|  |
| --- |
| IÉSEG School of management – Optimization techniques |
| Group Project |
| |  | | --- | | ABDALLAH Hytham  DIAZ Fernando  RAMIREZ Luisa  DESHPANDE Siddharth |   IÉSEG - Ecole de Commerce à Paris et Lille : Bachelor, Masters, MSc |

Table of Contents

[Table of Contents 1](#_Toc138020953)

[Introduction 2](#_Toc138020954)

[Concept explanation 2](#_Toc138020955)

[Integer Linear Programming 2](#_Toc138020956)

[Simplex method 2](#_Toc138020957)

[Planning problems 3](#_Toc138020958)

[Problem definition 3](#_Toc138020959)

[Modelling 3](#_Toc138020960)

[Defining the Sets 3](#_Toc138020961)

[Parameters 3](#_Toc138020962)

[Variables 4](#_Toc138020963)

[Yij 4](#_Toc138020964)

[Chj 4](#_Toc138020965)

[B 4](#_Toc138020966)

[Objective function 4](#_Toc138020967)

[Constraints 4](#_Toc138020968)

[Results 5](#_Toc138020969)

[Conclusions 7](#_Toc138020970)

[Further study 8](#_Toc138020971)

[Bibliography 8](#_Toc138020972)

# Introduction

Efficient task assignment and scheduling are crucial components of successful project management. This report addresses the challenge of optimizing task allocation and scheduling in a project setting. By formulating the problem as an integer linear programming (ILP) model, we aim to find the optimal solution that minimizes total time while considering various constraints. Optimization problems, as the one proposed in this report, aim to find solutions which are optimal to satisfy an objective while following a process that allows one to solve it. In this sense, this report shows the development, results and concepts of a project scheduling problem, using software like AMPL and CPLEX.

# Concept explanation

## Integer Linear Programming

Linear Programming is a mathematical technique that aims to find the best outcome in each mathematical model, for a list of requirements in a mathematical model. In other words, Linear programming is a technique for the optimization of a linear objective function, subject to linear equality and linear inequality constraints. In this way a linear programming problem must meet one of the following conditions:

1. the problem is unbounded,
2. ii) the problem is infeasible,
3. (iii) the problem has an optimal solution (Mahto, 2012)

This is known as the fundamental theorem of Linear Programming and it ensures that every linear programming problem has at least one feasible solution, and if an optimal solution exists, it will be found at one of the corners or vertices of the feasible region. This means that the linear programming method searches through the polytope's vertices to find the point that minimizes or maximizes the objective function. Thus, an LP problem requires to define a linear objective function, constraints in the availability of the resources and an alternative course of action to select the best option (Mahto, 2012).

Within linear programming we find a subcategory which is linear integer programming, which refers to the problems where the variables are required to take on integer values and where all the functions are linear (Kolman & Beck, 1995). This integer requirement introduces a new level of complexity and consideration to the optimization process. Unlike in linear programming, where variables can be any real number within a given range, linear integer programming restricts the feasible solutions to discrete values. This discrete nature of the variables reflects real-world situations where decisions must be made in whole units or quantities. One example of this is planning problems, where the aim is to find the optimal resource allocation for a certain project.

## Simplex method

The simplex method is frequently the most efficient method of solving linear programming (LP) problems, and it involves finding the best solution for a linear objective function while satisfying a set of linear constraints. The simplex method explores different points within a polytope, which represents the feasible solutions of the problem. It starts at one vertex of the polytope and systematically moves to adjacent vertices, checking if the objective function improves or remains the same at each step. It typically takes a number of iterations that is proportional to the number of equality constraints in the problem (Carreira-Perpiñán, 2023)

## Planning problems

One of the most common type of problems that are attempted to be solved through optimization are project scheduling problems (PSP), a type of planning problem that aims to optimize the profit of a project, the time, or the cost, based on limited resources and under specific constraints. This type of problem is especially relevant, because of them depends on the success of a project. According to (Fahmy, 2016), scheduling has become one of the most research topics withing the operational research context, as it has many practical applications within multiple industries. However, it is also one of the most challenging problems (Fahmy, 2016)

The basic form of the PSP aims to minimize the time of a project which has finish to start logic relations and that does not include lags. Thus, the most common approach in literature is the static scheduling, a deterministic approach, which consists in identifying how and when the activities of the schedule should be executed (Fahmy, 2016)

# Problem definition

For the purpose of this project, we will try to make a simple project scheduling problem in order to minimize the amount of time. We will have resources (contractors) which are able to perform all of the 10 tasks, we will have a budget constraint, also we will add a constraint of non-consecutive tasks. This last one means that a contractor cannot perform two tasks consequently, since the company wants the risk of the project to spread between several contractors and not depend only on one. In summary each of the contractors will have a cost and time for each of the activities, no contractor can do two consecutive activities, the cost of the activities should not surpass the budget and total time of the project should be minimized. Most project scheduling problems have parallel activities, nevertheless this is not included in the scope of the project.

# Modelling

We move on to the modelling where we define the model, which consists of 5 main steps -

1. Defining the Sets
2. Parameters
3. Variables
4. Objective function
5. Constraints

## Defining the Sets

As mentioned in the previous problem definition, our data mainly comes from a set of tasks to be completed defined by J, and a set of contractors who will be assigned a task to complete based on the various constraints, taken from the set C.

## Parameters

To provide a set of values which would enable the model to evaluate and come to an optimal solution, we would have to define the parameters of cost and time. The parameter P, is the price or cost of a contractor for a particular task and the parameter t is the amount of time it takes for him/her to complete a task. In an ideal world each contractor would have different efficiencies and would similarly value how much a task is worth differently.

## Variables

For this problem we will use three variables which are defined below

### Yij

Variable Yij represents a binary integer matrix where the assignation of the tasks will be done. If task i is assigned to contractor j then Yij will be equal to 0, on the other hand if the task i is not assigned to contractor j then Yij is equal to 1. As we will see later on the constraints, only one of the task can be assigned to 1 contractor.

### Chj

This variable will be the cumulative time for each of the tasks. In other words, Ch1 will be the duration of task 1 according to the contractor to whom the activity was assigned. Nevertheless, Ch2 will the sum of task 1 and task 2 according to the contractor(s) to whom the activity was assigned. This variable will be always equal to 0 (there cannot be negative time) and it will be an integer.

### B

B refers to the budget that will be used in the project. We defined this variable in order to know how much of the budget was used after simplex does the assignation of tasks. This variable will be defined as integer and should be greater or equal to 0.

## Objective function

The entire purpose of our problem is to minimize the completion time taken for the entire building project. The variable “Ch” serves the purpose of being the cumulative completion time of all the tasks done till that point in time. Thus, our objective function involves minimizing only the last completion time (Ch[10]), which is the completion time after the 10th task has been completed.

## Constraints

This is one of the most important aspects of the entire model as it ensures that it is feasible, allows us to model real world limitations and assist us in achieving very specific goals and objectives, which will be explained in further detail.

*“subject to AssignTask {j in J}:*

*sum {i in C} y[i, j] = 1;”*

The purpose of this constraint is to ensure that each task is assigned one contractor. It postulates that for every j in the set of tasks, the sum of the binary variable y (whether a contractor has been given a task or not) for all contractors should be equal to 1.

*“subject to MaxTasksPerContractor {i in C}:*

*sum {j in J} y[i, j] <= 10; “*

The constraint above makes sure that a contractor cannot be assigned more than 10 tasks. It basically is used to restrict the sum of the y of a contractor for each task to be within the limit.

*subject to BudgetConstraint:*

*sum {i in C, j in J} y[i, j] \* P[i, j] = B;*

*“subject to BudgetValue:*

*B<=550;”*

The Budget constraint has been divided into two parts to simplify the structure of the model. This constraint ensures that the entire project does not exceed the amount of monetary resources available which is similar in real world scenarios. It states that the sum of every price P multiplied by y(whether the task has been assigned to the contractor or not) should be less than the allocated budget.

“*subject to TimeConstraint {j in 2..10}:*

*Ch[j] = sum {i in C} y[i, j] \* t[i, j] + Ch[j-1]; “*

One of the primary constraints aims to cumulate the time taken for a particular task based on the order of the task. For every j in the set of tasks from the second task till the end, the completion time of a task will be equal to the sum of the time taken to complete by the contractor chosen (y\*t) along with the time of completion of the previous task.

*“subject to First\_Ch:*

*Ch[1] = sum {i in C} y[i, 1] \* t[i, 1];*

*subject to Last\_Ch:*

*Ch[10] >= sum {i in C, j in J} y[i, j] \* t[i, j] ; “*

Both these constraint equations make sure that the cumulative time for the first one is defined, and the cumulative time taken of the last task is always greater than the rest. As we observed in the previous constraint, the completion time of the first task was not defined, hence we have taken the sum of every i in the set of contractors for first task, such that y(binary variable defining if contract is established per contractor) multiplied by time taken by that contractor.

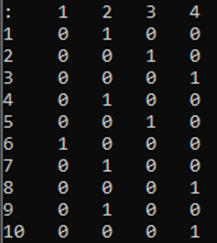
*“subject to non\_consecutive\_tasks {i in C, j in 2..10}:*

*y[i, j] + y[i, j - 1] <= 1;”*

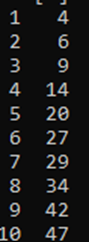
Finally, the last constraint, ensures that no contractor is assigned consecutive tasks. This is because we have tried to compare it to real life situations where a contractors efficiency decreases as he works a lot. So we have tried to create buffers in between for each contractor so that he/she is not overwhelmed by the entire project. In this equation, the sum of the binary variable y for a given task( assigning contractor to task - yes/no) along with the y of the task before is less than equal to 1.

# Results

Our model was able to assign the tasks over the 4 contractors. As you can see below, contractor 1 completed only task 6, while contractor 2 did four different tasks 1,4,7, and 9. On the other side 3rd contractor handled tasks 2 and 3 and finally the 4th contractor handled tasks 3 8 and 10.



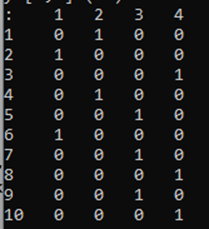
We were also able to calculate the overall time needed for this project (47 days) and the cumulative time taken with each task.



And what is most important that all these tasks were done without exceeding the budget 600 we specified.

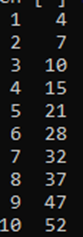


Meanwhile, if our budget decreases to 550 we can see a change in tasks



As tasks 2 is completed by contactor 1 instead of contractor 3, same for task 7 it was completed by contractor 4 instead of 3, and task 9 wad handeled to contractor 3 instad of 1.

This certainly will cause and increase in the overall completion time of the tasks as some contractors will have to delay and could need more time to finish a task. And we can see that the model gave us an increase of 5 days.



With a lower overall cost

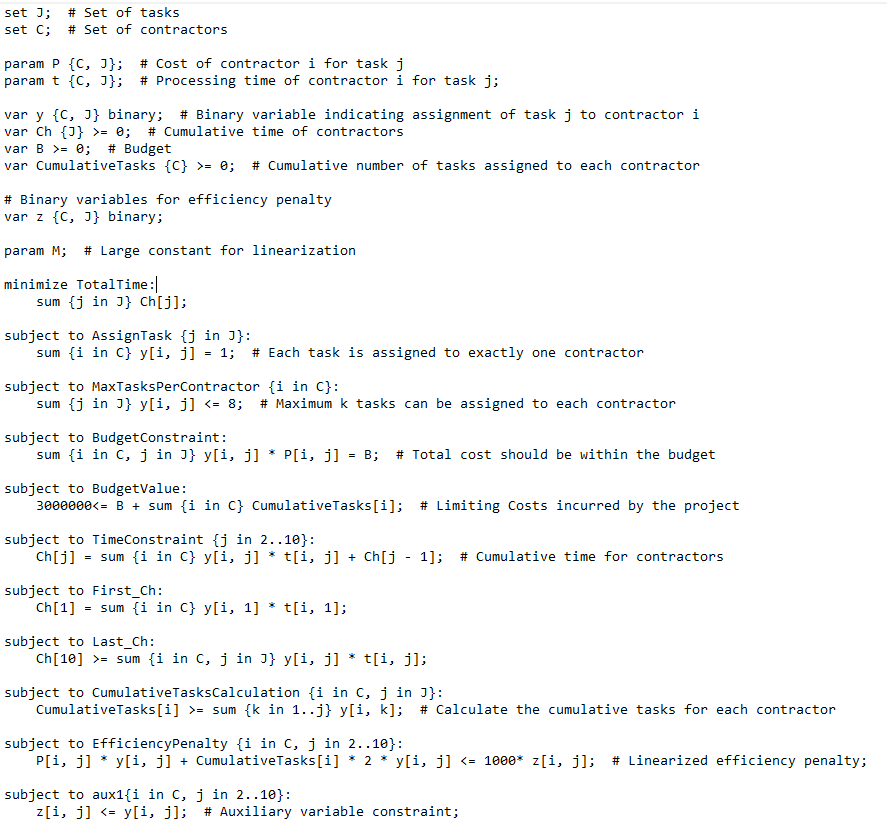


# Conclusions

Like a real-world situation we had resource and time constraints which could be a complicated and a difficult task to do in project management. Our model was able to assign different contractors to different tasks based on the cost parameters of each contractor for doing a task which assures us that our model didn’t assign contractors simultaneously. And as we saw in the results sections that when we decreased the budget to 550, different tasks were assigned to different contractors causing an increase in completion time of the project, which can conclude that the higher the budget the faster completion time of the project.

# Further study

Another aspect we wanted to explore was the introduction of an efficiency penalty constraint which is dependent on whether a contractor has done the previous task and so the problem becomes more complex and non-linear. We have attempted to solve this issue and linearize but it could not work, and it is an aspect worth discovering further as it will drastically make the model more realistic and complex. We have provided a snapshot of the code we attempted to linearize it. It requires further study and investigation.



# Bibliography

Carreira-Perpiñán, M. A. (2023). *Simplex method.* Retrieved from Wolfram Mathworld: https://mathworld.wolfram.com/SimplexMethod.html

Fahmy, A. (2016). *Optimization Algorithms in Project Scheduling.* CHAP.

Kolman, B., & Beck, R. E. (1995). *Integer Programming Problem.* Retrieved from Science Direct: https://www.sciencedirect.com/topics/mathematics/integer-programming-problem

Mahto, D. (2012, 12 01). *Essentials of operations Research.* Retrieved from RsearchGate: https://www.researchgate.net/publication/297715752\_Linear\_Programming