except for the number of classrooms. I removed the classrooms I added before starting the third tests, which are “O2 Bovy (3/7b)”, “O2 Lejeune (1/3a)”, “O2 Noppius (2/3a)”, “O2 Pousseur (3/7a)”, “O2 Thiry (2/8a)”, “A1 Salle Gothot (1/36)”.

I ran the code and got the following results :

For the first bloc of business engineering (*fig. 20*):

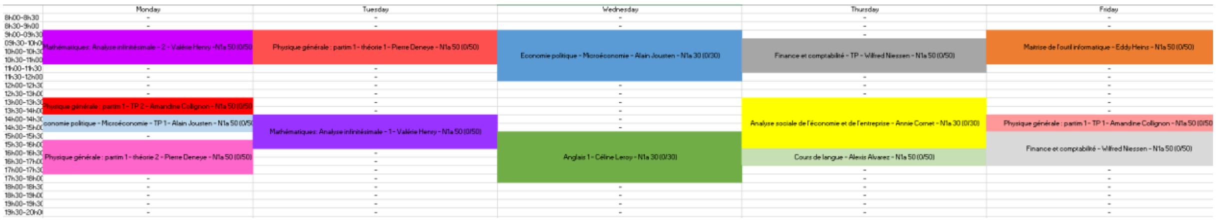


Figure 20. Schedule – 2.C’’. – Bloc 1 of Business Engineering – *Results V2*

For the second bloc of business engineering (*fig. 21*):

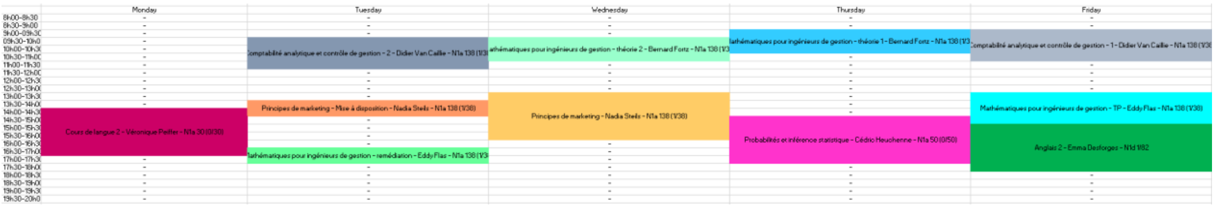


Figure 21. Schedule – 2.C’’. – Bloc 2 of Business Engineering – *Results V2*

For the third bloc of business engineering (*fig .22*):

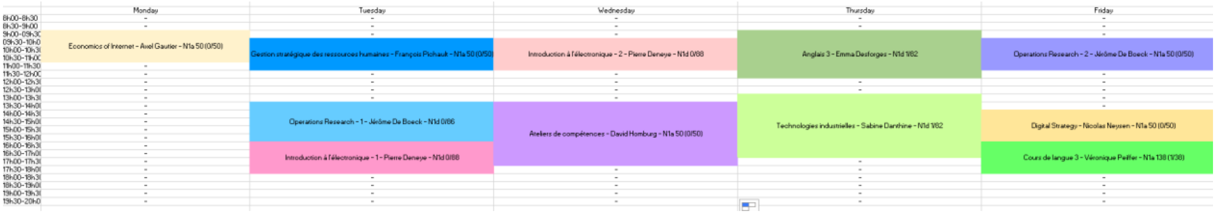


Figure 22. Schedule – 2.C’’. – Bloc 3 of Business Engineering – *Results V2*

It took 59.30 seconds in total for the ILP to be optimized, and the objective value is 157, which is the same value as what I obtained with the previous instance.

By removing some classrooms, the time required for the ILP to be optimized decreased compared to the two previous tests containing the three blocs of business engineering.

The new classrooms were not necessary to find a feasible solution.

9.2.4. Limitations of this second model

The main limitation of this second model is the fact that students are considered as groups and not as individuals. If they were considered as individuals, it would be easier to create sub-groups when needed. For example, as mentioned earlier, the "Finance et comptabilité – TP", "Physique Générale : partim 1 – TP 1" and "Physique Générale : partim 1 – TP 2 " courses are given three times a week to three different student sub-groups. Therefore, an arrangement should be made so that when a student sub-group has one of those courses, the two other sub-groups have the two other courses so they would have similar schedules.

In addition, the integration of the Décret Paysage into the model could improve it to be closer to reality. The Décret Paysage allows students to have courses from different blocs of students. In this way, they can pursue their studies without being blocked in a specific bloc.

9.3. Tests on the third model

I coded the third model in Julia with the solver Gurobi to solve it and find an optimal solution.

While coding the model, I set a time limit of 1800 seconds, so 30 minutes.

9.3.1. First test – 1 student group

For the first test, I considered an instance "*Information_needed – 3.A.*" composed of 21 classrooms all located in the Building of Rue Louvrex so that students would not have to move from one building to another one.

I considered only the first bloc of business engineering and the 19 courses they are enrolled in during the first semester with the 9 instructors responsible for those courses. There are 5 more courses than for the first tests of the first and second models. Here are the courses I added:

- The course "Economie politique – Microéconomie – TP" has been split into "Economie politique - Microéconomie - TP - groupe 1" and "Economie politique - Microéconomie - TP - groupe 2". The objective is that the two groups are considered as this is the only course that is divided between two groups.

- The courses "Economie politique - Microéconomie - Remédiation", "Finance et comptabilité - Questions / Réponses", "Finance et comptabilité - Remédiation", "Mathématiques: Analyse infinitésimale – Remédiation" have been added to the instance so that this one is more complete.

The course "Cours de langue" has been replaced by "Espagnol 1". The aim, later, is to consider all the different language courses that the student can potentially choose.

The table "Type" is exactly the same as it was for the first test of the version two of the ILP model which means that the courses "Mathématiques: Analyse infinitésimale – 1" and "Mathématiques: Analyse infinitésimale – 2", "Physique générale : partim 1 – théorie 1" and "Physique générale : partim 1 – théorie 2", and finally, "Physique générale : partim 1 – TP 1" and "Physique générale : partim 1 – TP 2" cannot be given within the same day.

For the availability, I used exactly what I used for the first test with less availability I made on the second model.

Concerning the students, I considered two students who are both enrolled in every course except that the first student is enrolled in “Economie politique – Microéconomie – TP – group 1”, while the other one is registered in the course “Economie politique – Microéconomie – TP – groupe 2”.

I ran the code and got these results (*fig. 23*) :

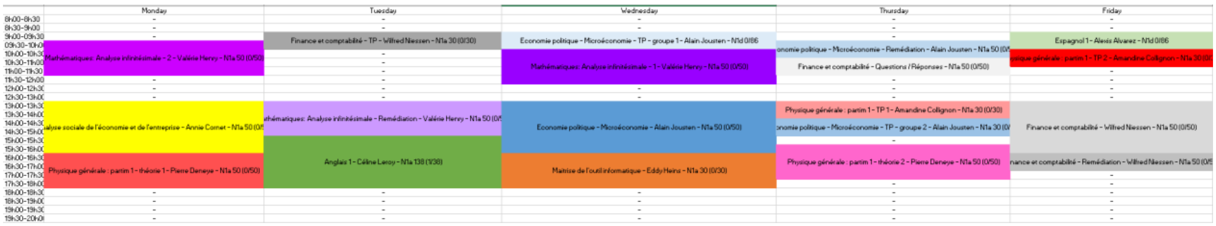


Figure 23. Schedule – 3.A. – Bloc 1 of Business Engineering – *Results V3*

It took 2.98 seconds to for the ILP to be optimized, and the objective value is 0, which means that there are no course conflicts.

Compared to the results of the first test of the first model, after implementing the constraint that prevents courses from being given before 9 am, after 6 pm, and between 12 am and 1 pm and compared to the results of the first test of the second model after removing availabilities for the first time, there are no courses that are given during the same time slots.

New classrooms that have not been assigned for the first tests of the other models are now considered for some courses. This is the case for the course “Anglais 1” which has been attributed the classroom “N1a 138 (1/38)” which has a capacity of 120 students, the courses “Economie politique - Microéconomie - TP - groupe 1” and “Espagnol 1” that have been assigned the classroom “N1d 0/86” that has a capacity of 108 students. Moreover, the course “Finance et comptabilité – TP”, “Physique générale : partim 1 – TP 1” and “Physique générale : partim 1 – TP 2” have been assigned classroom “N1a 30 (0/30)” which has a capacity of 228 students. These courses have never had this classroom assigned during the previous tests.

These new classroom assignments are due to the fact that the number of students registered for these courses has changed compared to the previous tests. In fact, the number of students for the course “Anglais 1” changed from 213 to 43:

$$\frac{243}{5} = 48.6$$

As it is planned to have 5 English courses given by 5 different instructors, at the same time.

As the course “Economie politique - Microéconomie – TP” has been split into two separate courses, the number of students per course has been split into two too :

$$\frac{201}{2} = 100.5$$

Concerning the course “Finance et comptabilité - TP”, as it is supposed to be given to three different groups of students by the same instructor, the number of students enrolled in the course has changed from 229 to 77 because :

$$\frac{229}{3} = 76.33$$

Like the “Finance et comptabilité – TP” course, the “Physique générale : partim 1 - TP 1” and “Physique générale : partim 1 - TP 2” courses should be given three times a week, which is why their number of students changed from 231 to 77 as :

$$\frac{231}{3} = 77$$

Previously, all the students were grouped in the course “Cours de langue”, regardless of possible groups. However, it is planned to consider every different language course as a unique course. In this instance, I only considered the course “Espagnol 1” which is supposed to be given to 248 students, both from business engineering and economics and management sciences. We also know that the course “Espagnol 1” is supposed to be given five times a week to five different student groups. Therefore, the number of students for the course “Cours de langue” changed from 248 to 50 for the course “Espagnol 1” :

$$\frac{248}{5} = 49.6$$

Finally, even though its classroom assignment has already been assigned to it previously, its number of students changed from 191 to 64 because it is supposed to be given three times a week to three different student groups by the same instructor.

9.3.1.1. Complete version of the results

By adding the four other “Anglais 1” groups, the 2 other “Maitrise de l’outil informatique” groups, the two other “Finance et comptabilité – TP” groups, the two other “Physique générale : partim 1 - TP 1” and “Physique générale : partim 1 - TP 2” groups and the language courses “Allemand 1”, “Allemand 1+”, “Chinois 1 (anglaise)”, “Espagnol 1+”, “Italien 1”, “Néerlandais débutant” and “Néerlandais 1”, I got the following schedule (*fig. 24*) :

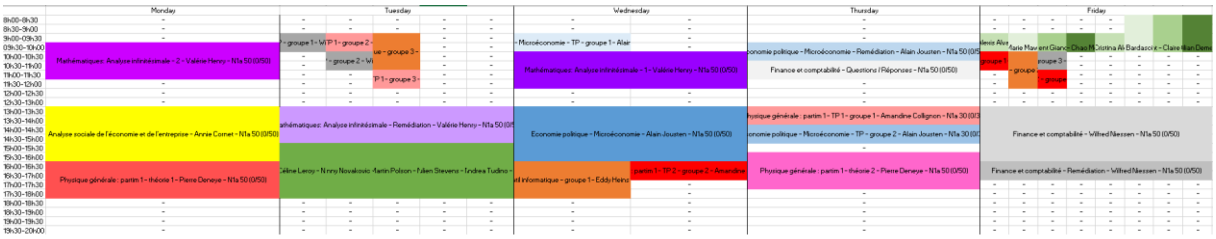


Figure 24. Schedule – 3.A. – Bloc 1 of Business Engineering – complete version – *Results V3*

I started by adding the two other “Maitrise de l’outil informatique” groups as they were the courses divided into groups that must be given during different time slots that have the highest duration, which is here, four time slots. I put them when the first group had other courses, for example, “Finance et comptabilité - TP” and “Physique générale : partim 1 - TP 2”.

Then, whenever, there were enough free time slots to put either “Finance et comptabilité – TP”, “Physique générale : partim 1 - TP 1” or “Physique générale : partim 1 - TP 2” courses for one of the groups, I integrated them in the planning. Finally, I assigned a classroom that had enough capacity to accommodate every student in the course to each course.

Monday (*fig. 25*):

	Monday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	-
09h30-10h00	Mathématiques: Analyse infinitésimale - 2 - Valérie Henry - N1a 50 (0/50)
10h00-10h30	
10h30-11h00	
11h00-11h30	
11h30-12h00	-
12h00-12h30	-
12h30-13h00	-
13h00-13h30	Analyse sociale de l'économie et de l'entreprise - Annie Cornet - N1a 50 (0/50)
13h30-14h00	
14h00-14h30	
14h30-15h00	
15h00-15h30	
15h30-16h00	
16h00-16h30	Physique générale : partim 1 - théorie 1 - Pierre Deneye - N1a 50 (0/50)
16h30-17h00	
17h00-17h30	
17h30-18h00	
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 25. Schedule – 3.A. – Bloc 1 of Business Engineeingr – Monday – *Results V3*

On Monday, no additional courses have been added as these are courses that are given to the full bloc 1 of business engineer.

Tuesday (*fig. 26*):

	Tuesday				
8h00-8h30	-	-	-	-	-
8h30-9h00	-	-	-	-	-
9h00-09h30	Finance et comptabilité - TP - groupe 1 - Wilfred Niessen - N1a 30 (0/30)	Physique générale : partim 1 - TP 1 - groupe 2 - Amandine Collignon - N1d 1/82	Maitrise de l'outil informatique - groupe 3 - Eddy Heins - N1a 126 (1/26)	-	-
09h30-10h00				-	-
10h00-10h30	-	Finance et comptabilité - TP - groupe 2 - Wilfred Niessen - N1a 130 (1/30)		-	-
10h30-11h00	-			-	-
11h00-11h30	-	-	Physique générale : partim 1 - TP 1 - groupe 3 - Amandine Collignon - N1d 1/82	-	-
11h30-12h00	-	-		-	-
12h00-12h30	-	-	-	-	-
12h30-13h00	-	-	-	-	-
13h00-13h30	Mathématiques: Analyse infinitésimale - Remédiation - Valérie Henry - N1a 50 (0/50)				
13h30-14h00					
14h00-14h30					
14h30-15h00					
15h00-15h30	Anglais 1 - Céline Leroy - N1a 138 (1/38)	Anglais 1 - Fanny Novakovic - N1a 115 (1/15)	Anglais 1 - Martin Polson - N1a 119 (1/19)	Anglais 1 - Julien Stevens - N1a 120 (1/20)	Anglais 1 - Andrea Tudino - N1a 15 (0/15)
15h30-16h00					
16h00-16h30					
16h30-17h00					
17h00-17h30					
17h30-18h00					
18h00-18h30	-	-	-	-	-
18h30-19h00	-	-	-	-	-
19h00-19h30	-	-	-	-	-
19h30-20h00	-	-	-	-	-

Figure 26. Schedule – 3.A. – Bloc 1 of Business Engineering – Tuesday – *Results V3*

On Tuesday, as the course "Anglais 1" given by Ms. Céline Leroy has been assigned to this day, I added the four other "Anglais 1" courses. I simply put them at the same time slots as the first one, adapted the instructor and to choose the classroom, I took classrooms that were sufficiently big to accommodate all the students registered in the course. This was simple as the only constraint that I had to respect was that this classroom could not be assigned to another English course at the same time.

As we can see in the figure, from 9h00 to 11h00, courses are given at the same time but to different groups, in different classrooms. This way, there are no course conflicts.

Wednesday (*fig.27*):

	Wednesday	
8h00-8h30	-	-
8h30-9h00	-	-
9h00-09h30	Economie politique - Microéconomie - TP - groupe 1 - Alain Jousten - N1d 0/86	-
09h30-10h00		-
10h00-10h30	Mathématiques: Analyse infinitésimale - 1 - Valérie Henry - N1a 50 (0/50)	
10h30-11h00		
11h00-11h30		
11h30-12h00		
12h00-12h30	-	-
12h30-13h00	-	-
13h00-13h30	Economie politique - Microéconomie - Alain Jousten - N1a 50 (0/50)	
13h30-14h00		
14h00-14h30		
14h30-15h00		
15h00-15h30		
15h30-16h00		
16h00-16h30	Maitrise de l'outil informatique - groupe 1 - Eddy Heins - N1a 30 (0/30)	Physique générale : partim 1 - TP 2 - groupe 2 - Amandine Collignon - N1d 1/82
16h30-17h00		
17h00-17h30		-
17h30-18h00		-
18h00-18h30	-	-
18h30-19h00	-	-
19h00-19h30	-	-
19h30-20h00	-	-

Figure 27. Schedule – 3.A. – Bloc 1 of Business Engineering – Wednesday – *Results V3*

At the same time as the course "Maitrise de l'outil informatique – group 1" is given, I added the course "Physique Générale : partim 1 – TP 2 – group 2" with a different classroom so that they can both be given at the same time.

Thursday (*fig. 28*):

	Thursday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	-
09h30-10h00	Economie politique - Microéconomie - Remédiation - Alain Jousten - N1a 50 (0/50)
10h00-10h30	
10h30-11h00	Finance et comptabilité - Questions / Réponses - N1a 50 (0/50)
11h00-11h30	
11h30-12h00	-
12h00-12h30	-
12h30-13h00	-
13h00-13h30	Physique générale : partim 1 - TP 1 - groupe 1 - Amandine Collignon - N1a 30 (0/30)
13h30-14h00	
14h00-14h30	Economie politique - Microéconomie - TP - groupe 2 - Alain Jousten - N1a 30 (0/30)
14h30-15h00	
15h00-15h30	-
15h30-16h00	Physique générale : partim 1 - théorie 2 - Pierre Deneye - N1a 50 (0/50)
16h00-16h30	
16h30-17h00	
17h00-17h30	
17h30-18h00	-
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 28. Schedule – 3.A. – Bloc 1 of Business Engineering – Thursday – *Results V3*

No additional courses have been added on Thursday.

Friday (*fig. 29*):

	Friday							
8h00-8h30	-	-	-	-	-	Italien 1 - Alex Bardascino - N1d 1/86	Néerlandais débutant - Claire Cutaia - N1a 115 (1/15)	Néerlandais 1 - Kilian Demey - N1a 120 (1/2
8h30-9h00	-	-	-	-	-			
9h00-09h30	Espagnol 1 - Alexis Alvarez - N1d 0/86	Allemand 1 - Marie Mawhin - N1d 0/89	Allemand 1+ - Laurent Gianotti - N1a 224 (2/24)	Chinois 1 (anglais) - Chao Meng - N1a 34 (0/34	Espagnol 1+ - Cristina Alonso - N1d 3/91			
09h30-10h00								
10h00-10h30	Physique générale : partim 1 - TP 2 - groupe 1 - Amandine Collignon - N1a 30 (0/30)	Maitrise de l'outil informatique - groupe 2 - Eddy Heins - N1a 126 (1/2	Finance et comptabilité - TP - groupe 3 - Wilfred Niessen - N1a 130 (1/30)	-	-	-	-	-
10h30-11h00				-	-	-	-	-
11h00-11h30	-		Physique générale : partim 1 - TP 2 - groupe 3 - Amandine Collignon - N1d 1/8	-	-	-	-	-
11h30-12h00	-			-	-	-	-	-
12h00-12h30	-	-	-	-	-	-	-	-
12h30-13h00	-	-	-	-	-	-	-	-
13h00-13h30	Finance et comptabilité - Wilfred Niessen - N1a 50 (0/50)							
13h30-14h00								
14h00-14h30								
14h30-15h00								
15h00-15h30								
15h30-16h00								
16h00-16h30	Finance et comptabilité - Remédiation - Wilfred Niessen - N1a 50 (0/50)							
16h30-17h00								
17h00-17h30	-	-	-	-	-	-	-	-
17h30-18h00	-	-	-	-	-	-	-	-
18h00-18h30	-	-	-	-	-	-	-	-
18h30-19h00	-	-	-	-	-	-	-	-
19h00-19h30	-	-	-	-	-	-	-	-
19h30-20h00	-	-	-	-	-	-	-	-

Figure 29. Schedule – 3.A. – Bloc 1 of Business Engineering – Friday – *Results V3*

At the same time as the “Espagnol 1” course, I added the other language courses with their respective instructors and different classrooms that had enough capacity to accommodate all the students registered in the respective courses. However, the course “Espagnol 1” is given during only two time slots while the courses “Italien 1”, “Néerlandais débutant” and “Néerlandais 1” are given during four time slots. Fortunately, no courses were planned to be given from 8h00 to 9h00. Consequently, I scheduled them to start at 8h00 so that they finish at the same time as the other language courses.

In addition, from 10h00 to 12h00 some courses are given at the same time but to different student groups, by different instructors, and in different classrooms.

9.3.2. *Second test – 2 student groups*

For this new instance “*Information_needed – 3.B.”*, I added the 11 courses given to the second bloc of business engineering and with the 6 new instructors responsible for those courses in addition to the courses assigned to the first bloc of business engineer.

The course “Cours de langue 2” has been replaced by “Espagnol 2” as I want to have a complete model and add later the other language courses. Moreover, I set the instructor of this course to “Alexis Alvarez”. However, in reality, this instructor does not give this course to the second bloc of business engineering. I did this so that the courses “Espagnol 1” and “Espagnol 2” are not given at the same time because some other language courses are given by the same instructors to different students’ blocs. This is the case for the Italian course for example that is always given by Mr. Alex Bardascino.

The course “Anglais 2” does not have the same number of students as in the previous ILP model tests. It changed from 78 to 20 as this course is supposed to be given to four different groups.

$$\frac{78}{4} = 19.5$$

In addition, the “Espagnol 2” course’s number of students has changed from 86 to 29 as the course is supposed to be given to three different student groups and:

$$\frac{86}{3} = 28.66$$

As for the second tests of the second ILP model, in the “Type” sheet of the Excel file gathering the data, I added the courses “Comptabilité analytique et contrôle de gestion – 1” and “Comptabilité analytique et contrôle de gestion – 2”, and the courses “Mathématiques pour ingénieurs de gestion - théorie 1” and “Mathématiques pour ingénieurs de gestion - théorie 2” as they should not be given within the same day.

Finally, the availability of the new instructors is the same as what I had after removing the first availabilities during the second test of the second model.

I ran the code and got the following results:

For the first bloc of business engineering (*fig. 30*):

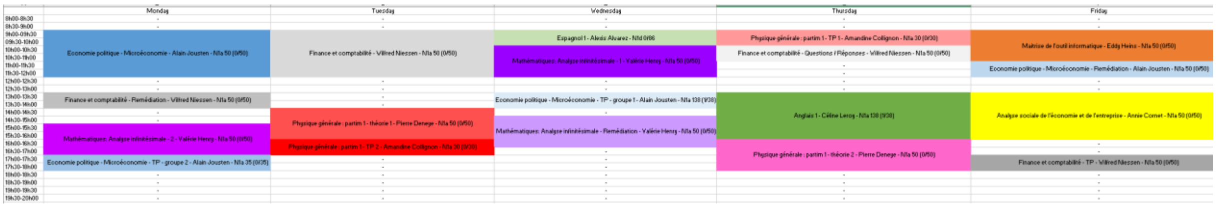


Figure 30. Schedule – 3.B. – Bloc 1 of Business Engineering – *Results V3*

For the second bloc of business engineering (*fig. 31*):

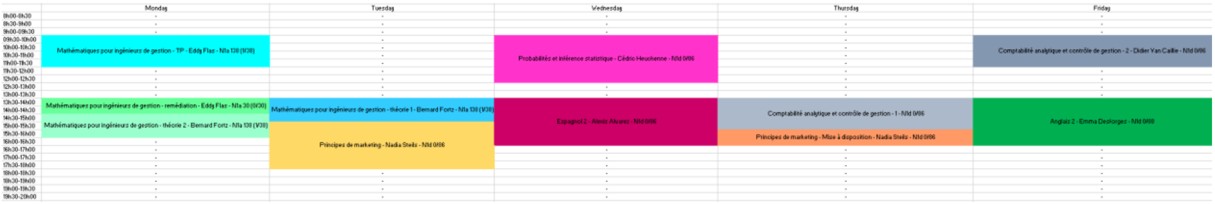


Figure 31. Schedule – 3.B. – Bloc 2 of Business Engineering – *Results V3*

It took 37.6 seconds in total for the ILP to be optimized, and the objective value is 0, which means that there are no course conflicts.

By comparing these results with the results, I got with the second tests of the first and second ILP model, I observed that no courses are given at the same time slots. However, by comparing with the results of the first tests of this third ILP model, I can say that "5.26% of courses are given at the same time slots for the first bloc of business engineer". In other words, only one course has the same time slots assigned as the previous test we made on the 3rd version of the ILP.

I can also say that 78.95% of courses kept the same classroom as during the previous test for the first bloc of business engineering.

9.3.2.1. Complete version of the results

By adding the missing courses of the first bloc of business engineering as I did for the previous test, and the language courses for the second bloc of business engineering which are "Allemand 2", "Allemand avancé 2", "Chinois 2", "Espagnol avancé 2", "Italien 2" and "Néerlandais 2" as well as the 3 other "Anglais 2" groups, I got the following planning :

For the first bloc of business engineering (*fig. 32*):

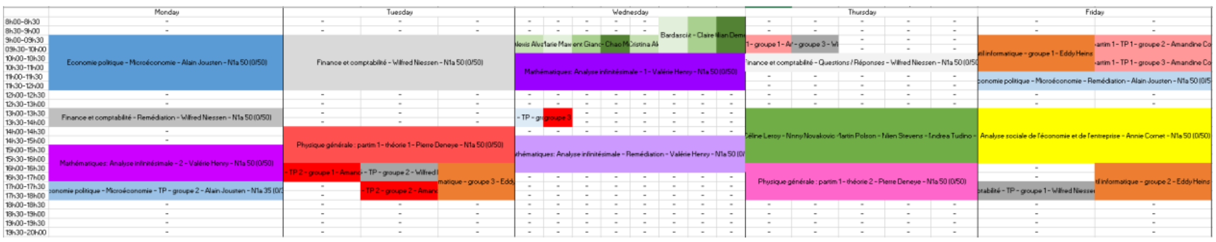


Figure 32. Schedule – 3.B. – Bloc 1 of Business Engineering – complete version – *Results V3*

For the second bloc of business engineering (*fig.33*):

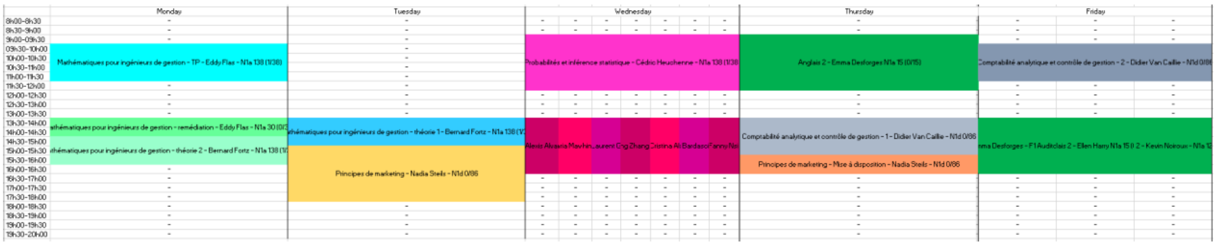


Figure 33. Schedule – 3.B. – Bloc 2 of Business Engineering – complete version – *Results V3*

As I did not consider the second bloc of business engineering during the previous instance, I will focus on this one.

Monday (*fig. 34*):

	Monday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	-
09h30-10h00	Mathématiques pour ingénieurs de gestion - TP - Eddy Flas - N1a 138 (1/38)
10h00-10h30	
10h30-11h00	
11h00-11h30	
11h30-12h00	-
12h00-12h30	-
12h30-13h00	-
13h00-13h30	-
13h30-14h00	Mathématiques pour ingénieurs de gestion - remédiation - Eddy Flas - N1a 30 (0/30)
14h00-14h30	
14h30-15h00	Mathématiques pour ingénieurs de gestion - théorie 2 - Bernard Fortz - N1a 138 (1/38)
15h00-15h30	
15h30-16h00	
16h00-16h30	-
16h30-17h00	-
17h00-17h30	-
17h30-18h00	-
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 34. Schedule – 3.B. – Bloc 2 of Business Engineering – Monday – *Results V3*

No additional courses have been added.

Tuesday (*fig. 35*):

	Tuesday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	-
09h30-10h00	-
10h00-10h30	-
10h30-11h00	-
11h00-11h30	-
11h30-12h00	-
12h00-12h30	-
12h30-13h00	-
13h00-13h30	-
13h30-14h00	athématiques pour ingénieurs de gestion - théorie 1 - Bernard Fortz - N1a 138 (1/
14h00-14h30	
14h30-15h00	
15h00-15h30	Principes de marketing - Nadia Steils - N1d 0/86
15h30-16h00	
16h00-16h30	
16h30-17h00	
17h00-17h30	
17h30-18h00	
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 35. Schedule – 3.B. – Bloc 2 of Business Engineering – Tuesday – *Results V3*

No additional courses have been added.

Wednesday (*fig. 36*):

	Wednesday						
8h00-8h30	-	-	-	-	-	-	-
8h30-9h00	-	-	-	-	-	-	-
9h00-09h30	Probabilités et inférence statistique - Cédric Heuchenne - N1a 138 (1/38)						
09h30-10h00							
10h00-10h30							
10h30-11h00							
11h00-11h30							
11h30-12h00							
12h00-12h30	-	-	-	-	-	-	-
12h30-13h00	-	-	-	-	-	-	-
13h00-13h30	-	-	-	-	-	-	-
13h30-14h00	Espagnol 2 - Alexis Alvarez - N1d 0/86	Allemand 2 - Maria Mawhin -N1a 224 (2/24)	Allemand avancé 2 - Laurent Gianotti - N1a 120 (1/20)	Chinois 2 - Hong Zhang - N1a 34 (0/34)	Espagnol avancé 2 - Cristina Alonso Martin - N1d 3/91	Italien 2 - Alex Bardascino - N1d - 1/86	Néerlandais 2 - Fanny Nsita - N1a 115 (1/15)
14h00-14h30							
14h30-15h00							
15h00-15h30							
15h30-16h00							
16h00-16h30							
16h30-17h00	-	-	-	-	-	-	-
17h00-17h30	-	-	-	-	-	-	-
17h30-18h00	-	-	-	-	-	-	-
18h00-18h30	-	-	-	-	-	-	-
18h30-19h00	-	-	-	-	-	-	-
19h00-19h30	-	-	-	-	-	-	-
19h30-20h00	-	-	-	-	-	-	-

Figure 36. Schedule – 3.B. – Bloc 2 of Business Engineering – Wednesday – *Results V3*

At the same time as the course “Espagnol 2”, I added the courses “Allemand 2”, “Allemand avancé 2”, “Chinois 2”, “Espagnol avancé 2”, “Italien 2” and “Néerlandais 2” with their respective instructor. I assigned classrooms that were free at that time and that had enough capacity to accommodate all the students enrolled in the course.

Thursday (*fig. 37*):

	Thursday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	Anglais 2 - Emma Desforges N1a 15 (0/15)
09h30-10h00	
10h00-10h30	
10h30-11h00	
11h00-11h30	
11h30-12h00	
12h00-12h30	-
12h30-13h00	-
13h00-13h30	-
13h30-14h00	Comptabilité analytique et contrôle de gestion - 1 - Didier Van Caillie - N1d 0/86
14h00-14h30	
14h30-15h00	
15h00-15h30	
15h30-16h00	Principes de marketing - Mise à disposition - Nadia Steils - N1d 0/86
16h00-16h30	
16h30-17h00	-
17h00-17h30	-
17h30-18h00	-
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 37. Schedule – 3.B. – Bloc 2 of Business Engineering – Thursday – *Results V3*

As the "Anglais 2" course has three instructors assigned but has to be given four times, I added it once from 9h00 to 12h00 on Thursday as nothing else was planned and the instructor was available at that time.

Friday (*fig. 38*):

		Friday	
8h00-8h30	-	-	-
8h30-9h00	-	-	-
9h00-09h30	-	-	-
09h30-10h00	Comptabilité analytique et contrôle de gestion - 2 - Didier Van Caillie - N1d 0/86		
10h00-10h30			
10h30-11h00			
11h00-11h30			
11h30-12h00	-	-	-
12h00-12h30	-	-	-
12h30-13h00	-	-	-
13h00-13h30	-	-	-
13h30-14h00	Anglais 2 - Emma Desforges - F1 Auditoire B03 (1/33)	Anglais 2 - Ellen Harry N1a 15 (0/15)	Anglais 2 - Kevin Noiroux - N1a 120 (1/20)
14h00-14h30			
14h30-15h00			
15h00-15h30			
15h30-16h00			
16h00-16h30			
16h30-17h00	-	-	-
17h00-17h30	-	-	-
17h30-18h00	-	-	-
18h00-18h30	-	-	-
18h30-19h00	-	-	-
19h00-19h30	-	-	-
19h30-20h00	-	-	-

Figure 38. Schedule – 3.B. – Bloc 2 of Business Engineeingr – Friday – *Results V3*

I added two other "Anglais 2" groups at the same time as the first one but with the two other instructors. I also assigned classrooms that were free at those time slots.

9.3.3. Third test – 3 student groups

For this new instance "*Information_needed – 3.C.*", I added the 12 courses of the third bloc of business engineering with the 7 new instructors responsible for those courses in addition to the courses assigned to the first and second blocs of business engineer and to the instructors already considered for the previous blocs.

I replaced the course “Cours de langue 3” with the course “Espagnol 3” and I put “Alexis Alvarez” as the instructor so that this course cannot be given at the same time as the other language courses.

I added the course “Introduction à la programmation” because, students of the third bloc of business engineering have the option to take, during the first semester, “Introduction à l’électronique” or “Introduction à la programmation” or other courses but during the second semester. As the course “Introduction à la programmation” is supposed to be given during 2 blocs of 4 time slots and one bloc of 7 time slots, I only added the one of 7 time slots in the instance, and I will add manually the two other ones at the same time slots as the two courses “Introduction à l’électronique” as their duration is 4 time slots.

The new courses in the “Type” sheet of the Excel file that gathers the data are the courses “Operations Research – 1” and “Operations Research – 2” as for the third test of the second model.

The number of students for the “Anglais 3” course changed from 66 to 17 as it is supposed to be given to four different groups of students and :

$$\frac{66}{4} = 17$$

The number of students for the “Espagnol 3” course which was earlier called “Cours de langue 3” also decreased from 119 to 40 as it is supposed to be given 3 times a week.

$$\frac{119}{4} = 39.66$$

In the “Type” table, I added the courses “Introduction à l’électronique – 1” and “Introduction à l’électronique – 2”, as well as “Operations Research – 1” and “Operations Research – 2” as they cannot be given within the same day.

Finally, I put the same availabilities for the new instructors as what I had after removing the first availabilities during the third test of the second model. However, the instructor “Benoît Donnet” who is responsible for the course “Introduction à la programmation” was not in that instance, which is why I set full availability for him.

I ran the code and here are the results I got:

For the first bloc of business engineering (*fig. 39*) :

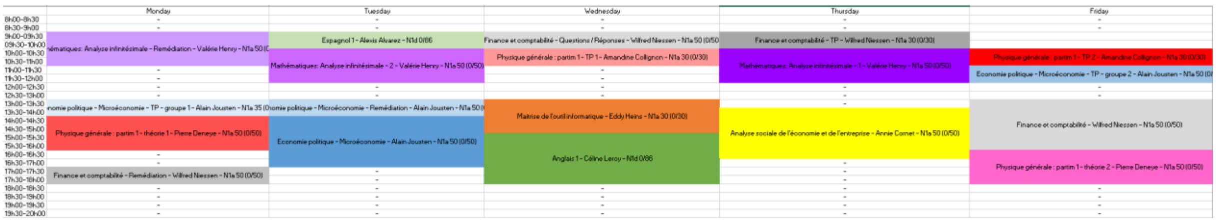


Figure 39. Schedule – 3.C. – Bloc 1 of Business Engineering – *Results V3*

For the second bloc of business engineering (*fig. 40*):

Figure 40. Schedule – 3.C. – Bloc 2 of Business Engineering – *Results V3*

For the third bloc of business engineering (*fig. 41*) :

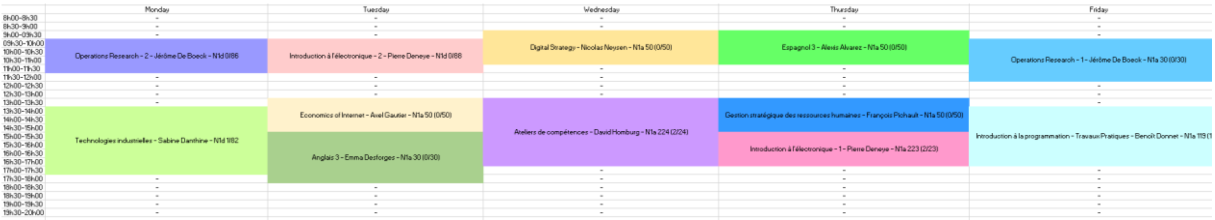


Figure 41. Schedule – 3.C. – Bloc 3 of Business Engineering – *Results V3*

It took 152.04 seconds in total for the ILP to be optimized, and the objective value is 0, which means that there are no course conflicts.

Once again, the bigger is the instance, the higher is the time required for the ILP to be optimized and as already mentioned in the analysis of the previous ILP model, the higher is the number of constraints, the higher is the time required for the ILP to be optimized.

Considering only the results for the first and the second blocs of business engineering, I noticed that no courses had the same time slots assigned as during the previous test and that 73.33% of the courses has the same classroom as the one, they had during the previous test.

9.3.3.1. Complete version of the results

By adding the missing courses for the first and the second blocs of business engineering as well as the other language courses that should be given at the same time as “Espagnol 3”, which are “Allemand 3”, “Allemand avancé 3”, “Espagnol avancé 3”, “Italien 3”, “Néerlandais 3” and “Chinois 3 (anglaise)”, as well as the two other courses of “Introduction à la programmation” and the three other “Anglais 3” courses. Here are the schedules I got :

For the first bloc of business engineering (*fig. 42*):

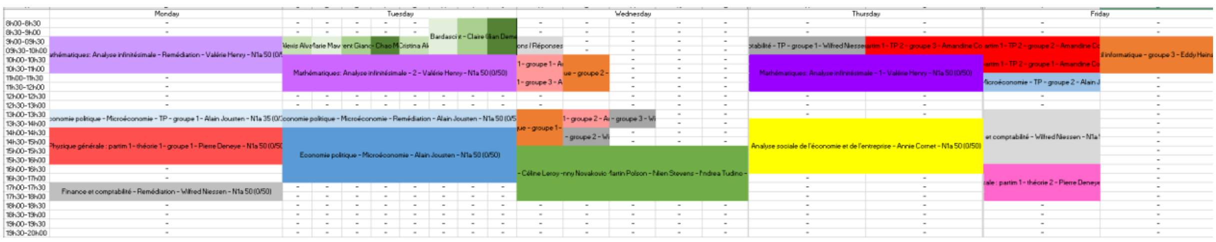


Figure 42. Schedule – 3.C. – Bloc 1 of Business Engineering – complete version – *Results V3*

For the second bloc of business engineering (*fig. 43*):

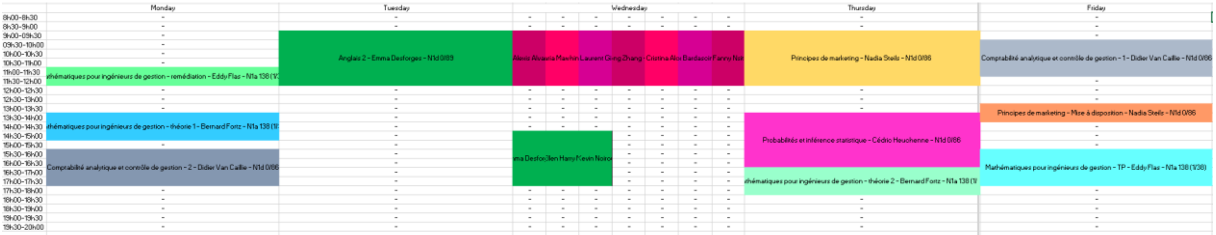


Figure 43. Schedule – 3.C. – Bloc 2 of Business Engineering – complete version – *Results V3*

For the third bloc of business engineering *(fig. 44*):

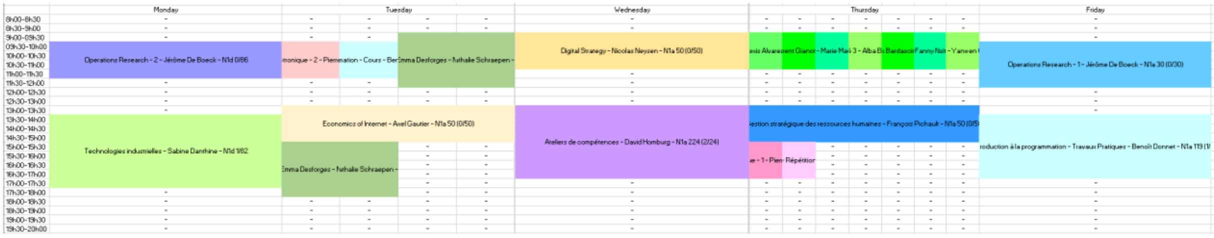


Figure 44. Schedule – 3.C. – Bloc 3 of Business Engineering – complete version – *Results V3*

As considering the third bloc of business engineering is new to this instance, I will focus on this bloc.

Monday (*fig. 45*):

	Monday
8h00-8h30	-
8h30-9h00	-
9h00-09h30	-
09h30-10h00	Operations Research - 2 - Jérôme De Boeck - N1d 0/86
10h00-10h30	
10h30-11h00	
11h00-11h30	
11h30-12h00	-
12h00-12h30	-
12h30-13h00	-
13h00-13h30	-
13h30-14h00	Technologies industrielles - Sabine Danthine - N1d 1/82
14h00-14h30	
14h30-15h00	
15h00-15h30	
15h30-16h00	
16h00-16h30	
16h30-17h00	
17h00-17h30	
17h30-18h00	-
18h00-18h30	-
18h30-19h00	-
19h00-19h30	-
19h30-20h00	-

Figure 45. Schedule – 3.C. – Bloc 3 of Business Engineering – Monday – *Results V3*

There are no new courses.

Tuesday (*fig. 46*):

<table>
<tr><th></th><th colspan="4">Tuesday</th></tr>
<tr><td>8h00-8h30</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>8h30-9h00</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>9h00-09h30</td><td>-</td><td>-</td><td rowspan="6">Anglais 3 - Emma Desforges - N1a 30 (0/30)</td><td rowspan="6">Anglais 3 - Nathalie Schraepen - N1a 35 (0/35)</td></tr>
<tr><td>09h30-10h00</td><td rowspan="4">Introduction à l'électronique - 2 - Pierre Deneye - N1d 0/88</td><td rowspan="4">Introduction à la programmation - Cours - Benoît Donnet - N1a 119 (1/19)</td></tr>
<tr><td>10h00-10h30</td></tr>
<tr><td>10h30-11h00</td></tr>
<tr><td>11h00-11h30</td></tr>
<tr><td>11h30-12h00</td><td>-</td><td>-</td></tr>
<tr><td>12h00-12h30</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>12h30-13h00</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>13h00-13h30</td><td colspan="4" rowspan="4">Economics of Internet - Axel Gautier - N1a 50 (0/50)</td></tr>
<tr><td>13h30-14h00</td></tr>
<tr><td>14h00-14h30</td></tr>
<tr><td>14h30-15h00</td></tr>
<tr><td>15h00-15h30</td><td rowspan="6">Anglais 3 - Emma Desforges - N1a 30 (0/30)</td><td rowspan="6">Anglais 3 - Nathalie Schraepen - N1a 35 (0/35)</td><td>-</td><td>-</td></tr>
<tr><td>15h30-16h00</td><td>-</td><td>-</td></tr>
<tr><td>16h00-16h30</td><td>-</td><td>-</td></tr>
<tr><td>16h30-17h00</td><td>-</td><td>-</td></tr>
<tr><td>17h00-17h30</td><td>-</td><td>-</td></tr>
<tr><td>17h30-18h00</td><td>-</td><td>-</td></tr>
<tr><td>18h00-18h30</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>18h30-19h00</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>19h00-19h30</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>19h30-20h00</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table>

Figure 46. Schedule – 3.C. – Bloc 3 of Business Engineering – Tuesday – *Results V3*

I added the course “Introduction à la programmation – Cours” at the same time as the “Introduction à l’électronique – 2” course as they are supposed to be given to two different student groups.

Moreover, as the course “Anglais 3” is supposed to be given to four different groups by only two instructors, I added one group at the same time as the one we obtained by running the code (from 15h00 to 18h00) and I supposed that, as the students have the choice between “Introduction à l’électronique”, “Introduction à la programmation” that are given during the first semester and other courses that are given during the second semester, the students that did not register to one of the course given during the first semester could get the English course at the same time as those courses are given. That’s why I added them at the same time as those courses.

Wednesday (*fig.47*):

<table>
<tr><th></th><th>Wednesday</th></tr>
<tr><td>8h00-8h30</td><td>-</td></tr>
<tr><td>8h30-9h00</td><td>-</td></tr>
<tr><td>9h00-09h30</td><td rowspan="4">Digital Strategy - Nicolas Neysen - N1a 50 (0/50)</td></tr>
<tr><td>09h30-10h00</td></tr>
<tr><td>10h00-10h30</td></tr>
<tr><td>10h30-11h00</td></tr>
<tr><td>11h00-11h30</td><td>-</td></tr>
<tr><td>11h30-12h00</td><td>-</td></tr>
<tr><td>12h00-12h30</td><td>-</td></tr>
<tr><td>12h30-13h00</td><td>-</td></tr>
<tr><td>13h00-13h30</td><td rowspan="8">Ateliers de compétences - David Homburg - N1a 224 (2/24)</td></tr>
<tr><td>13h30-14h00</td></tr>
<tr><td>14h00-14h30</td></tr>
<tr><td>14h30-15h00</td></tr>
<tr><td>15h00-15h30</td></tr>
<tr><td>15h30-16h00</td></tr>
<tr><td>16h00-16h30</td></tr>
<tr><td>16h30-17h00</td></tr>
<tr><td>17h00-17h30</td><td>-</td></tr>
<tr><td>17h30-18h00</td><td>-</td></tr>
<tr><td>18h00-18h30</td><td>-</td></tr>
<tr><td>18h30-19h00</td><td>-</td></tr>
<tr><td>19h00-19h30</td><td>-</td></tr>
<tr><td>19h30-20h00</td><td>-</td></tr>
</table>

Figure 47. Schedule – 3.C. – Bloc 3 of Business Engineering – Wednesday – *Results V3*

No additional courses have been added.

Thursday (*fig. 48*):

Figure 48. Schedule – 3.C. – Bloc 3 of Business Engineering – Thursday – *Results V3*

I added the other language courses, which are “Allemand 3”, “Allemand avancé 3”, “Espagnol avancé 3”, “Italien 3”, “Néerlandais 3” and “Chinois 3 (anglais)”, at the same time as “Espagnol 3”.

In addition, I set the course “Introduction à la programmation – Répétition” at the same time as the course “Introduction à l’électronique – 1” as the students registered for one course cannot be given the other one.

I assigned classrooms that were available and that had enough capacity to welcome the students registered for the course.

Friday (*fig.49*):

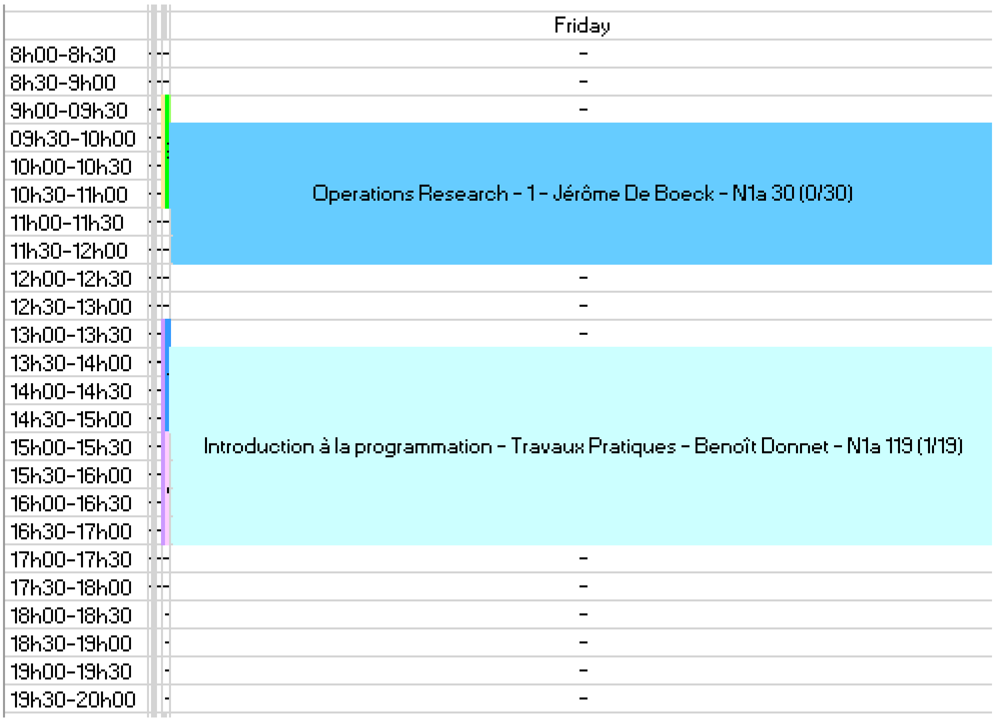


Figure 49. Schedule – 3.C. – Bloc 3 of Business Engineer – Friday – *Results V3*

No additional courses have been added.

9.3.4. Limitations of this third model

Even though this version of the ILP model is the most complete one, it presents some limitations.

The first one is the size of the instance that is limited. While building the first instance to make the first tests on the model, I encountered a few problems. My idea was to build an instance *“Information_needed – 243 students”* that was as close to reality as possible which means that I had

243 individual students, 45 courses as I had split all the courses that needed to be divided between different groups, 23 instructors, a table 243x45 that represented the enrolment of students to courses and a table "same" in which was written the courses that needed to start at the same time. For example, the courses "Anglais 1 – groupe 1" and "Anglais 1 – groupe 2" as there are given by different instructors to different groups of students.

However, with such a huge instance, I was not able to find a solution even though I waited for more than 30 minutes.

Consequently, I decided to reduce the instance "*Information_needed – 12 students"* by decreasing the number of students to 12. Nevertheless, the instance was still too big to provide any solution. Then, I came out with an even more reduced instance which is the one I used for the tests as I got solutions from running the code.

This instance is composed of 2 students, 19 courses as I decided to simplify some courses and not consider every sub-group for every course. I had then 9 instructors and I removed the "same" table".

What was planned was to include this parameter in the mathematical model:

- *Same(c,c'): binary variable that takes value 1 if a course c and a course c' must be given at the same timeslots. For example, if there are 2 English courses given by two different groups of students by two different teachers that must be given at the same time.*

And this constraint:

- $Y(c,t,d) \times same(c',c) = Y(c',t,d) \times same(c,c') \ \forall c,c',t,d$ ensures that if two courses c and c' should start at the same time, then, it is the case.

However, as I removed some courses that needed to start at the same time to simplify the model, this parameter and this constraint were not useful anymore. Instead, I added the courses I removed from the instance, after running the code, in the schedule, at the same time as the courses they are supposed to begin with, by hand.

A more general limitation for every ILP result given by the optimization of the ILP model is that the code provides values of the variables and not a direct schedule. Every time I would get the results from the test of an instance, I had to interpret the results by myself to incorporate them into a schedule in Excel. The human intervention is still needed to interpret the results.

that I encountered for every code of every version is the fact that I needed to adapt some part of the code so that it would fit perfectly for the new instance.

9.4. Tests on the algorithm

To test the algorithm, I coded it in Julia and tested it on three different instances.

9.4.1. First instance – 1 student group

For the first test, I considered an instance "*AlgorithmA*" composed of 21 classrooms, only the first bloc of business engineering and the 19 courses they are enrolled in during the first semester with the 9 instructors responsible for those courses. The Type Table and the Availability Table are the same as they were in the first instance tested on the third ILP model. However, unlike in the first instance tested on the third ILP model, I considered students as a group and not as individuals.

I ran the code and got the following results (*fig.50*) :

	Monday	Tuesday	Wednesday	Thursday	Friday
8h00-8h30	-	-	-	-	-
8h30-9h00	-	-	-	-	-
9h00-09h30	Economie politique – Microéconomie – Alain Jousten – N1a 30 (0/30)	Mathématiques: Analyse infinitésimale – 2 – Valérie Henry – N1a 50 (0/50)	Analyse sociale de l'économie et de l'entreprise – Annie Cornet – N1a 30 (0/30	Maitrise de l'outil informatique – Eddy Heins – N1a 126 (1/36)	Finance et comptabilité – TP – Wilfred Niessen – N1d 1/82
09h30-10h00					
10h00-10h30					-
10h30-11h00					-
11h00-11h30		Physique générale : partim 1 – TP 1 – Amandine Collignon – N1d 1/82		Physique générale : partim 1 – TP 2 – Amandine Collignon – N1d 1/82	-
11h30-12h00					-
12h00-12h30	-	-	-	-	-
12h30-13h00	-	-	-	-	-
13h00-13h30	Economie politique – Microéconomie – Remédiation – Alain Jousten – N1a 30 (0/30)	Mathématiques: Analyse infinitésimale – Remédiation – Valérie Henry – N1a 50 (0/50)	Physique générale : partim 1 – théorie 2 – Pierre Deneye – N1a 50 (0/50)	Finance et comptabilité – Wilfred Niessen – N1a 50 (0/50)	-
13h30-14h00					-
14h00-14h30	Economie politique – Microéconomie – TP – groupe 1 – Alain Jousten – N1d 0/86				-
14h30-15h00					-
15h00-15h30	Economie politique – Microéconomie – TP – groupe 2 – Alain Jousten – N1d 0/86	Physique générale : partim 1 – théorie 1 – Pierre Deneye – N1a 50 (0/50)	Anglais 1 – Céline Leroy – N1a 15 (0/15)		-
15h30-16h00					-
16h00-16h30	Mathématiques: Analyse infinitésimale – 1 – Valérie Henry – N1a 50 (0/50)			Finance et comptabilité – Questions / Réponses – N1a 50 (0/50)	-
16h30-17h00					-
17h00-17h30		Espagnol 1 – Alexis Alvarez – N1a 126 (1/26)		Finance et comptabilité – Remédiation – Wilfred Niessen – N1a 50 (0/50)	-
17h30-18h00					-
18h00-18h30	-	-	-	-	-
18h30-19h00	-	-	-	-	-
19h00-19h30	-	-	-	-	-
19h30-20h00	-	-	-	-	-

Figure 50. Schedule – A. – Bloc 1 of Business Engineering – *Results algorithm*

By comparing the results with the ones obtained with the first instance with less availability tested on the second ILP model, I noticed that the only course that has the same time slots assigned is the course "Physique générale : partim 1 – TP 2", which means that only 7.14% of the courses that were already present in the other instance are located at the same time slots in the schedule, even though the availabilities were set the same way for each instructor in both instance. In addition, in the courses that were already present in the other instance, 42.86% kept the same classroom. The change in the number of students in some courses, for example in the course "Espagnol 1", can be an explanation of these changes observed in terms of classroom. There was a desire to have a number of students enrolled in courses closer to reality, so the instance has been adapted.

A more interesting instance to compare these results with is the one used for the first test on the third ILP model. The availabilities were set the same and the courses are also identical. Surprisingly, only the course "Mathématiques: Analyse infinitésimale – Remédiation" has the same time slots assigned, which represents 5.26% of the courses. In addition, 42.10% of the courses have the same classroom assigned. Moreover, within these courses that have the same classroom assigned, 100% have classroom "N1a 50 (0/50)" assigned. This is the classroom having the highest capacity and every course that haw a number of students enrolled higher than 228 had already been assigned classroom N1a 50 (0/50) in the other instance, knowing that 228 is the maximum number of students the second classroom having the highest capacity can accommodate.

Finally, after the first main loop to assign consecutive time slots, a day, and a classroom to courses, no courses were placed in the Unscheduled Courses Vector, which means that the second main loop trying to place the courses in the Unscheduled Courses Vector in the scheduled was not necessary and did not have any impact on the Scheduled Courses Dictionary.

9.4.2. Second instance – 2 student groups

For this new instance "*AlgorithmB*", I added the 11 courses given to the second bloc of business engineering and with the 6 new instructors responsible for those courses in addition to the courses assigned to the first bloc of business engineering. The availabilities of the instructors are set the same way it was for the second instance tested on the third ILP model.

I ran the code and got the following results:

For the first bloc of business engineering (*fig. 51*):

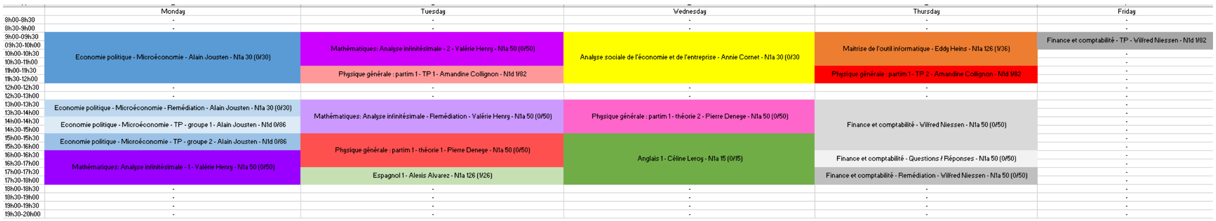


Figure 51. Schedule – B. – Bloc 1 of Business Engineering – *Results algorithm*

For the first bloc of business engineering (*fig. 52*):

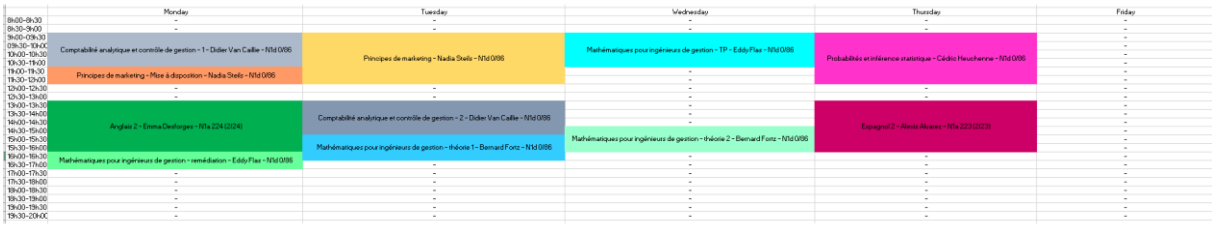


Figure 52. Schedule – B. – Bloc 2 of Business Engineering – *Results algorithm*

By comparing the results obtained for the first bloc of business engineering with the results of the previous instance, I observed that 100% of the courses have the same time slots and courses assigned. This can be explained by the fact the algorithm first treats the courses of the first bloc of business engineering in ascending order of the instructors related to the courses availabilities and as this order has not changed, the algorithm treated them the same way as previously.

Compared to the results of the second instance tested on the third ILP model, which had the same availabilities set to the instructors, I noticed that only one course, "Economie politique – Microéconomie" has the exact same time slots attributed, which represents 3.33% of the courses. Concerning the classrooms, 43.33% of the courses have the same classrooms assigned and within these courses, 51.54% have the "N1a 50 (0/50)" classroom attributed, which is all the courses that kept the same classroom from the first bloc of business engineer.

As for the previous instance, no courses were concerned by the loop trying to schedule the courses in the Unscheduled Courses Vector because they were all already scheduled which means that this loop had no impact on the courses in the Scheduled Courses Dictionary.

9.4.3. Third instance – 3 student groups

For this new instance "*AlgorithmC*", I added the 12 courses of the third bloc of business engineering with the 7 new instructors responsible for those courses in addition to the courses assigned to the first and second blocs of business engineering and to the instructors already considered for the previous blocs. The instructors' availabilities are the same as the ones that were set for the third instance tested on the third ILP model.

I ran the code and got the following results :

For the first bloc of business engineering (*fig. 53*):

Figure 53. Schedule – C. – Bloc 1 of Business Engineering – *Results algorithm*

For the second bloc of business engineering (*fig. 54*):

Figure 54. Schedule – C. – Bloc 2 of Business Engineering – *Results algorithm*

For the third bloc of business engineering (*fig. 55*):

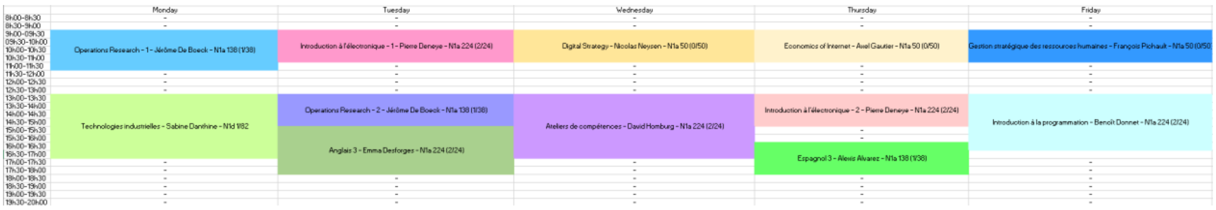


Figure 55. Schedule – C. – Bloc 3 of Business Engineering – *Results algorithm*

Compared to the previous results, 100% of the courses of the first and second bloc of business engineer have the same time slots and the same classrooms assigned.

By comparing the results with the ones obtained for the third instance on the third ILP model, I observed that 9.52% of the courses have the same time slots. Concerning the classrooms, 42.86% had the same classroom assigned. For the first bloc of business engineering, all the courses that have the classroom “N1a 50 (0/50)” assigned already had this classroom attributed during the third instance tested on the third ILP model. For the second bloc of business engineering, only some of the courses that have the classroom “N1d 0/86” had already this classroom assigned in the third instance tested on the third model.

As for the other models, no courses were concerned by the loop placing the courses of the Unscheduled Courses Vector as they were already all scheduled.

9.3.4.1. Less Availability

In the three instances tested, none of them needed the loop that tries to place the courses that are in the Unscheduled Courses Vector. However, I wanted to see the impact of this loop on the schedule. To constrain the algorithm, I removed all the availabilities linked to the time slots attributed to the courses in the results of the third test of the algorithm to create a new instance “*AlgorithmC’*”.

I ran the code and got the following results:

For the first bloc of business engineering (*fig. 56*):

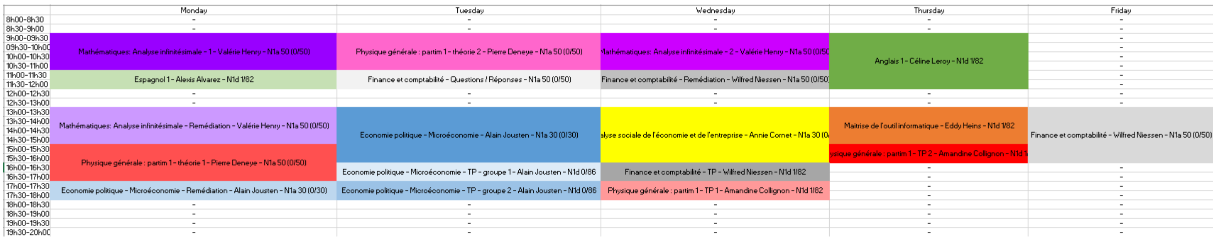


Figure 56. Schedule – C'. – Bloc 1 of Business Engineering – *Results algorithm*

For the second bloc of business engineering (*fig. 57*):

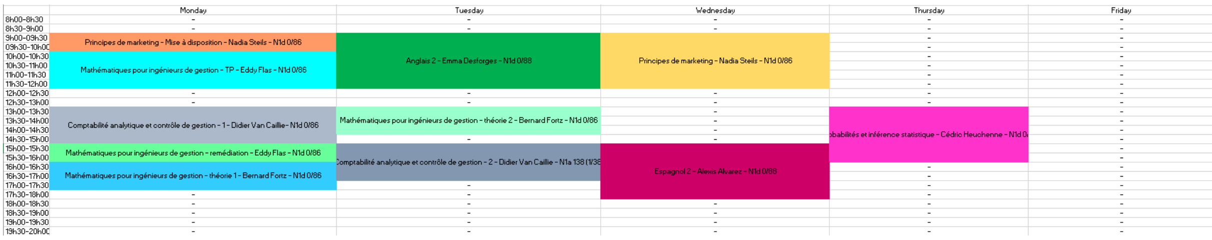


Figure 57. Schedule – C'. – Bloc 2 of Business Engineering – *Results algorithm*

For the third bloc of business engineering (*fig. 58*):

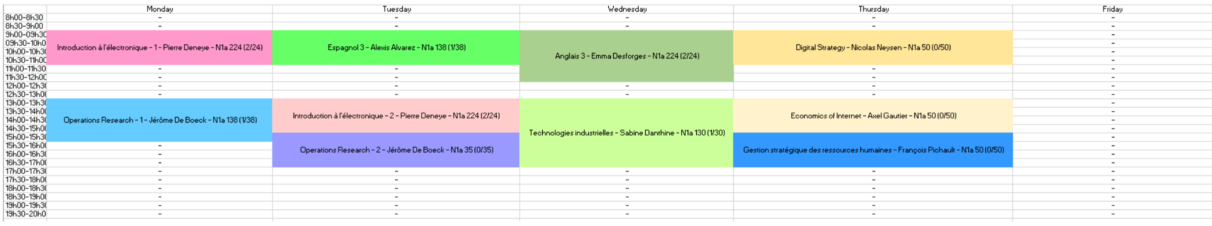


Figure 58. Schedule – C'. – Bloc 3 of Business Engineering – *Results algorithm*

Of course, as the time slots assigned to the courses in the previous results are now considered as unavailable for the courses, there are no courses that have the same time slots attributed. Moreover, some courses have a change in terms of classroom assigned:

- In the previous instance, the course "Technologies industrielles" had the classroom "N1d 1/82" assigned and now, it is the classroom "N1a 130 (1/30)". They both have a capacity of 120 students but classroom "N1d 1/82" is before classroom "N1a 130 (1/30)" in the Room Availability Matrix, which means that the algorithm will check first if the classroom "N1d 1/82" is free and if not, it will consider the next classroom which is the "N1a 130" one. This change can be explained by the fact that when the course "Technologies industrielles" is given, the courses "Finance et comptabilité – TP" and then the course "Physique Générale: partim 1 – TP 1" are given in classroom "N1d 1/82" which means it is available during the full duration of the course "Technologies industrielles".

- The course "Comptabilité Analytique et contrôle de gestion – 2" had the classroom "N1d 0/86", which has a capacity of 108 students, assigned in the previous instance and has now the classroom "N1a 138 (1/38)", which has a capacity of 120 students, assigned. This can be explained by the fact that, at a part of the time this course is given, the course "Economie politique – Microéconomie – TP – groupe 1" is given in the classroom "N1d 0/86", so as the classroom was not available at those time slots, the algorithm considered the next classroom in the Room Availability Matrix and this classroom is “N1a 138 (1/38)”.

- The course “Operations Research – 2” switched from the classroom “N1a 138 (1/38)” to the classroom “N1a 35 (0/35)”. They both have the same capacity, which is 80 students, however, while sorting the courses by order of capacity, the classroom “N1a 138 (1/38)” is before the classroom “N1a 35 (0/35)”. This classroom change can be explained by the fact that the course “Comptabilité Analytique et contrôle de gestion – 2” is given at the same time as the course “Operations Research – 2” but in the classroom “N1a 138 (1/38)”.

What can be observed is that the classroom assigned to courses remains the same in the different instances, provided that instructors’ availabilities remain the same. However, adding instructors’ unavailabilities provokes classroom changes.

Two courses have been placed in the Unscheduled Courses Vector by the Algorithm, these courses are “Ateliers de compétences” and “Introduction à la programmation”. This can seem surprising as Friday is a completelt free day for the third bloc of business engineering. However, on Friday, the “Ateliers de compétences” instructor has been set as unavailable from 1 pm until 5 pm and the “Introduction à la programmation” instructor has been set as unavailable from 1 pm until 4:30 pm. As it has been set that courses cannot be given before 9 am, between 12 am and 1 pm, and after 18 am, due to their respective instructors’ unavailabilities and due to the order, the algorithm treats the courses, there were no feasible consecutive time slots to accommodate these courses in the schedule.

An observation that can be made is that the decrease in instructors’ availabilities increases the risk of having unscheduled courses

Nevertheless, the algorithm includes a loop that sets the time slots before 9 am and after 18 am as available, except for the days some courses cannot be given due to another course that should be given on another day. Moreover, this loop tries to schedule the courses present in the Unscheduled Courses Vector and assign them classrooms.

If no feasible time slots match the courses and the classroom availabilities, some courses are moved to other timeslots to make the adequate classroom available. However, this last part was not necessary as the algorithm found feasible time slots that matched the courses and the classroom availabilities.

Here are the results obtained:

<u>For the first bloc of business engineering (*fig. 59*):</u>

	Monday	Tuesday	Wednesday	Thursday	Friday
8h00-8h30	-	-	-	-	-
8h30-9h00	-	-	-	-	-
9h00-09h30	Mathématiques: Analyse infinitésimale - 1 - Valérie Henry - N1a 50 (0/50)	Physique générale : partim 1 - théorie 2 - Pierre Deneye - N1a 50 (0/50)	Mathématiques: Analyse infinitésimale - 2 - Valérie Henry - N1a 50 (0/50)	Anglais 1 - Céline Leroy - N1d 1/82	-
09h30-10h00					-
10h00-10h30					-
10h30-11h00					-
11h00-11h30	Espagnol 1 - Alexis Alvarez - N1d 1/82	Finance et comptabilité - Questions / Réponses - N1a 50 (0/50)	Finance et comptabilité - Remédiation - Wilfred Niessen - N1a 50 (0/50)		-
11h30-12h00					-
12h00-12h30	-	-	-	-	-
12h30-13h00	-	-	-	-	-
13h00-13h30	Mathématiques: Analyse infinitésimale - Remédiation - Valérie Henry - N1a 50 (0/50)	Economie politique - Microéconomie - Alain Jousten - N1a 30 (0/30)	Analyse sociale de l'économie et de l'entreprise - Annie Cornet - N1a 30 (0/30)	Maitrise de l'outil informatique - Eddy Heins - N1d 1/82	Finance et comptabilité - Wilfred Niessen - N1a 50 (0/50)
13h30-14h00					
14h00-14h30					
14h30-15h00					
15h00-15h30	Physique générale : partim 1 - théorie 1 - Pierre Deneye - N1a 50 (0/50)			Physique générale : partim 1 - TP 2 - Amandine Collignon - N1d 1/82	
15h30-16h00					
16h00-16h30		Economie politique - Microéconomie - TP - groupe 1 - Alain Jousten - N1d 0/86	Finance et comptabilité - TP - Wilfred Niessen - N1d 1/82	-	-
16h30-17h00				-	-
17h00-17h30	Economie politique - Microéconomie - Remédiation - Alain Jousten - N1a 30 (0/30)	Economie politique - Microéconomie - TP - groupe 2 - Alain Jousten - N1d 0/86	Physique générale : partim 1 - TP 1 - Amandine Collignon - N1d 1/82	-	-
17h30-18h00				-	-
18h00-18h30	-	-	-	-	-
18h30-19h00	-	-	-	-	-
19h00-19h30	-	-	-	-	-
19h30-20h00	-	-	-	-	-

<u>Figure 59. Schedule – C’. – Bloc 1 of Business Engineering – After second loop– *Results algorithm*</u>

For the second bloc of business engineering (*fig. 60*):

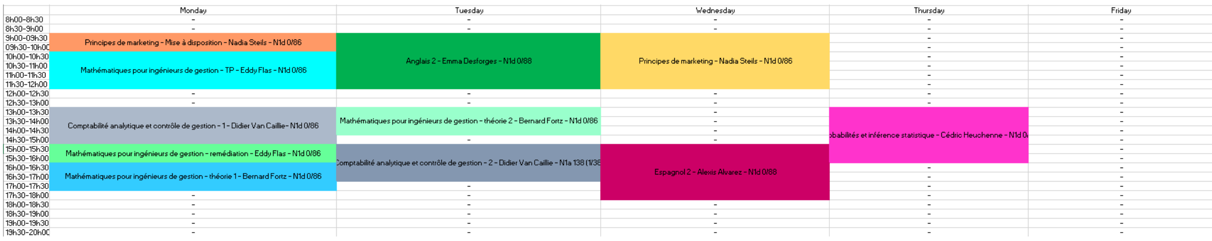


Figure 60. Schedule – C’. – Bloc 2 of Business Engineering – After second loop– *Results algorithm*

For the third bloc of business engineering (*fig. 61*):

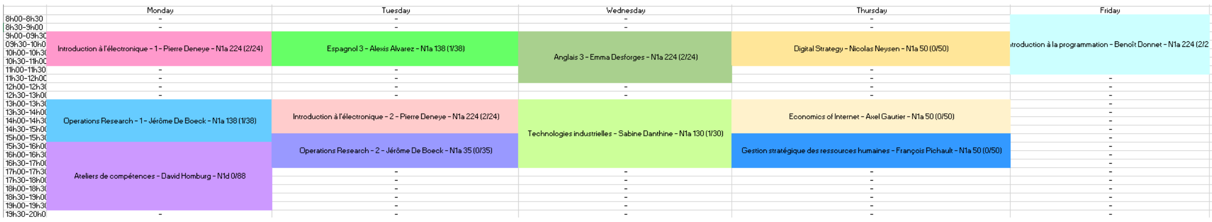


Figure 61. Schedule – C’. – Bloc 3 of Business Engineering – After second loop– *Results algorithm*

By comparing with the results before treating the courses in the Unscheduled Courses Vector, I observed that no courses had any classroom change or time slots change. The only change is the assignation of the courses “Ateliers de compétences” and “Introduction à la programmation” to time slots, day, and classroom.

After the full process of the algorithm, all courses of the instance have a day, a starting time slot, an ending time slot, and a classroom assigned and there are no courses in the Unscheduled Courses Vector.

By decreasing the number of available time slots per instructor, the number of unscheduled courses after the first main loop increased. The fewer instructors’ availabilities there are, the higher is the risk of having unscheduled courses. However, the second main loop was useful for scheduling these courses.

9.4. Limitations of the algorithm

One of the limitations of the algorithm is the fact that students are considered as groups and not as individuals, which means that, except for the course “Economie politique – Microéconomie – TP” which was split into 2 groups, the subgroups were not considered as they should, however, the number of students related to the courses were accurate. What could have been done is adding the subgroups and sub courses after the algorithm proceeded the instance to have the full schedule as I did for the tests on the third ILP model.

Another limitation instructors could work more than four days per week as it was not specified anywhere in the algorithm that instructors needed to work less than or equal to four days, unlike for the second and the third ILP models which can result in a non-optimal workload distribution within the schedules.

As already observed for the results given by the tests made on the ILP, each time I would get the results from the test of an instance, I had to interpret the results by myself to incorporate them in a schedule on Excel. The human intervention is still needed to interpret the results.

Finally, courses are sorted by the number of availabilities of their instructor and then by student groups. However, this can increase the number of unscheduled courses. If the courses that have the least availabilities have a smaller duration than courses having higher availabilities, it means that these courses could be placed at time slots that would prevent courses with a higher duration from finding consecutive feasible time slots. Nevertheless, it was a choice to sort the courses this way and not by duration.

10. Conclusions

This master thesis aimed at answering the question "How can mathematical models and a scheduling algorithm be effectively applied to design course schedules that minimize conflicts while respecting constraints in the Faculty of HEC?"

To answer this question, I went through a full process of collecting information, designing solutions, and analysing results. Firstly, I collected information about what has already been done in the class scheduling problem treatment by other researchers by writing a literature review. This way, I gained knowledge about how they implemented solutions in other schools and what data they needed to perform their implementation.

Then, I interviewed HEC's program manager, Ms. Boxus to understand how she proceeds to design class timetables and what kind of constraints she considers. I also collected quantitative data to test the models and the algorithm I wanted to implement. The idea was to use mathematical models and an algorithm to solve the class scheduling optimization problem in the HEC case. I decided to create integer linear programming models and design a First Fit heuristic.

I started by designing the ILP models incrementally. I created three of them, from the most basic one to the most complicated one. The idea was to implement more constraints every time I designed a new model. The first ILP model was the simplest one, considering only the non-negotiable constraints. Then, the second one introduced instructors' availability, the constraint preventing some courses from being given on the same day to increase the diversity of courses given on a day for students, and the one preventing instructors from working more than four days per week. Lastly, the third model considered students as individuals and allowed course conflicts by taking into account the "Décret Paysage". Generally speaking, the more advanced ILP models treat some of the limitations of their previous model.

I also designed a First Fit Heuristic that iterates through the list of courses sorted by capacity and then by student groups to find feasible time slots when the instructor, the student group, and the classroom are available. It also iterates through each unscheduled course to find feasible time slots by adding earlier or later time slots throughout the day or by changing time slots assigned to some courses to make the classroom attributed free for the unscheduled course.

Then, I tested the mathematical models and the heuristic with adapted instances. Each of them has been tested at least by three instances, the first one considering only the first bloc of business engineering, the second one taking into account the second bloc of business engineering as well and the third one considering every bachelor bloc of business engineering.

On the second mathematical model and the algorithm, I also tested the impact of removing additional instructors' availabilities.

By comparing the results obtained from testing the instances on the ILP models, I observed that the higher is the instance, the higher is the time required for the ILP to be optimized. Moreover, this time also increases by removing instructors' availabilities and by adding constraints.

Another observation made is that the diversity of classrooms used increases with the number of courses as the same classrooms cannot be available for many courses at the same time, other ones need to be used.

In the ILP models, the classrooms assigned to courses switched easily from one instance to another, while in the First Fit heuristic, the classrooms assigned to a bloc in the previous instance remained the same in the next instance, provided that the instructors' availabilities remained the same. In other words, the courses from the first and second blocs of business engineer kept the same classrooms assigned for every instance where the instructors' availabilities were the same and they were considered in. This is not the case for the third bloc of business engineering and this bloc was only considered in the third instance. However, it has been observed that changing instructors' availabilities provoke classroom changes for the courses.

By removing instructors' availabilities, the risk of not finding a solution in the case of the third and second ILP models and having unscheduled courses in the case of the First Fit heuristic increases.

10.1. Limitations

Throughout the process of implementing the mathematical models and scheduling algorithm, I noticed that several limitations remained :

1. The third ILP model faced a challenge concerning large instances. It was not possible to get any results while testing instances with more than 12 students and 44 courses, even after waiting for over 30 minutes. The biggest instance that gave results was one composed of 5 students and 42 courses which highlights scalability issues of the model.

2. Both ILP models and the heuristic algorithm required human intervention to interpret the results and incorporate them into a summary timetable. Moreover, the integration of sub-courses also needed to be addressed by a human due to the scalability issue mentioned before, otherwise, there would have been too many courses to be treated.

3. The First Fit heuristic does not consider the maximum number of four working days per week for instructors unlike the second and the third ILP models, which can result in a non-optimal workload distribution within the schedules.

4. Another limitation was the lack of real-life instructors' availabilities data. Due to confidentiality reasons, I was not able to be provided with this list. The same case was encountered with students' enrolment in classes. Even though I had the data specifying how many students were registered to courses, I did not have the precise registration to courses per student.

10.2. Recommendations

To address these limitations and improve the scheduling process, I would give the following recommendations:

1. Improving the scalability of ILP models to be able to process real-sized instances. To do so, a more powerful computer could be used, which means one has more performant processes and more RAM (random-access memory) which provides temporary storage on the machine to have direct access to it.

2. Automating results' interpretation by developing tools that could automatically interpret the results given by the code and transform them into a schedule without requiring human intervention, this would increase the efficiency of the scheduling process and be less time-consuming.

3. Refining the heuristic algorithm to consider students as individuals instead of student groups to consider the “Décret Paysage” that allows students to be enrolled in courses from different student groups.

4. Integrating the limit of four days of instructors’ workload within the First Fit algorithm to make it more accurate to HEC real-life case.

5. Experimenting with different sorting methods for the course list used in the First Fit heuristic to analyse if it would decrease the number of unscheduled courses. For example, the courses could be sorted by duration from the highest to the lowest instead of being tested by their respective instructors’ availabilities so that the courses having the highest duration would be scheduled first.

6. Conducting new tests with another variety of courses and classes, for example, instead of considering the business engineering, considering blocs of economics and management science. Or instead of considering the bachelor’s students, considering the master’s students and the diversity of the master programs existing.

7. Incorporating 15-minute breaks between courses would make the models closer to reality as Ms. Boxus tries to take these breaks into account while designing class schedules to allow students to move from one classroom to another outside of the courses’ time slots.

8. Testing the models and the heuristic with real instructors’ availabilities and course enrolments per student by someone allowed to use these data could be a way of ensuring the accuracy of the models.

9. Minimizing students’ trips between buildings on the same day as Ms. Boxus tries to minimize the number of trips students’ groups have to make for ecological reasons.

10.3. Future research

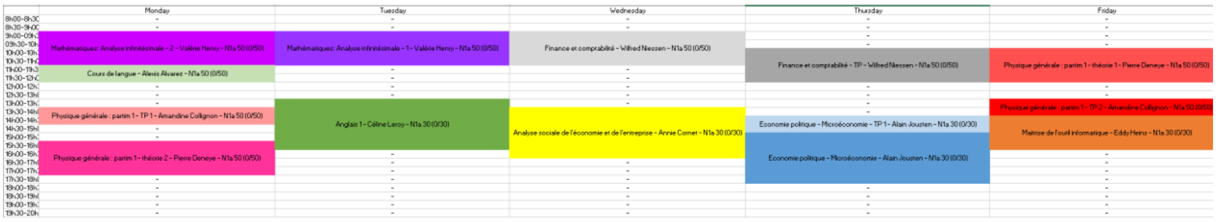
By addressing these limitations and implementing these recommendations, future research could build upon the findings of this thesis to develop even more robust and efficient solutions for solving the class scheduling problem at HEC.

A potential future research question arising from this work could be: “How can algorithms be designed to create personalised class schedules for individual students by incorporating the “Décret Paysage”, and what impact do these personalized schedules have on conflict minimization ?”

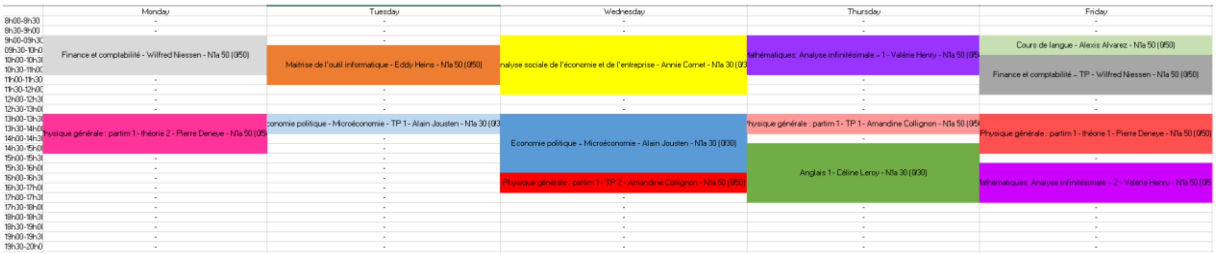
This question aims at testing the limits of schedule personalization and analyse their influence on course conflicts encountered by students.

11. Appendices

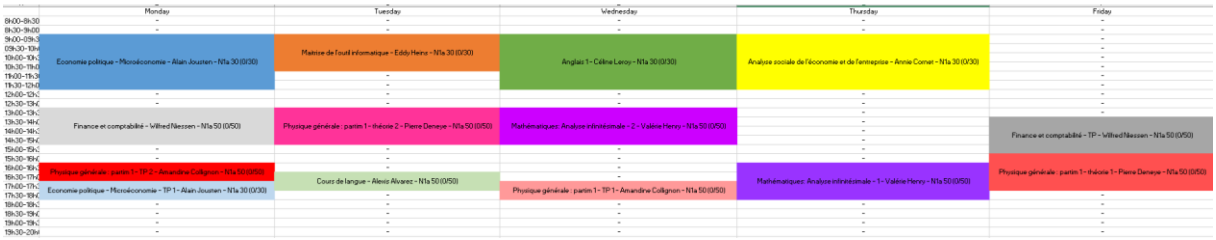
Appendix A. First tests “*Information_needed – 2.A’’.*” of the second ILP model – second instance with less availabilities – *Results V2*



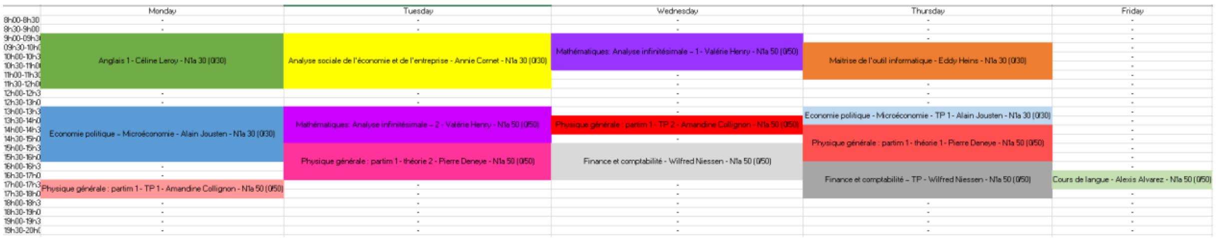
Appendix B. First tests “*Information_needed – 2.A’’’.*” of the second ILP model – third instance with less availabilities – *Results V2*



Appendix C. First tests “*Information_needed – 2.A’’’’.*” of the second ILP model – fourth instance with less availabilities – *Results V2*



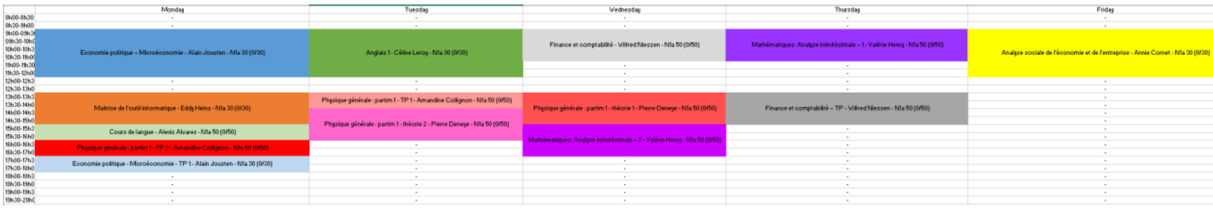
Appendix D. First tests “*Information_needed – 2.A’’’’’.*” of the second ILP model – fifth instance with less availabilities – *Results V2*



Appendix E. First tests “*Information_needed – 2.A’’’’’’.*” of the second ILP model – sixth instance with less availabilities – *Results V2*

Appendix F. Second tests “*Information_needed – 2.B′′.*” of the second ILP model – second instance with less availabilities – *Results V2*

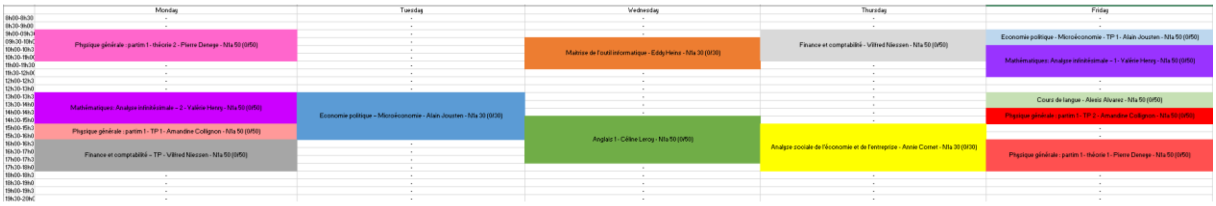
For the first bloc of business engineering:



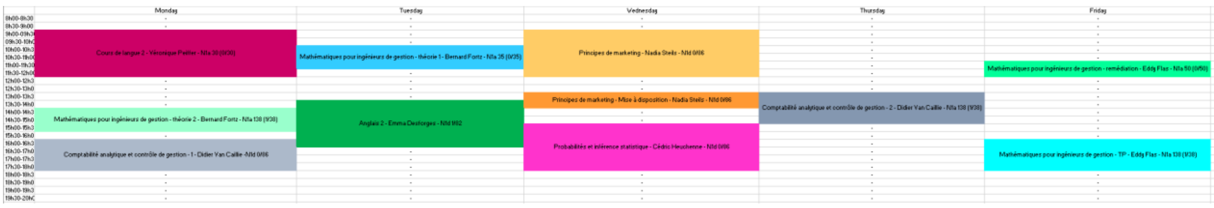
For the second bloc of business engineering:

Appendix G. Second tests “*Information_needed – 2.B′′′.*” of the second ILP model – third instance with less availabilities – *Results V2*

For the first bloc of business engineering:

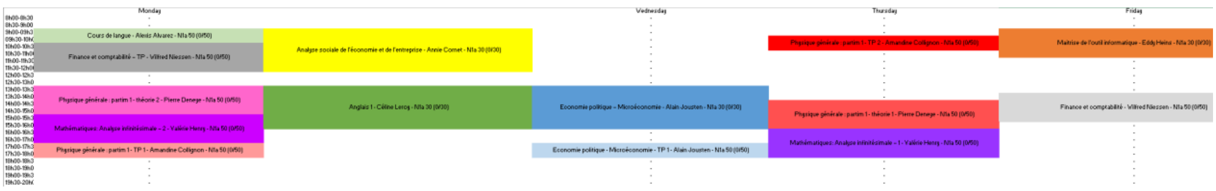


For the second bloc of business engineering:



Appendix H. Second tests “*Information_needed – 2.B′′′′.*” of the Second ILP model– fourth instance with less availabilities – *Results V2*

For the first bloc of business engineering:



For the second bloc of business engineering:

	Monday	Tuesday	Wednesday	Thursday	Friday
8h00-8h30	-	-	-	-	-
8h30-9h00	-	-	-	-	-
9h00-09h3	robabilités et inférence statistique - Cédric Heuchenne - N1d 0/86	Anglais 2 - Emma Desforges - N1d 1/82	-	-	-
09h30-10h			Mathématiques pour ingénieurs de gestion - théorie 2 - Bernard Fortz - N1a 138 (1/38)	-	-
10h00-10h3				-	-
10h30-11h0				-	-
11h00-11h30			-	-	-
11h30-12h0			-	-	-
12h00-12h3	-	-	-	-	-
12h30-13h0	-	-	-	-	-
13h00-13h3	Cours de langue 2 - Véronique Pfeiffer - N1a 30 (0/30)	-	-	-	-
13h30-14h0		Mathématiques pour ingénieurs de gestion - remédiation - Eddy Flas - N1a 138 (1/38)	-	-	-
14h00-14h3			Principes de marketing - Nadia Steils - F1 Grand auditoire B01 (0/2)	Mathématiques pour ingénieurs de gestion - théorie 1 - Bernard Fortz - N1a 138 (1/38)	Mathématiques pour ingénieurs de gestion - TP - Eddy Flas - N1a 138 (1/38)
14h30-15h0		-			
15h00-15h3		Comptabilité analytique et contrôle de gestion - 2 - Didier Van Caillie - N1d 0/86			
15h30-16h0				-	
16h00-16h3	-			-	Comptabilité analytique et contrôle de gestion - 1 - Didier Van Caillie - N1d 0/86
16h30-17h0	-			-	
17h00-17h3	-	-	Principes de marketing - Mise à disposition - Nadia Steils - N1d 0/86	-	
17h30-18h0	-	-		-	
18h00-18h3	-	-	-	-	-
18h30-19h0	-	-	-	-	-
19h00-19h3	-	-	-	-	-
19h30-20h	-	-	-	-	-

12. List of resource persons

Ms. Célia Paquay, my supervisor and teacher at HEC.

Ms. Marie Baratto, my thesis reader and teacher at HEC.

Ms. Marie-Gabrielle Boxus, the program master at HEC.

Ms. Fabienne Fontaine, data architect and data analyst of the operational excellence program at HEC.

13. Bibliography and references

Appleby, J. S. (1961). Techniques for Producing School Timetables on a Computer and their Application to other Scheduling Problems. *Computer Journal*, *3*(4), 237–245. https://doi.org/10.1093/comjnl/3.4.237

Aubin, J., & Ferland, J. A. (1989). A large scale timetabling problem. *Computers & Operations Research*, *16*(1), 67–77. https://doi.org/10.1016/0305-0548(89)90053-1

Bachelier en ingénieur de gestion. (n.d.). https://www.programmes.uliege.be/cocoon/20232024/programmes/G1IGES01_C.html

Barraclough, E. D. (1965). The application of a digital computer to the construction of timetables. *Computer Journal*, *8*(2), 136–146. https://doi.org/10.1093/comjnl/8.2.136

Chen, R., & Shih, H. (2013). Solving University Course Timetabling Problems Using Constriction Particle Swarm Optimization with Local Search. *Algorithms*, *6*(2), 227–244. https://doi.org/10.3390/a6020227

Chen, Y., Bayanati, M., Ebrahimi, M., & Khalijian, S. (2022). A Novel Optimization Approach for Educational Class Scheduling with considering the Students and Teachers' Preferences. *Discrete Dynamics in Nature and Society*, *2022*, 1–11. https://doi.org/10.1155/2022/5505631

Day, C., Kington, A., Stobart, G., & Sammons, P. (2006). The personal and professional selves of teachers: stable and unstable identities. *British Educational Research Journal*, *32*(4), 601–616. https://doi.org/10.1080/01411920600775316

Delsaux, J. (2023). *Information_needed - 3.A - 12 students* (Excel) [Dataset].

Delsaux, J. (2024a). *AlgorithmA* (Excel) [Dataset].

Delsaux, J. (2024b). *AlgorithmB* (Excel) [Dataset].

Delsaux, J. (2024c). *AlgorithmC* (Excel) [Dataset].

Delsaux, J. (2024d). *AlgorithmC'* (Excel) [Dataset].

Delsaux, J. (2024e). *Information_needed - 1.A.* (Excel) [Dataset].

Delsaux, J. (2024f). *Information_needed - 1.A'.* (Excel) [Dataset].

Delsaux, J. (2024g). *Information_needed - 1.B.* (Excel) [Dataset].

Delsaux, J. (2024h). *Information_needed - 1.C.* (Excel) [Dataset].

Delsaux, J. (2024i). *Information_needed - 2.A.* (Excel) [Dataset].

Delsaux, J. (2024j). *Information_needed - 2.A’.* (Excel) [Dataset].

Delsaux, J. (2024k). *Information_needed - 2.A’’.* (Excel) [Dataset].

Delsaux, J. (2024l). *Information_needed - 2.A’’’.* (Excel) [Dataset].

Delsaux, J. (2024m). *Information_needed - 2.A’’’’.* (Excel) [Dataset].

Delsaux, J. (2024n). *Information_needed - 2.A’’’’’.* (Excel) [Dataset].

Delsaux, J. (2024o). *Information_needed - 2.A’’’’’’.* (Excel) [Dataset].

Delsaux, J. (2024p). *Information_needed - 2.A’’’’’’’.* (Excel) [Dataset].

Delsaux, J. (2024q). *Information_needed - 2.B.* (Excel) [Dataset].

Delsaux, J. (2024r). *Information_needed - 2.B’.* (Excel) [Dataset].

Delsaux, J. (2024s). *Information_needed - 2.B’’.* (Excel) [Dataset].

Delsaux, J. (2024t). *Information_needed - 2.B’’’.* (Excel) [Dataset].

Delsaux, J. (2024u). *Information_needed - 2.B’’’’.* (Excel) [Dataset].

Delsaux, J. (2024v). *Information_needed - 2.C.* (Excel) [Dataset].

Delsaux, J. (2024w). *Information_needed - 2.C’.* (Excel) [Dataset].

Delsaux, J. (2024x). *Information_needed - 2.C’’.* (Excel) [Dataset].

Delsaux, J. (2024y). *Information_needed - 3.A.* (Excel) [Dataset].

Delsaux, J. (2024z). *Information_needed - 3.B.* (Excel) [Dataset].

Delsaux, J. (2024aa). *Information_needed - 3.C.* (Excel) [Dataset].

Delsaux, J. (2024ab). *Information_needed - 243 students* (Excel) [Dataset].

Delsaux, J. (2024ac). *Results algorithm* (Excel) [Dataset].

Delsaux, J. (2024ad). *Results V1* (Excel) [Dataset].

Delsaux, J. (2024ae). *Results V2* (Excel) [Dataset].

Delsaux, J. (2024af). *Results V3* (Excel) [Dataset].

Dinkel, J. J., Mote, J., & Venkataramanan, M. A. (1989). OR Practice—An efficient decision support system for academic course scheduling. *Operations Research*, *37*(6), 853–864. https://doi.org/10.1287/opre.37.6.853

Even, S., Itai, A., & Shamir, A. (1976). On the Complexity of Timetable and Multicommodity Flow Problems. *SIAM Journal on Computing*, *5*(4), 691–703. https://doi.org/10.1137/0205048

Fontaine, F. (2023). *Course bac HEC 2023-2024* (Excel) [Dataset].

Hertz, A. (1991). Tabu search for large scale timetabling problems. *European Journal of Operational Research*, *54*(1), 39–47. https://doi.org/10.1016/0377-2217(91)90321-l

Mooney, E. L., Rardin, R. L., & Parmenter, W. (1996). Large-scale classroom scheduling. *IIE Transactions*, *28*(5), 369–378. https://doi.org/10.1080/07408179608966284

Mulvey, J. M. (1982). A classroom/time assignment model. *European Journal of Operational Research*, *9*(1), 64–70. https://doi.org/10.1016/0377-2217(82)90012-1

Murphy, J., & Sutter, R. (1964). School Scheduling by Computer, the Story of GASP. *Educational Facilities Labs.* http://files.eric.ed.gov/fulltext/ED031066.pdf

Pinedo, M. L. (2009). Interval scheduling, reservations, and timetabling. In *Springer eBooks* (pp. 207–230). https://doi.org/10.1007/978-1-4419-0910-7_9

Samiuddin, J., & Haq, M. A. (2019). A novel two-stage optimization scheme for solving university class scheduling problem using binary integer linear programming. *Operations Management Research/Operations Management Research : Advancing Practice Through Research*, *12*(3–4), 173–181. https://doi.org/10.1007/s12063-019-00146-8

Silva, L., & Dario, J. (2003). *Metaheuristic and multiobjective approaches for space allocation*. http://eprints.nottingham.ac.uk/10147/

Tripathy, A. (1984). School Timetabling—A case in large binary integer linear programming. *Management Science*, *30*(12), 1473–1489. https://doi.org/10.1287/mnsc.30.12.1473

Université de Liège. (2020). *Liste des salles*.

Wasfy, A., & Aloul, F. (n.d.). *Solving the university class scheduling problem using advanced ILP techniques*.

Yazdani, M., Naderi, B., & Zeinali, E. (2017). Algorithms for university course scheduling problems. *Tehnički Vjesnik/TehničKi Vjesnik*, *24*(Supplement 2). https://doi.org/10.17559/tv-20130918133247

Executive Summary

This master thesis aimed to address the research question: "How can mathematical models and a scheduling algorithm be effectively applied to design course schedules that minimize conflicts while respecting constraints in the Faculty of HEC?" To answer this question, a process of information collection, solution design, solution testing, and result analysis was undertaken.

Firstly, a literature review was conducted to understand existing approaches to the class scheduling problem. This provided identification of solutions implemented in other educational institutions and ideas of the data required to have a close to the real-life case solution.

Afterward, an interview with Ms. Boxus, HEC's program manager, was conducted to collect information about the current scheduling process at the school and the main constraints and objectives encountered while designing the timetables. Accurate quantitative data was also collected to test the models and the algorithm.

Three integer linear programming (ILP) models were developed incrementally, each involving more constraints than the last. The simplest method contains only the non-negotiable constraints. The second one treats instructors' availability, workload balance, and courses that should be given on different days. The third one includes the consequences of the "Décret Paysage" and considers students as individuals to allow potential conflicting schedules. Additionally, a First Fit heuristic was designed to find feasible time slots for courses based on room capacity, and instructors', student groups', and classrooms' availabilities.

The models and the algorithm were tested using various instances, from the first one considering only the first bloc of business engineering to the last one considering every bachelor bloc of business engineering.

The results showed that the time required to optimize the models increased with the size of the instance, the removal of instructors' availabilities, and the number of constraints in the model. The diversity of classrooms used also increased with the number of courses.

Several limitations were identified during the research such as human intervention that was still needed to interpret the test results to design a readable schedule, some scalability issues has been faced for large instances, and the fact that the heuristic considers students as groups rather than individuals.

To address these limitations, recommendations were made, including solutions to improve the scalability of ILP models, more automated result interpretation, and the creation of a heuristic to consider students as individuals.

A potential future research question analyse the impact and limits of incorporating the "Décret Paysage" in algorithms to design more personalised schedules for individual students.

Word count = 26.742