Resource-constrained mitigation controller

Thesis

Faculty: Faculty of Civil Engineering and Geosciences
Department: Construction Management and Engineering

Student name: Solange Cipolla

Student number: 6084796

First TU supervisor: Ruud Binnekamp
Second TU supervisor: Hans Ramler
Third TU supervisor: Teun Janssen

Abstract

Delays and cost overruns are common problems in the construction industry. Despite extensive research on their causes and mitigations, these problems persist, suggesting that the challenge is not identifying possible mitigation measures, but rather selecting the optimal combination. Recently, a decision-support tool, Mit-C, was developed to help find the optimal strategy by combining a Monte Carlo simulation with mathematical optimization. However, the tool has a relevant limitation: it does not consider the resources required by the mitigation measures, potentially leading to unrealistic or infeasible solutions.

Consequently, this master thesis addresses this limitation by further developing the project management decision-support tool, Mit-C, through the inclusion of the resources availability and demand required by the mitigation measures. For this purpose, the addition of a significant number of variables and constraints into the original mathematical model was needed, increasing the computational time of the program but improving the realism of the results. The altered model was then validated using a case study: the construction of a warehouse. The tool was used with both simplified and detailed project data to test its performance.

The results demonstrated that including resource constraints has a significant effect on the optimal mitigation strategy and leads to a lower and more realistic probability of finishing the project on time. This difference is more noticeable when using detailed data. It is therefore concluded that while the resource-constrained model produces more pessimistic results, it offers a significantly more realistic, reliable and therefore valuable decision-making tool for project managers.

Table of Content

1. Introduction

The construction industry is known for having projects with delays and cost overruns. Several researchers investigate the causes of these overruns to propose mitigation measures. Moreover, they usually focus their studies on specific types of projects or countries to provide mitigation measures that work best in the given context. Besides all the efforts, there are still delays and cost overruns in the construction industry. According to Rivera et al. (2017), on average, 72% of construction projects exceed their target duration by 38%, while 63% exceed their budget by 24%.

Since generally research on causes of overruns and mitigation measures involve interviews to specialists in the industry, it is evident that project managers know how to mitigate risk, but perhaps the problem is how to choose which mitigation measure to implement. In other words, project managers know what they can do, but maybe they need assistance to help them decide what to do.

When a project is delayed, it is necessary to implement mitigation measures. As stated before, project managers must decide which measures to implement, and this choice is usually based on the impact that the measures have on the probability of finishing the project on time (Kammouh et al., 2021). The authors explain that, in practice, Monte Carlo scheduling is usually employed to evaluate the measures' effect on the project duration.

However, according to Kammouh et al. (2021), the problem with using the Monte Carlo Simulation (MCS) is that it does not take into account the "project manager goal-oriented control behavior". The authors state that the project manager always tries to optimize by implementing only the mitigation measures needed to finish on the target time. Thus, they propose a decision-support tool, Mit-C, that combines MCS with a mathematical optimization. In the latter one, the goal is to "find the set of mitigation measures that ensures timely completion at the least amount of costs" (Kammouh et al., 2021).

In the next sub-section, a short review of the different versions of the mitigation controller is given, while in section 2, a more in-depth explanation of the basic Mit-C will be provided. For more information regarding the subsequent models refer to Appendix A or the cited references.

1.1 Mitigation controller

As introduced earlier, Kammouh et al. (2021) developed a tool which helps project managers identify the optimal mitigation strategy to implement in a construction project. In general terms, the tool works by performing an optimization in each iteration of a MCS in which the project duration exceeds the target. The aim of the first version of the tool is to find the best cost-effective measures that help reduce delay in a project by implementing a multi-objective linear optimization.

Khalifé (2022) further developed the original tool by optimizing the project's budget instead of the duration. Moreover, the author incorporated a "multi-criteria assessment of negative impacts". In this case, the process for finding the best mitigation measures is very similar to the first model, the main changes are done in the optimization equations to reflect the different objective and in the definition of effectiveness which now incorporates the negative impact of the measures.

The original Mit-C only considers the paths with a delay and cannot consider the possible benefits of finishing the project before the target time. Consequently, Kammouh et al. (2022) further developed the initial model to include different types of contractual project completion performance schemes in which penalties and/or rewards for the project's duration are considered. In order to do this, the objective function was changed to find the mitigation strategy that minimizes the net cost which includes the cost of the mitigation measures and the penalties and/or rewards. Moreover, Kammouh et al. (2022) introduced in the new model the difference between the uncorrelated duration and the correlated duration of activities.

The model of Kammouh et al. (2022) was later further developed by Manoj Philip (2022). This new research focused on studying the influence of the project network structure on the mitigation controller. The main modification implemented by Manoj Philip (2022) is the use of the Graphical Evaluation and Review Technique (GERT) instead of the PERT. As explained by the authors, this technique allows to tackle the limitations of the PERT. For instance, while PERT follows a deterministic branching, GERT follows a probabilistic one. Also, while in the original approach the activities are executed in a linear way, the GERT allows to repeat activities by "feedback loops".

Further research was done by Teuber et al. (2024) based on Kammouh et al.(2022) model. The authors presented a new model called Open Design and Dynamic Control (Odycon), which integrates the MCS with the Integrative Maximization of Aggregated Preferences (IMAP) optimization method. With this approach the interests of all the stakeholders are taken into account. Also, considering that stakeholders may have more than one interest, Odycon allows the inclusion of several objectives, broadening the original mitigation controller tool which only focuses on the duration or the budget of the project.

As can be noticed, in the last few years some improvements have been made to the mitigation controller tool, however it still has some relevant limitations which will be discussed in the following sub-section (Sub-section 1.2).

1.2 Resource constraints

Up to now, no research regarding the mitigation controller has included the resources required by the mitigation measures. As explained in Kammouh et al. (2021), in the original model, this is because it is assumed that project managers are conscious of the available resources when including mitigation measures and, thus, if there are not enough resources for a specific measure, they will not be using that measure. In other words, resources are not incorporated in the initial model, but they are assumed to be taken into account by the users of the tool.

Moreover, Kammouh et al. (2022) mention resource availability as an example of shared uncertainty causes, when introducing the concept of correlated duration of activities. Here, the resource problem is acknowledged as a source of uncertainty that may cause delay, but it is not directly taken into account in the selection of the mitigation measures strategy. In broad terms, the tool tries to find the measures that reduce the project duration, and the resource availability is one of the possible factors that may affect the makespan. However, the tool does not consider that the measures demand specific resources to be implemented.

Furthermore, the project managers do not take into account that the tool may suggest using the same mitigation measure in different tasks occurring at the same time or different measures that require the same resources for activities happening in parallel. This can happen because the tool does not consider the resources availability and demand by the mitigation measures. As explained by Kammouh et al. (2022), assuming unlimited resources leads to boundless use of simultaneous measures and, thus, the authors suggest that further research should focus on the incorporation of resources limitation in the model.

As already stated, some mitigation measures share resources and implementing them at the same time may cause trouble. Moreover, some resources are renewable, such as workers and machinery, whereas others, like materials, are non-renewable and cannot be reused in a different mitigation measure or in cases where the same measure is applied to multiple activities. Consequently, there is a clear need to incorporate the resource availability and demand required by the mitigation measures into the mitigation controller model.

1.3 Development Statement and Goal

The aim of this master thesis is:

To further develop the project management decision-support tool called Mit-C by including the resources availability and demand required by the mitigation measures in a construction project.

The following sub-questions are relevant for the development:

- **What are the existing techniques or methods to tackle the resource-constrained problem?**
- **How can the resource constraints be incorporated into the existing mitigation controller model?**
- **What is the effect of including the resource constraints in the mitigation controller?**

The objective of this master thesis is to further develop the existing mitigation controller tool, Mit-C, by incorporating the availability of and demand for resources into the model. The previous developed code will be used as a starting point, and alterations will be guided by research on the Resource Constrained Project Scheduling Problem (RCPSP), since this body of literature provides methods for minimizing project duration while considering resource and precedence constraints, which aligns with the objective of adding resource constraints into Mit-C. Once the modifications are completed, possibly after some iterations, it will be tested on a finished construction project to evaluate its benefits. Furthermore, two interviews will be conducted with professionals from the industry, who participated in the case study as the project manager and the construction superintendent (site manager), to determine the value of the tool. Similarly, a comparison of the tool results with and without the resource constraints will be performed.

2. Mitigation Controller

In this section, a more detailed explanation of how the Mit-C model works is given. This description will focus on the versions of Kammouh et al. (2021) and Kammouh et al. (2022), since the inclusion of the constraints will be carried out on the latter model and this one is based on the first one. For an even more extensive explanation of the models refer to any of the previously mentioned sources.

This section is divided into three parts. The first one outlines the steps of the model, while the second one explains the mathematical optimization, and the last one summarizes the generated plots.

2.1 General workflow

The tool described in Kammouh et al. (2021) works as follows. First, the user needs to input all the required data. This includes information about the activities, risk events, mitigation measures, and the relationship between them. For the activities the most likely, pessimistic and optimistic values of their duration are needed as well as their precedence relationship. Likewise, the required input includes the three capacity estimates for each mitigation measure, the cost associated with the most likely capacity and its cost-capacity correlation factor, and the activities on which each measure can be implemented. Moreover, the user also needs to input the three estimates for the effect of each risk, their probability of occurrence and their relationship with the activities. In the model of Kammouh et al. (2022), in addition to the previous data, it is necessary to input the three estimates for the shared uncertainties and the activities they relate to. Other relevant inputs include the target duration of the project, and the daily penalties and rewards if permitted by the contractual project completion performance scheme. It is relevant to mention that if the target duration is not provided by the user, the tool calculates a target using the most-likely duration of the activities.

Then, when the program is run, a MCS starts with the number of iterations established by the user. As explained by Kammouh et al. (2021) the MCS method is used to include the stochastic behavior of the variables. In each iteration, random values of the durations of the activities (uncorrelated and correlated), the capacities of the mitigation measures and the disruption duration of the risks are obtained by using the BETA-PERT (Program Evaluation and Review Technique) distribution and the three-point estimates entered for each parameter. In the case of the Kammouh et al. (2022) model, it is also necessary to obtain a random binary value from a Bernoulli distribution which determines the occurrence of each risk, and the costs associated with the random values obtained for the capacities of the mitigation measures. Once all these values are gathered, the following step is to calculate the durations of all the critical paths. Here are considered as critical those paths with a duration larger than the target one.

In Kammouh et al. (2021), the effectiveness of a mitigation measure is described as the ratio between the delay reduction and the cost of the corrective measure. Therefore, in the original model, it is also necessary to compute the delay reduction of the paths by incorporating one mitigation measure.

The next step in each MCS iteration consists of executing the optimization algorithm and saving the results. After this, the iteration ends and a new one begins until the number of iterations entered by the user is reached. It must be highlighted that the optimization is carried out only when the calculated project duration exceeds the target, otherwise a new iteration starts without optimizing. Finally, when the MCS ends, the output of all the iterations are aggregated and displayed in different plots.

Table 1 summarizes the data and variables involved in the functioning of the mitigation controller.

Type	Symbol	Definition
Input data	I	Number of activities in the project (with activities indexed by i)
	J	Number of mitigation measures (with mitigation measures indexed by j)
	E	Number of potential risk events (with risk events indexed by e)
	F	Number of factors of shared uncertainty (with factors indexed by f)
	K	Number of project schedule paths (with paths indexed by k)
	p_e	Probability of occurrence of risk e
	$s_{i,e}$	Relation parameter that states if risk event e can impact activity i
	$r_{i,j}$	Relation parameter that states if mitigation measure j can intervene upon activity i
	$v_{i,f}$	Relation parameter that defines the extent to which an uncertainty factor f affects the activities i
	d_i	Duration of activity i
	m_j	Mitigation capacity of measure j
	c_j	Cost of mitigation measure j
	n_j	Cost-capacity correlation factor of mitigation measure j
	d_e	Disruption duration caused by risk event e
	$U_{i,f}$	Shared uncertainty of activity i by factor f
	T_{Tar}	Target project completion duration
	P	Daily penalty
	R	Daily reward
Extra values	$p_{k,i}$	Relation parameter that defines if activity i is in path k
	T_{curr} , d_k^0	Current completion time (completion time with no mitigations)
Variables	x_j	Binary variable that states if a mitigation measure j is implemented or not
	Δ_1	Delay after implementing the mitigation measures
	Δ_2	Duration reduction beyond the target duration of the project

Table 1: Mit-C data and variables

2.2 Objective function

In the Kammouh et al. (2021) model, the objective function is composed of three terms. The first one searches for the optimal combination of measures that helps the project to reach the target duration while considering the effectiveness of the measures. The second term aims to prioritize the mitigation strategy that has the highest duration reduction. Lastly, the third term tries to minimize the delay after the target time in case the mitigation measures are not enough.

In the Kammouh et al. (2022) model, as mentioned earlier, the objective is to find the mitigation measures that minimize the net cost. The objective function (Equation 1) is composed of three terms. The first one aims to minimize the direct cost of implementing the measures, while the second term seeks to reduce the daily penalties for a late completion of the project, and the third term tries to maximize the daily reward obtained for an early completion. As explained in Kammouh et al. (2022), the solution must comply with the compatibility constraint in Equation 2, which states that the difference between the duration of each path before mitigation and the total mitigated duration must be less than or equal to the target duration plus the delay or minus the early completion margin. Additionally, Equation 3 defines how to compute the total mitigated duration in each path.

$$\min \sum_{j\,\mathcal{E}\,J} c_j \;.\; x_j + \Delta_1 \,.\, P \;-\; \Delta_2 \,.\, R \tag{1}$$

$$d_k^0 \;-\; MitDur_k \;\leq\; T_{Tar} + \Delta_1 - \Delta_2 \qquad \forall \;\; k \;\mathcal{E}\; K \tag{2}$$

$$MitDur_k \;=\; \sum_{j\,\mathcal{E}\,J}\sum_{i\,\mathcal{E}\,I} P_{k,i} \,.\, r_{i,j} \,.\, m_j \,.\, x_j \qquad \forall \;\; k \;\mathcal{E}\; K \tag{3}$$

2.3 Output

The output of the tool includes a graph showing the cumulative probability curves (s-curves) of the project duration, denoting the difference between not implementing the mitigation measures, using all of them and carrying out only the optimal ones according to the mitigation controller. Another relevant output is the criticality of mitigation measures. The critical index of a mitigation measure is defined as the "percentage of the Monte Carlo iterations in which a measure is included in the optimal mitigation strategy" (Kammouh et al., 2021). This output is vital to understand which are the best mitigation measures to implement. Other outputs include: the cost cumulative distribution of the project with the different scenarios stated before, the critical index of activities and critical paths (also differentiating using the Mit-C and not doing it).

3. Inclusion of Resource Constraints

To answer the first sub-question, **“What are the existing techniques or methods to tackle the resource-constrained problem?”**, this section provides a review of the relevant literature. The review starts by classifying the types of resources found in the Resource Constrained Project Scheduling Problem (RCPSP) literature, then provides an overview of the main solution methodologies, and ends with a short description of the constraints.

As introduced earlier and explained by Khajesaeedi et al. (2024), the RCPSP “seeks to optimize project schedules by minimizing the makespan (total project duration) while satisfying resource constraints and precedence relations among activities”. Thus, there are two main groups of constraints: temporal constraints and resource constraints. In this section, the focus will be only on the resource constraints, since the mitigation controller considers precedence relationships outside the optimization.

3.1 Classification of resource constraints

Appendix B provides a detailed overview of the types of resources found in the RCPSP. Here, in Table 2, a summary of that overview is presented.

Classification	Classification 1 Chaudhary and Meshram (2025)	Classification 2 Hartmann and Briskorn (2009)
Resources	• Human resources • Equipment availability • Material constraints • Financial constraints • IT resources	• Renewable resources • Non-renewable resources • Doubly-constrained resources • Partially renewable resources • Discrete resources • Continuous resources • Cumulative resources • Time-dependent resources • Multiple-Skills resources

Table 2: Resources classification

In the current development, the most relevant resource types to be considered are the human resources, equipment and materials, following Classification 1. The financial constraint is somewhat already taken into account in the existing tool, and the IT resource is not fully relevant for the model. Following Classification 2, the types of resources that will be included in the development are renewable and non-renewable. Other resource variants mentioned above, such as multiple-skill resources, may be relevant in the construction industry. However, they would probably increase the complexity of the tool, thus, their inclusion should be investigated in future research.

3.2 Overview of solution methodologies

As explained by Hartmann and Briskorn (2009), while the basic RCPSP considers only renewable resources, variants of the RCPSP like the Multi-Mode scheduling problem (MRCPSP or MMRCPSP) allow for the inclusion of other types of resources, such as non-renewable resources. To solve these problems, according to Khajesaeedi et al. (2024), the RCPSP literature includes three categories of solution methods that involve a trade-off between solution quality and computational effort:

1. **Heuristics and metaheuristics:** The authors state that these approaches, such as genetic algorithms, are designed to find feasible solutions quickly, which is useful for large problems. However, they do not guarantee that the solution found is the global optimum
2. **Exact methods:** These approaches, such as the Mixed Integer Linear Programming (MILP), are computationally more intensive, especially in large problems, but are designed to find the global solution.

According to Khajesaeedi et al. (2024), most RCPSP researchers use metaheuristic algorithms and there is a growing interest in hybrid approaches. Nevertheless, this thesis will continue to use the exact method established in the original mitigation controller, MILP, since it guarantees finding the global optimum. This will be further explained in Section 4.1.

3.3 General mathematical formulation of constraints

Now, regarding the typical constraints found in the RCPSP, Schwindt and Zimmermann (2015) propose some basic equations. For renewable resources, they suggest:

$$\sum\nolimits_{i \in V} \sum\nolimits_{\tau = t - pi + 1}^{t} r_{ik} \cdot x_{i\tau} \leq R_k \qquad (t \in H \; ; \; k \in \mathcal{R}) \tag{4}$$

Equation 4, according to Schwindt and Zimmermann (2015), expresses that "the sum of resource requirement of activities in progress at each time $t \in H$ cannot exceed the capacity of any resource $k \in \mathcal{R}$, being $\mathcal{R}$ the set of discrete renewable resources and H the scheduling horizon.

For non-renewable resource constraints within a multi-mode context, they propose the following general equation:

$$\sum\nolimits_{j \in V^a} \sum_{m \in \mathcal{M}_j} \sum\nolimits_{\tau = EC_j}^{LC_j} r_{jkm} \cdot z_{jmt} \leq R_k \qquad (k \in \mathcal{R}^n) \tag{5}$$

Equation 5, proposed by Schwindt and Zimmermann (2015), ensures that the non-renewable resource limit is not exceeded. As can be noted, this formulation uses “modes” to describe the relationship between the duration of activities and their resource consumption. If modes are used, the renewable resource constraints (Equation 4) must also be modified to account for them.

4. Mathematical formulations

This section presents the mathematical formulation of the changes that were carried out on the existing mitigation controller tool. It will try to answer the second sub-question: **How can the resource constraints be incorporated into the existing mitigation controller model?** More specifically, it will start with a short explanation of why the problem should be kept linear. Then, a description of some changes implemented related to the penalties and rewards will be introduced. This will be followed by the definition of renewable and non-renewable resource constraints. Lastly, the section will end with a diagram summarizing all the equations.

Table 3 outlines the symbols used in this section.

Symbol / notation	Description
$\boldsymbol{\Delta_1}$	Delay after implementing the mitigation measures
$\boldsymbol{\Delta_2}$	Duration reduction beyond the target duration of the project
$\boldsymbol{C_{nr}}$	Capacity of non-renewable resource nr
$\boldsymbol{C_r}$	Capacity of renewable resource r
$\boldsymbol{D_i}$	Duration of activity i with no mitigation
$\boldsymbol{f_i}$	Finish time of activity i. f_I is the finish time of the last dummy activity
$\boldsymbol{I}$	Number of activities in the project (with activities indexed by i)
$\boldsymbol{J}$	Number of mitigation measures (with mitigation measures indexed by j)
$\boldsymbol{M}$	“Big-M” constant used for linearization
$\boldsymbol{MC_j}$	Mitigation capacity of measure j
$\boldsymbol{NR}$	Number of non-renewable resources (with non-renewable resources indexed by nr)
$\boldsymbol{p_{i,w}}$	Relation parameter that states if activity i is preceded by activity w
$\boldsymbol{q}$	Auxiliary binary variable used to linearize delta-related constraint
$\boldsymbol{R}$	Number of renewable resources (with renewable resources indexed by r)
$\boldsymbol{r_{i,j}}$	Relation parameter that states if mitigation measure j can intervene upon activity i
$\boldsymbol{r_{j,nr}}$	Relation parameter representing the number of non-renewable resources nr required by mitigation measure j
$\boldsymbol{r_{j,r}}$	Relation parameter representing the number of renewable resources r required by mitigation measure j
$\boldsymbol{s_i}$	Start time of activity i. s_1 is the start time of the first dummy activity
$\boldsymbol{T}$	Number of time periods (days), indexed by t
$\boldsymbol{T_{end}}$	Duration of the project after being mitigated
$\boldsymbol{T_{tar}}$	Target project completion duration
$\boldsymbol{x_j}$	Binary variable that states if mitigation measure j is implemented or not
$\boldsymbol{y1_{j,t}}$	Auxiliary binary variable used to linearize renewable-resource constraints
$\boldsymbol{y2_{j,t}}$	Auxiliary binary variable used to linearize renewable-resource constraints
$\boldsymbol{z_{j,t}}$	Binary variable that states if mitigation measure j is implemented at time t

Table 3: Description of symbols / notations

4.1 Integer Linear Programming

A programming model can be linear or non-linear. As described by Williams (2013), linear programs are characterized by having objective functions and constraints as linear expressions. In contrast, when at least one of the functions, either objective or constraints, is non-linear the program is considered non-linear.

Linear programs are usually preferred for the simple reason that they are easier to solve than non-linear ones. As explained in Hillier and Lieberman (2001), the easiest types of models with only two variables can be solved by a graphical procedure. Additionally, Luenberger and Ye (2021) state that linear formulations are preferred not only because they are easy to solve, but also because of the simplicity of their objective and constraints definitions. Moreover, since all linear programs are convex, if a feasible solution is found, this will coincide with the global optimum (Nocedal & Wright, 2006). The global optimum is relevant because it is the best feasible solution (either minimum or maximum depending on the objective) and the value is unique. It should be clarified, however, that a linear program might have one single value of global optimum but there can be multiple optimum solutions (combination of the values of the variables) that reach this optimum. For instance, in the case of the mitigation controller, if the model is convex, there will be a single minimum net cost but there will be probably different combinations of mitigation measures that can lead to this minimum.

Although linear programs are simple to define, easy to solve, and finding a global optimum is guaranteed, they have a significant drawback. As explained by Hillier and Lieberman (2001), to make a model manageable, some approximations and simplifying assumptions are needed. Consequently, linear models often lack realism.

Non-linear problems are, in some sense, the opposite to linear ones. While they represent reality more accurately, they are more complex and, thus, they require more computational time. Also, the solution found is a local optimum. While this local optimum may also be the global optimum, this is not guaranteed, and additional verification is needed.

By nature, the mitigation controller model is non-linear. However, due to the advantage of the global optimum, it is preferred to have a linear model. Williams (2013) states that some non-linear programs can be converted to linear ones by implementing integer programming but highlights that this comes with a high computational cost. This difficulty arises because restricting variables to be integers introduces a more complex search method, making the model more computationally demanding than a purely continuous linear program. Williams (2013) also mentions that integer program models can be pure integer when all the variables are of this type, or mixed integer when some of the variables are integers and some other are continuous. In the following sections (4.2-4.4), it will be explained how the mitigation controller program can be kept linear by adding integer variables.

4.2 Penalties and rewards

As mentioned earlier in Section 2, a previous model of the mitigation controller included the concept of penalties and rewards for late and early project completion, respectively. Kammouh et al. (2022) explain that there are three possible scenarios after the optimization. As shown in the figure extracted from the same source (Figure 1), the duration of the project after the optimization (i.e. post-mitigation) can be equal to, greater than, or less than the target time. Clearly, the three scenarios cannot happen at the same time. Therefore, one of the constraints included in the original mitigation controller is that either Δ_1 (residual delay), Δ_2 (reduction beyond target time), or both must be zero and, thus, the product equals zero, as in Equation 6 (Kammouh et al., 2022).

$$\Delta_1 \times \Delta_2 = 0 \tag{6}$$

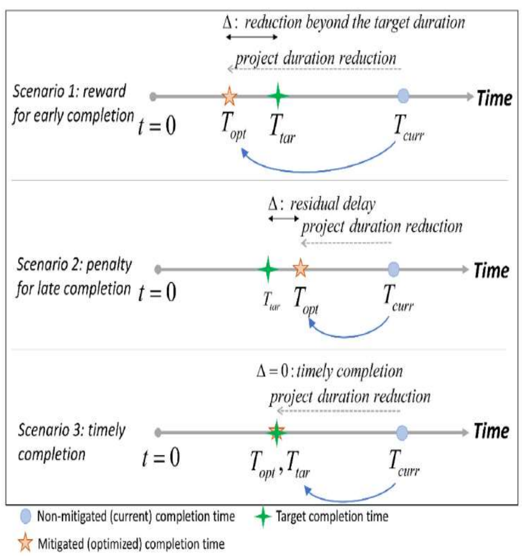

Figure 1: Optimization scenarios (Kammouh et al., 2022)

Since the previous equation implies the multiplication of two variables, the equation is non-linear. However, as explained in section 4.1, it is desired to keep all the constraints as linear equations or inequations in order to find the global optimal solution. Consequently, the previous constraint must be linearized.

It is evident that the mentioned constraint can be classified as a “Either/Or Constraint” as described in Forrester and Waddell (2022). In other words, either $\Delta_1 \geq 0$ or $\Delta_2 \geq 0$. The authors explain that this type of restrictions can be modeled by incorporating an “auxiliary binary variable" and a big number ("Big M"). The value of the binary variable is the one that defines which of the two constraints is imposed. The authors also explain that the Big M is usually a large number that, when its term is different from zero, it causes the constraint to lose its relevance. Nevertheless, in the current case, a different approach is taken. A switch binary variable q is introduced as shown in Equations 7 and 8.

$$\Delta_1 \geq 0 \rightarrow \Delta_1 \leq M \, . \, q \tag{7}$$

$$\Delta_2 \geq 0 \rightarrow \Delta_2 \leq M \, . \, (1 - q) \tag{8}$$

Here, if q takes a value of 1, Δ_1 will have a high upper bound while Δ_2 will be limited to be less than or equal to zero. Conversely, if q takes a value of 0, Δ_1 will be forced to be less than or equal to zero and Δ_2 will have a large upper bound. It should be highlighted that to ensure that one of the Δ is equal to zero and not a negative number, the lower boundary of these two variables must be zero.

Concerning the value of the "Big M", in this case a prudent number would be the duration of the longest critical path before mitigations are applied. This is because it is basically impossible that a delay (Δ_1) or a duration reduction (Δ_2) can be higher than the longest duration of a project.

Moreover, the constraint cited in Section 2.2 must be modified as a consequence of the introduction of the auxiliar variable q . Now, the difference between the duration of the project after being mitigated (T_{end}) and the target duration (T_{tar}) must be equal to either the delay (Δ_1) or the duration reduction (Δ_2), as defined in Equation 9. Furthermore, the inequality in Equation 10 must be introduced to define the value of q. For instance, if the duration of the project after mitigation is larger than the target time, the value of q will be forced to be 1 by Equation 10. Here it is relevant to remember that since q is a binary variable it can only take the values 1 or 0.

$$\Delta_1 - \Delta_2 = T_{end} - T_{tar} \tag{9}$$

$$M \, . \, q \geq T_{end} - T_{tar} \tag{10}$$

Figure 2 illustrates the linearization of the constraint included in the original model. It can be noted how adding a switch binary variable can expand the model, in terms of constraints, to keep it linear.

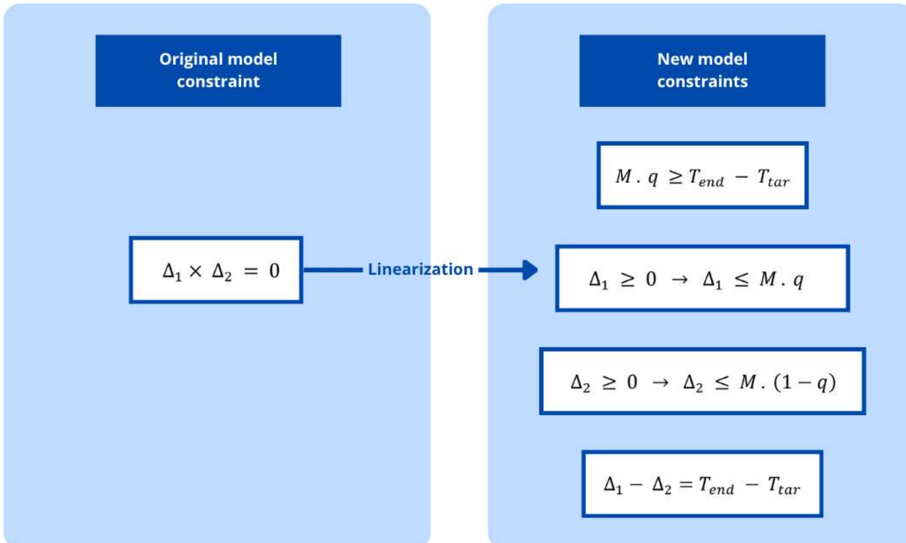

Figure 2: Old constraint linearization

4.3 Renewable resources

The renewable resource constraint could be easily defined as the following: the amount of a certain resource r consumed by all the mitigation measures j being implemented at a certain time t must be equal to or less than the resource capacity C_r. This constraint must be verified for all the renewable resources and for all the time periods of the project. It sounds quite simple, but in fact this involves multiple constraints, and due to the actual modeling of the tool some extra complexities are encountered.

To begin with, the actual mitigation controller model uses the binary variables x_j which symbolize if the mitigation measure j should be implemented or not. Now, the first problem is that the new model needs to know if the mitigation measures are being implemented or not at each specific time of the project. According to Schwindt and Zimmermann (2015), the "time-indexed" integer linear programming formulation is frequently used to model resource-constrained scheduling problems. The authors explain that this type of formulation uses a type of variable, for instance $x_{i,t}$, that indicates the status of an activity i at time t . In a similar way, $z_{j,t}$ binary variables are added to the mitigation controller model. These new variables represent if, at time t , mitigation measure j is implemented or not (1 or 0, respectively).

A second challenge is how to determine the periods of time at which mitigation measures can be implemented. This seems quite intricate to do in the existing model because of a series of factors. Firstly, the periods of time at which each mitigation measure can act should be the same as the time periods of the activities they can be implemented on. Secondly, the time periods of each activity are not known precisely. All the possible paths and the activities durations are determined before the optimization, so the time periods in which each activity is ongoing could be determined. However, after optimization the durations will change based on which measures are activated and their capacity. In other words, the time periods that each mitigation measure may be active depends not only if the optimization decides to implement it or not, but also on the time periods of the activities on which they can be applied to and, simultaneously, this depends on which measures are being implemented.

Consequently, following a similar approach to the time-indexed formulations, a new set of variables are incorporated into the mitigation controller model. These will be the starting and ending times of the activities, s_i and f_i, respectively. The idea is to find these values during the optimization, including the effect of the mitigation capacities, for the purpose of later stablishing $z_{j,t}$. The incorporation of these new variables requires the restrictions stated in Equations 11 and 12. These two inequations imply that the starting (s_i) and finishing times (f_i) of each activity must be equal to or greater than 0. In the code, this is simply defined as the lower boundary of the variables mentioned instead of a constraint.

$$s_i \geq 0 \quad \forall \; i \; \mathcal{E} \; I \tag{11}$$

$$f_i \geq 0 \quad \forall \; i \; \mathcal{E} \; I \tag{12}$$

Moreover, the starting time of the first activity (dummy task) must be equal to 0, as defined in Equation 13. Additionally, the starting time (s_i) of an activity i must be equal to or greater than the finishing time of its predecessor activities (f_w), as expressed by Equation 14. Here, w is used to refer to the predecessor activities and not confuse them with i and, clearly, $I = W$. Also, a new binary relation parameter is used here, $p_{i,w}$, which states if activity i is preceded by activity w or not.

$$s_1 = 0 \tag{13}$$

$$s_i \geq f_w \quad \forall \; i \; \mathcal{E} \; I \quad \wedge \quad w \; \mathcal{E} \; p_{i,w} = 1 \tag{14}$$

Equation 15 establishes that the finishing time (f_i) of each activity must be equal to the sum of the starting time (s_i) and the duration (D_i) of the activity minus the mitigation capacity of the mitigation measures being implemented. Here, the activities duration is the sum of the random values of the activity duration, the risks effect that act on them and the shared uncertainties duration, as described in Section 2.1. The last term of the inequation represents the summation of all the mitigation capacities of those measures which can be applied to the specific activity and that the optimization chooses as optimal. Furthermore, Equation 16 expresses that the finishing time of the last activity (dummy task), f_I, must be equal to or greater than the finishing times of all the preceding activities.

$$f_i \; = \; D_i \; + \; s_i \; - \; \Sigma_{j=1}^{J} MC_j \; . \; x_j \; . \; r_{i,j} \qquad \forall \;\; i \; \mathcal{E} \; I \tag{15}$$

$$f_I \; \geq \; f_i \qquad \forall \;\; i \; \mathcal{E} \; I \tag{16}$$

Figure 3 shows the equations used to determine the start and finish times of activities after being mitigated.

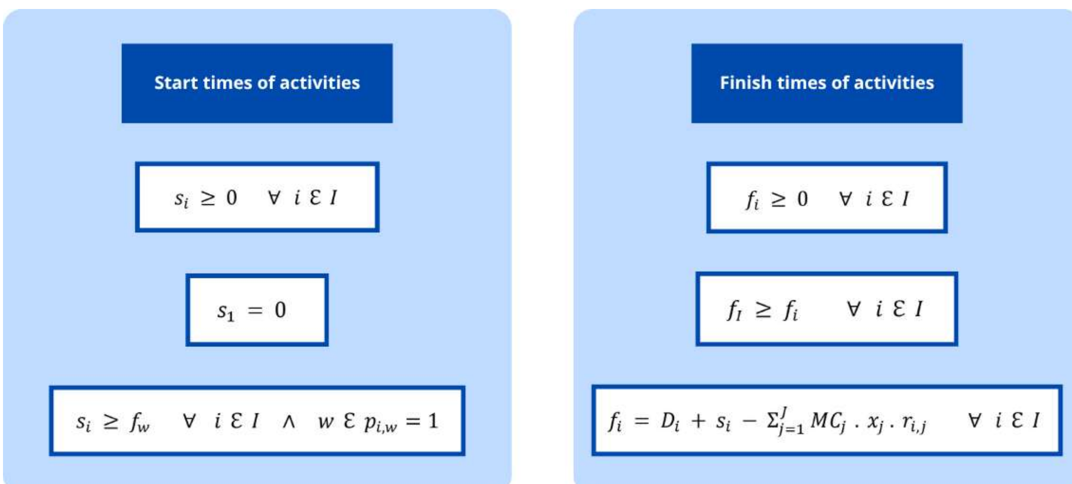

Figure 3: Equations for activity start and finish times

Now that the starting and finishing times of activities after mitigation is known, the renewable resource constraints can be formulated based on the examples given in Schwindt and Zimmermann (2015). Equation 17 states that the sum of the renewable resources ($r_{j,r}$) consumed by all the mitigation measures being implemented and active on time t ($z_{j,t}$) must be equal to or less than the capacity of each renewable resource (C_r). This must be verified for each renewable resource, r , and for each time, t , in the project mitigated makespan. Moreover, it is important to note that, since the t values represent each day of the project duration, T is basically the finishing time of the last activity, that is one of the optimization variables, and this affects the linearity of the model. Therefore, a better approach is to stablish T as the duration of the longest critical path before mitigation. This may create constraints that will not be needed, but it will cover all the possible duration of the project after mitigations are applied. Furthermore, the amount that each mitigation measure j needs of each r resource ($r_{j,r}$) is introduced by the user as an input and then it is actualized based on the mitigation capacities obtained randomly in each MCS iteration. The relationship between the capacities and the resources consumption of the mitigation measure will be further developed in next section (Section 4.4).

$$\Sigma_{j=1}^{J} z_{j,t} \, . \, r_{j,r} \, \leq \, C_r \qquad \forall \;\; r \; \mathcal{E} \; R \quad \wedge \quad t \;\; \mathcal{E} \; T \tag{17}$$

Here, one last obstacle was encountered. The value of $z_{j,t}$ is determined by three conditions: if mitigation measure j is implemented or not, if time t is after the start of activity i affected by measure j or not, and if time t is before the end of activity i affected by measure j or not. This can be expressed using Equation 18, in which each factor on the right side of the inequality represents one of the previous conditions.

$$z_{j,t} \, \geq \, x_j \, . \, (t \, - \, s_i \, - \, 1) \, . \, (f_i \, + \, 1 \, - \, t) \qquad \forall \;\; j \; \mathcal{E} \; J \quad \wedge \quad t \;\; \mathcal{E} \; T \tag{18}$$

The issue with this inequation (Equation 18) is that variables are being multiplicated and, thus, is non-linear. Therefore, a similar procedure to the one explained in Section 4.2 is followed. Two binary auxiliary variables are added per each $z_{j,t}$: $y1_{j,t}$ and $y2_{j,t}$. These extra variables will help to determine if each t is before or after s_i and f_i by including Equations 19 and 20.

$$M \, . \, y1_{j,t} \, \geq \, t \, - \, s_i \, - \, 1 \tag{19}$$

$$M \, . \, y2_{j,t} \, \geq \, f_i \, - \, t \, + \, 1 \tag{20}$$

In Equation 19, if t is greater than s_i the value of $y1_{j,t}$ is forced to be 1. Likewise, in Equation 20, if t is lower than f_i, then $y2_{j,t}$ is constrained to be 1. Regarding the value of the "Big M", like the case explained in Section 4.2, an appropriate choice would be the duration of the longest critical path before mitigating the project.

Finally, the value of $z_{j,t}$ can be obtained using Equation 21. Here, the three last terms on the right side of the inequation must be 1 for $z_{j,t}$ to take a value of 1. This would mean that the mitigation measure j is being implemented, and the time t is greater than the starting time and lower than the finishing time of the activity affected by the measure. In the rest of the possible cases, $z_{j,t}$ must take a value of 0, but as it can be noticed the inequation does not force this value. Thus, this could be considered a limitation of the model. Furthermore, it is important to highlight that all these auxiliar variables ($z_{j,t}$, $y1_{j,t}$ and $y2_{j,t}$) must be clearly stated as binary, if not they may influence the renewable resource constraint by multiplying the resources consumption.

$$z_{j,t} \geq -2 + x_j + y1_{j,t} + y2_{j,t} \tag{21}$$

Figure 4 summarizes the equations added in the mathematical formulation to incorporate the renewable resource constraints, starting by the general resource constraint inequation, followed by the addition of decision variables and finishing with the linearization of the constraints.

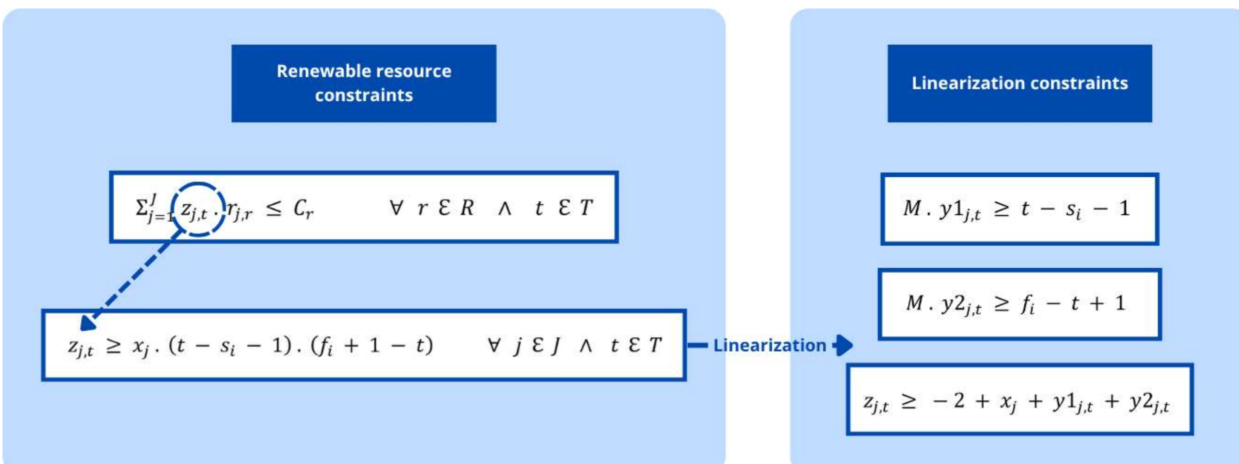

Figure 4: Renewable resource constraints

4.4 Non-renewable resources

As introduced in the previous section (Section 3), the basic Resource Constrained Project Scheduling Problem (RCPSP) only includes renewable resources and "modes" are usually implemented to incorporate the non-renewable type. Schwindt and Zimmermann (2015) explain that the modes define the relation between the resource consumption of the activities and their duration. The concept of modes could also be implemented for mitigation measures, describing the relation between the resource requirement and the mitigation capacity. However, this characterization may not be effective in the current model of the mitigation controller. The tool obtains the mitigation capacity randomly from a

Beta-PERT distribution and, consequently, it would be necessary to have as many modes as possible number from the distribution which would increase significantly the amount of data requested as input. It would be more efficient to have a function that defines the relationship between the mitigation capacity and the resource consumption. Nevertheless, these types of functions are challenging to define. The relationship between mitigation capacity and material consumption is clearly different to the one with human resources since the latter may include influencing factors such as tiredness. Thus, in the current development, these relationships will be simplified and described as linear. Future research may focus on including a more complex and realistic characterization of these relationships.

Regarding the non-renewable resource constraints mathematical formulation, it is a little bit more straightforward in comparison to the renewable one. Each mitigation measure can consume a specific amount of each non-renewable resource and each of these resources has a maximum capacity for the whole project. Since the resources cannot be renovated, the constraints only depend on the implementation of the mitigation measures, and time does not play a significant role.

Based on the previous description and other models, such as the ones introduced in Schwindt and Zimmermann (2015) and Ramos et al. (2023), the constraint in Equation 22 can be defined. It is important to highlight that the previous models are focused on the activities' resources and implement the modes method, thus, their inequations are a little bit more complex than the one presented here.

$$\Sigma_{j=1}^{J} x_j \,.\, r_{j,nr} \leq C_{nr} \qquad \forall \quad nr \;\mathcal{E}\; NR \tag{22}$$

Equation 22 implies that the summation of the resource nr consumed by each mitigation measure being implemented, x_j , must be equal to or less than the capacity of the nr resource, and this must be complied for all the NR resources. The amount that each mitigation measure j needs of each nr resource ($r_{j,nr}$) is defined by the user as an input and then it is actualized based on the mitigation capacities obtained randomly in each MCS iteration.

4.5 Summary

Figure 5 compares the original mitigation controller model with the new resource-constrained version. It can be noticed the significant number of variables and constraints needed to include the resources required by mitigation measures while keeping the model linear.

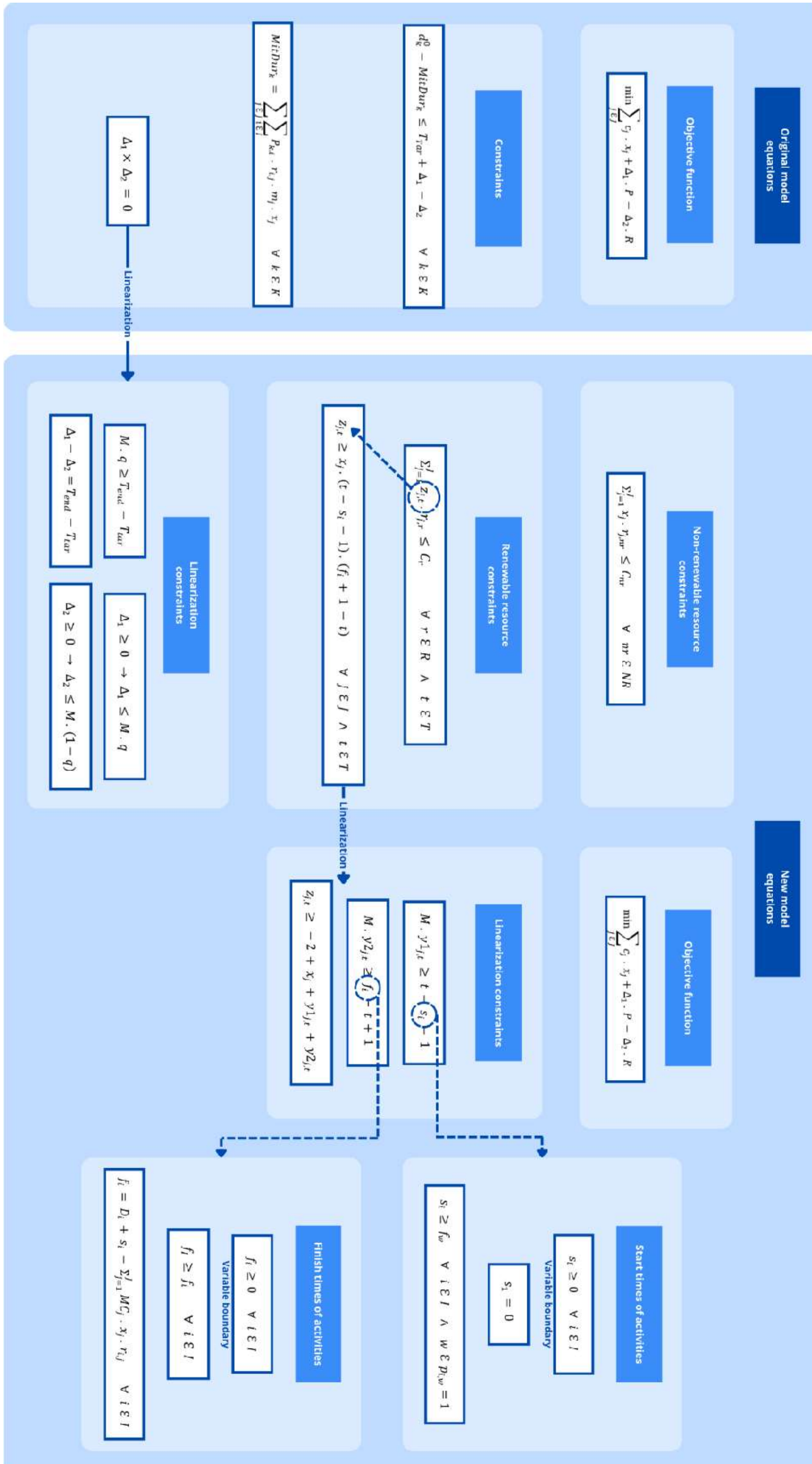

Figure 5: Mathematical formulation summary

5. Tool modelling

5.1 Optimization variables and functions

Table 4 below summarizes the variables used in the new version of the mitigation controller. As can be observed, there are both binary and continuous variables, thus the optimization problem is a Mixed-Integer Linear Program. To model the problem in Python the "milp" function from the "scipy.optimize" module was used.

The "milp" function in SciPy requires the constraints to be provided in matrix form. Since the problem involves a large number of constraints and variables, the resulting matrix would be very large if stored in dense form (saving all the coefficients of each variable in each constraint). However, most of its entries are zero, meaning that a dense representation is unnecessary. Therefore, the constraints were represented using sparse matrices from the "scipy.sparse" module, which store only the non-zero coefficients. This significantly reduces memory usage and improves the computational efficiency of the MILP formulation.

<table>
<tr><th>Variable</th><th>Type</th><th>Lower bound</th><th>Upper bound</th><th>Comments</th></tr>
<tr><td>x_j</td><td>Binary</td><td>0</td><td>1</td><td>There are J variables. These are the decision variables which state if each mitigation measure is implemented or not.</td></tr>
<tr><td>Δ_1</td><td>Continuous</td><td>0</td><td>None</td><td rowspan="2">These are the residual delay and the duration reduction beyond the target time. These two variables represent time, and since other time related variables in previous versions of the Mit-C were considered integers, it could be assumed that these also should be integers. However, it was opted to consider them continuous in order to relax the optimization.</td></tr>
<tr><td>Δ_2</td><td>Continuous</td><td>0</td><td>None</td></tr>
<tr><td>q</td><td>Binary</td><td>0</td><td>1</td><td>This is the switch variable that helps to linearize the constraints related to the previous variables as explained in Section 4.2</td></tr>
<tr><td>s_i</td><td>Continuous</td><td>0</td><td>None</td><td rowspan="2">There are I variables of each of these two. These variables represent at which time each activity starts and finishes. Also, similar to the previous time related variables, it was decided to consider them continuous variables to relax the program.</td></tr>
<tr><td>f_i</td><td>Continuous</td><td>0</td><td>None</td></tr>
<tr><td>$z_{j,t}$</td><td>Binary</td><td>0</td><td>1</td><td rowspan="3">There are $J \times T$ variables of these three. As explained in Section 4.3, the first ones state if each mitigation measure is active at each time of the project, and the second and third ones help define the first one.</td></tr>
<tr><td>$y1_{j,t}$</td><td>Binary</td><td>0</td><td>1</td></tr>
<tr><td>$y2_{j,t}$</td><td>Binary</td><td>0</td><td>1</td></tr>
</table>

Table 4: Optimization variables summary

5.2 Mitigation controller workflow

This section describes the workflow of the new mitigation controller shown in Figure 6. Taking into account that the tool is based on previous versions of it, most of the workflow is similar to the one defined in Kammouh et al. (2022) and summarized in Section 2.1.

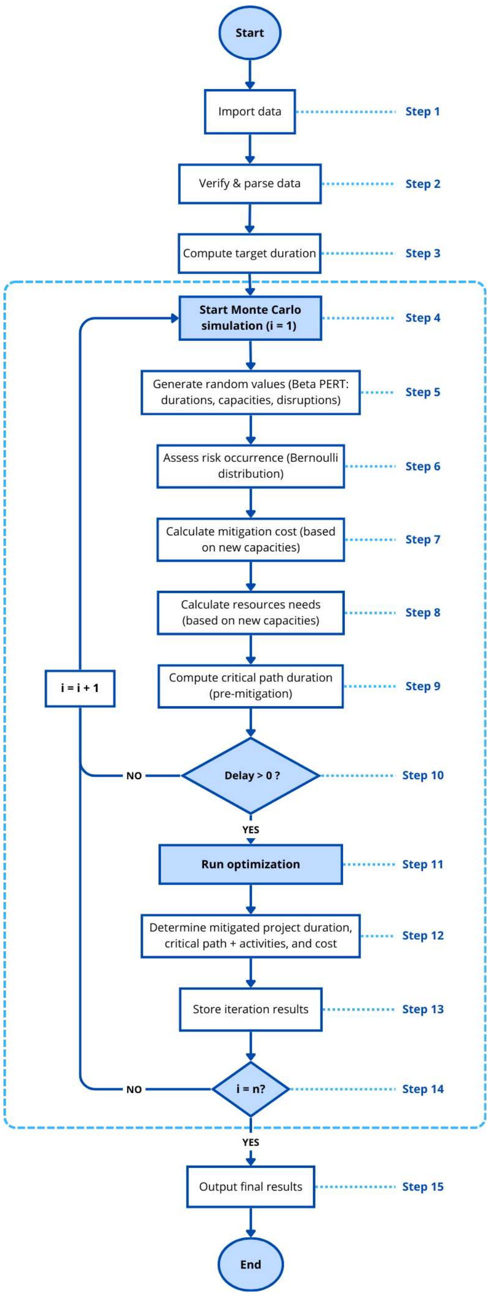

Figure 6: Mitigation controller flow diagram (based on Kammouh et al. (2022))

As can be seen in Figure 6, first, the data provided by the user as an excel file must be imported, verified for completeness and proper data entry, and then processed (Steps 1 and 2). The data required as input is basically the one of previous models (stated in Section 2.1) plus the renewable and non-renewable resources needed by each mitigation measure. In addition, if a target duration is not given as an input, this value must be computed as the project duration when the activities follow their most likely durations (Step 3).

Subsequently, as shown in Figure 6, the MCS can be started. In each iteration, Steps 5 to 13 must be followed. To begin with, some extra data processing must be carried out. Random values from BETA-PERT distributions must be obtained for the following: correlated and uncorrelated durations of each activity, disruption duration of each risk, and mitigation capacity of each measure (Step 5). It is also necessary to determine if each risk event occurs or not (binary value) by drawing a random value from a Bernoulli distribution (Step 6). Moreover, using the new random mitigation capacities of the measures, the cost associated with them and the resources required by them must be re-calculated (Steps 7 and 8).

Once all the previously mentioned data is acquired, the duration of each activity before mitigation can be calculated (only considering correlated and uncorrelated durations of activities, and the disruption durations of the risks occurring). These new activities durations are used to determine the critical path before mitigation and its duration, which later is implemented as the M and T values from Section 4.2 and 4.3 (Step 9).

Thereafter, if the project duration is delayed with respect to the target duration, the optimization is run considering the minimization function cited in Section 2.1, the variables summarized in Section 5.1 and the constraints described in Section 4 (Step 10 and 11). The aim is to “find the optimal combination of measures that minimizes the net cost, which is the summation of the mitigation costs and the penalty (in case of delay) or the reward (in case of early completion)” (Kammouh et al., 2022), while considering the resources requirements and capacities. The first J values of the optimization will define if the mitigation measures must be implemented or not to reach the minimum net cost.

With this information, it is possible to determine the optimal duration of the project, the critical path after applying the optimal mitigation measures, and the activities that comprise it (Step 12). Additionally, to enable graphical comparisons, the project’s cost and duration when implementing all the measures must also be calculated. Next, the results of the iteration must be stored (Step 13). This includes: the output of the optimization and the values calculated for project cost, duration, critical path and its activities (before and after mitigation and with all and only optimal mitigations).

As depicted in Figure 6, once the optimization results are saved for the current iteration, a new iteration begins. Likewise, a new iteration starts when there is no delay, and the optimization is not carried out. This will occur till the number of iterations reaches the one stablished by the user. Finally, the MCS ends and the graphs summarizing the results of all the iterations are shown (Steps 14 and 15).

6. Validation

6.1 Description of case study

This section introduces a case study used to validate the new mitigation controller and to try to answer the last sub-question: **What is the effect of including the resource constraints in the mitigation controller?**

The case study project consists of the construction of a warehouse which will have the function of sort center for an international e-commerce company. It includes the construction of a main building with a storage and office zone and works in the exterior area. The structure of the warehouse is made up of an internal steel structure forming the columns and trusses, surrounded by concrete walls built with the tilt-up method, supported by isolated footings for the columns and a continuous perimeter footing for the walls. In addition, the structure includes a concrete slab and a metal deck roof.

The construction project was carried out by a company that chose to be kept anonymous, and the data was shared by the project manager. The real data was simplified and slightly modified to maintain confidentiality. Typically, construction projects involve a larger number of activities, risks and mitigation measures than those used in the validation of the tool.

Two validation exercises were carried out. The first one was performed using a highly simplified version of the project data. The preliminary results showed a proper functioning of the tool. However, when these results were shown to the project manager, he pointed out that the tool would be more useful if the input data were more precise and realistic. Thus, a second validation was carried out using a more detailed version of the project data. Table 5 compares the data sizes for the two validation exercises.

	First Validation	Second Validation
Number of activities	13	42
Number of risks	19	31
Number of shared uncertainty factors	0	5
Number of mitigation measures	24	54
Number of renewable resources	5	8
Number of non-renewable resources	1	1

Table 5: Data size

Moreover, previous studies of the mitigation controller compared scenarios with different penalties and rewards. However, since the focus of this master thesis is the addition of resource constraints, the evaluation will focus on the scenario with no rewards for early completion and an extremely high penalty for delays (10^9). This scenario will concentrate on choosing the mitigation measures necessary for the project to reach the target duration.

Project schedule

The construction project, originally consisting of about 80 activities, was summarized to its 10 principal ones for the first validation exercise. Each of the selected activities had several sub-tasks in the original schedule. Table 1 of Appendix C shows the summarized schedule with the most-likely, pessimistic and optimistic estimates of the activities' durations, in addition to their precedence relationships. For the second validation, the schedule data was expanded to 40 activities, as shown in Table 2 of the same appendix. Furthermore, the difference between the two validation exercises can be seen in Figures 7 and 8, which show the project networks. It can be observed that the second validation exercise has a more intricate network with a higher number of parallel activities, which suggests a greater potential for conflicts over renewable resources.

This version of the mitigation controller only allows for strict finish-to-start precedence relationships. Consequently, the schedules from the two validations differ slightly from each other and more significantly from the real project schedule. This difference can be clearly noted in the resulting project duration. In this case, a target duration was not specified, therefore, the tool took the "original duration" as the target and calculated it using the most-likely duration of the activities. While the real project's most-likely duration was less than one year, the tool calculated a duration of 393 days in the first validation exercise and 376 days in the second one.

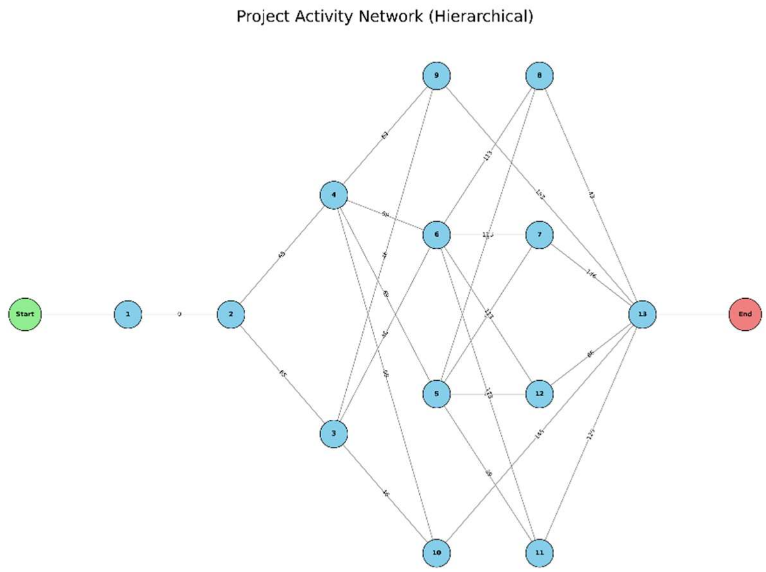

Figure 7: Project network (first validation exercise)

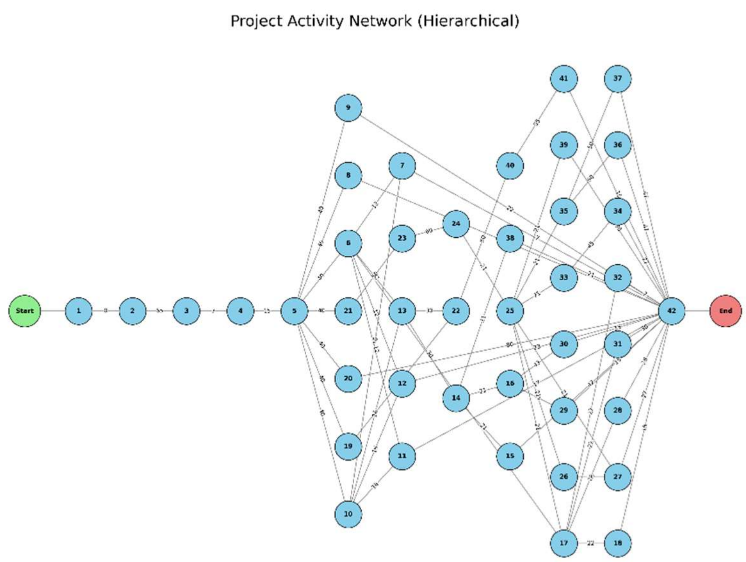

Figure 8: Project network (second validation exercise)

Risk events and Activities' correlation

Tables 3 and 4 of Appendix C display a summary of the risk events for the first and second validation, respectively. These tables include three-point estimates of the amount of delay caused by each risk event, their occurrence probability, and their relationship to the activities. Furthermore, if a risk affects more than one activity, it is entered into the table once for each affected activity.

The mitigation controller tool also requires as input a table including the shared uncertainty factors with their three-point estimates and the activities they affect. These factors, such as "issues with suppliers", are usually defined and managed in projects as risks. Thus, in this case, most of them are entered as risks factors. Only in the second validation exercise, in which the schedule data was more precise, were a few specific shared uncertainties also included (Table 5 of Appendix C).

Mitigation measures

The possible mitigation measures are shown in Tables 6 and 7 of Appendix C for the first and second validation, respectively. The three-point estimates of each measure's mitigation capacity are given in the tables, together with the associated costs (in USD) and the activity number it can be applied to. It is also worth mentioning that those mitigation measures which can be applied to more than one activity (e.g. "Hiring extra personnel") are entered in the table as many times as the number of activities they can be implemented on, just like previous models.

The renewable resources required by each mitigation measure and their available capacities are shown in Tables 8 and 9 from Appendix C for the first and second validation exercises, respectively. Similarly, the non-renewable resources demands and capacities for each exercise are included in Tables 10 and 11 from the same appendix.

Finally, it is relevant to highlight that the two validation datasets differ in the number of mitigation measures and resources included. The data from the first validation only includes 24 mitigation measures, 5 renewable resources (general workers, welding personnel, lifting equipment, crane and laser-drying equipment) and 1 non-renewable resource (high-early-strength concrete). In contrast, the second validation expands this data to 54 mitigation measures, 8 renewable resources (earthworks personnel, concrete personnel, general workers, welding personnel, earthworks equipment fleet, lifting equipment, crane and laser-drying equipment) and 1 non-renewable resource (high-early-strength concrete). It should be noted that for resources related to personnel the unit represents a crew, that is, a value of 1 corresponds to approximately 5 workers.

6.2 Preliminary results

In this sub-section, the results of the first validation are presented. The program was run with Monte Carlo simulations of different sizes: 50, 100, 500, 1000, 2000 and 4000 iterations. In this first exercise, since the data size was small, the execution time was extremely fast when running a small number of iterations. Figure 9 below shows how computational time increases with the number of iterations. For 50 iterations the execution time was only 2 minutes, while for 4000 iterations it was approximately five and a half hours.

In contrast, Figure 10 shows that the “matching result rate” remains stable, even when the number of iterations is increased. This rate is the percentage of iterations in which the resource-constrained Mit-C model reached the same result as the original tool. It can be noted that, when the number of iterations is equal to or larger than 100, this percentage is always around 76%. This indicates that it is not necessary to run a large number of iterations with a high execution time to get a reliable result. Therefore, when using the more detailed data, in the second validation exercise, the program will be run with only 100 iterations.

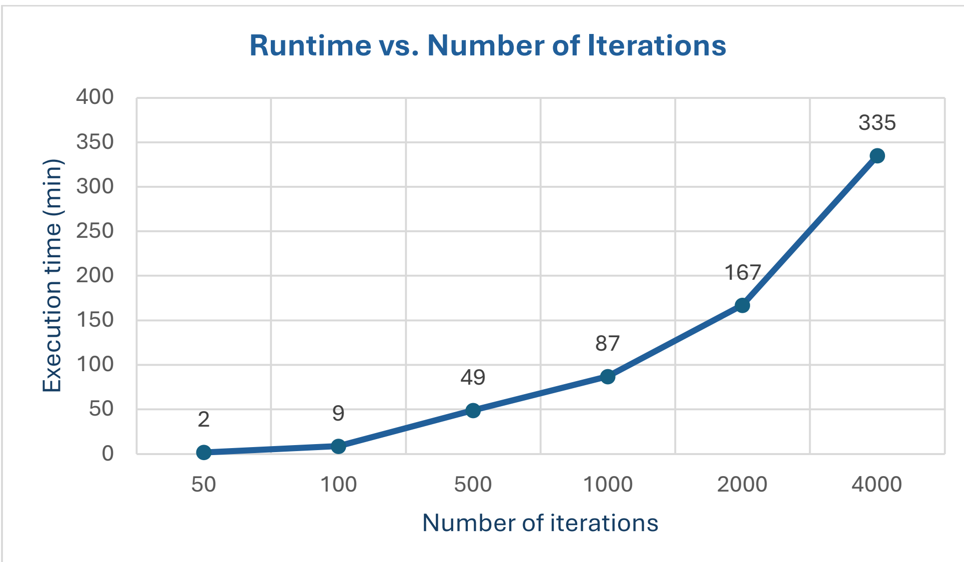

Figure 9: Runtime vs. number of iterations

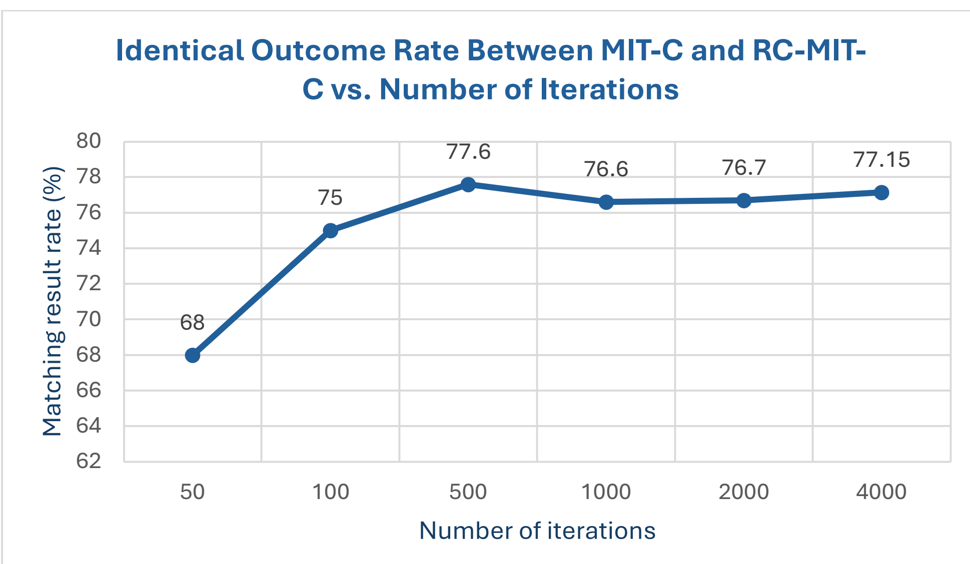

Figure 10: Identical outcome rate vs. number of iterations

Figures 11 to 13 show some of the most relevant graphs generated by the tool when running it with 100 iterations. Figure 11 displays the critical index of the mitigation measures, that is, the percentage of iterations in which they are used in the optimal mitigation strategy (Kammouh et al., 2021). The graph directly compares the results from the resourced-constrained Mit-C model with those from the original tool. Moreover, the graph reveals two key observations. The first one is that both the original and the resource-constrained tools selected the exact same set of mitigation measures (measures 2 to 10). The second observation is that, for these selected measures, the results from both tools are highly similar: each measure reached a comparable critical index in both tools.

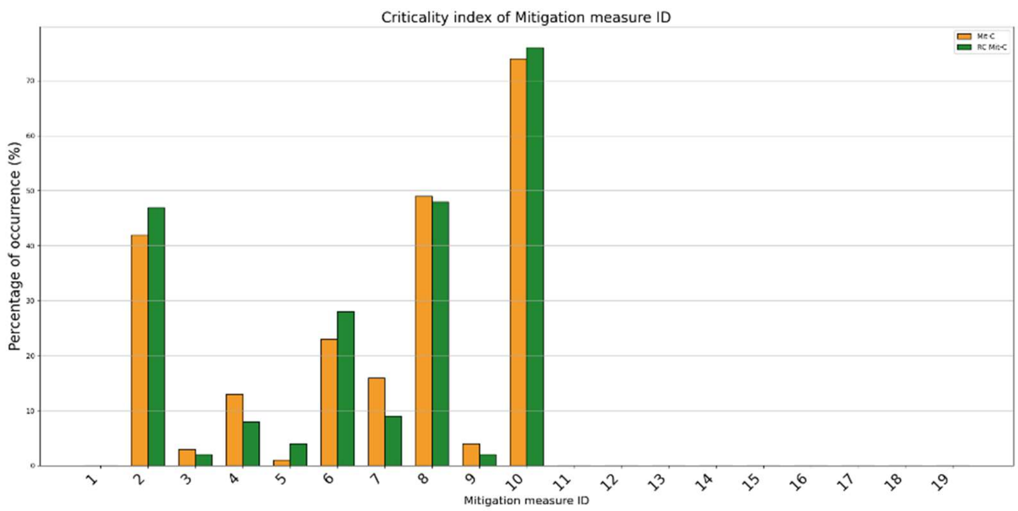

Figure 11: Critical index of mitigation measures (first validation exercise)

Figure 12 shows the cumulative distribution function (s-curve) of the project duration, displaying five different lines:

- The Target duration (black vertical line), representing the project's original most-likely duration.
- The Original scenario (blue s-curve), representing the project with no mitigations applied.
- The Permanent scenario (red s-curve), representing the project when all mitigations are applied.
- Two Tentative scenarios, representing the project when applying the optimal measures suggested by the tools: one using the original Mit-C (orange s-curve) and the other using the resource-constrained version (green s-curve).

Analysis of Figure 12 reveals three significant insights. First, the s-curves for the two Tentative scenarios are nearly identical. This outcome is a direct consequence of the finding from Figure 11. Since both tools selected the same mitigation measures with similar frequency, they logically produce nearly identical project durations. This is further confirmed in Figure 13, where the probability distributions for both Tentative scenarios are almost completely overlapped. This suggests that, for this specific low-detailed case, the inclusion of resource constraints has a minimal impact on the final project duration forecast.

Second, both Tentative scenarios achieve a very high probability of meeting the target duration. This outcome appears overly optimistic, particularly when considering Van Gunsteren et al. (2011) suggestion that a probability of on-time completion over 50% can be considered sufficient. This suggests that the target duration of 393 days, even though it is the original duration, may have been set too conservatively and that an earlier date could also be achievable. However, this result should be seen in context. A similar result was found in the case study with high-penalty condition presented in Kammouh et al. (2022). Furthermore, a closer analysis of the s-curves reveals that the completion date associated with a 50% probability is approximately 387 days, only six days less than the target. This small difference indicates that an even more aggressive mitigation strategy, resulting from a tighter target duration, would have offered only a minimal reduction in the project's overall duration.

Third, the steepness of the s-curves, for both Tentative scenarios, just before the target indicated a low probability of finishing before the target. As rationalized by Kammouh et al. (2022), this is a logical consequence of having no reward for early completion. The optimization has no incentive to use costly mitigations to finish the project before the target. This low variability is also visible in Figure 13, where the duration distributions for the Tentative scenarios are noticeably narrower than those for the Permanent and Original scenarios.

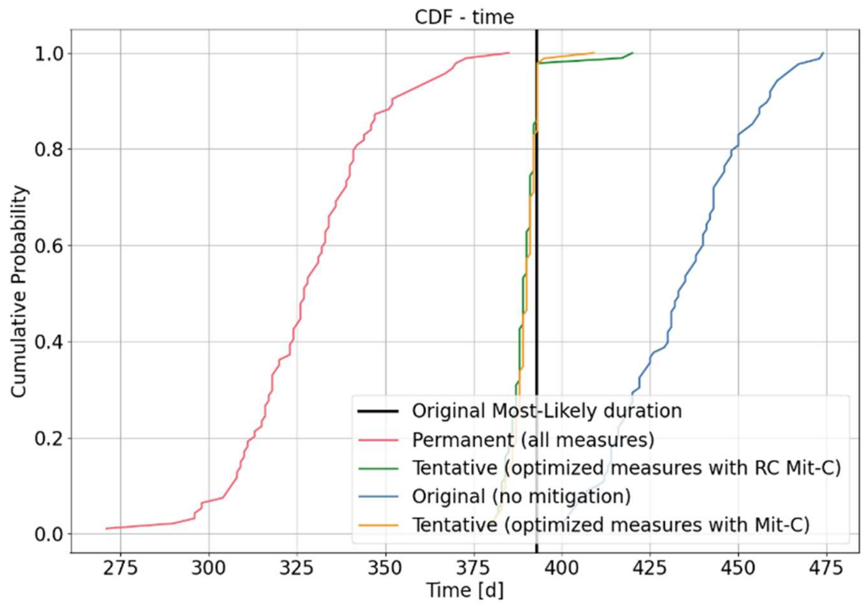

Figure 12: Cumulative distribution function of the project duration (first validation exercise)

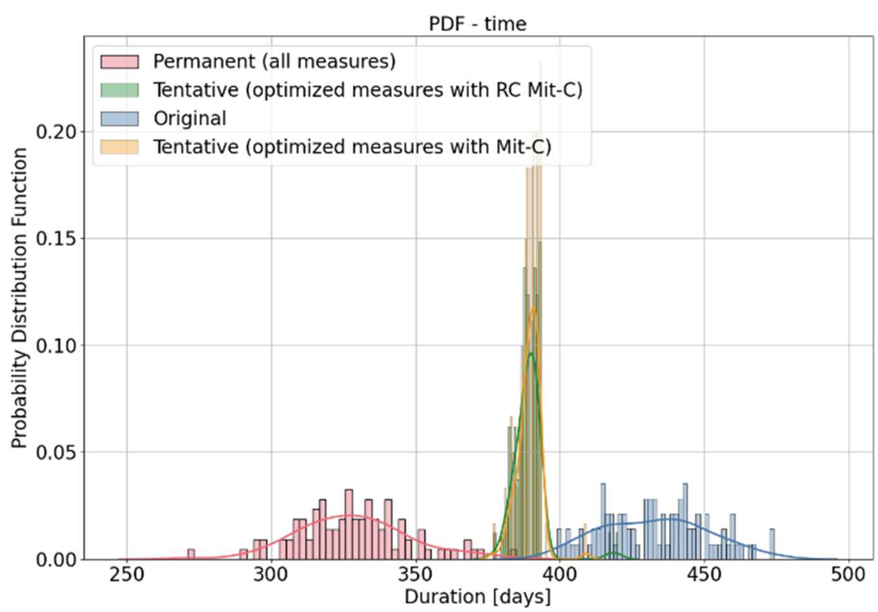

Figure 13: Probability distribution function of the project duration (first validation exercise)

At first glance, these results suggest that the resource-constrained model produces a nearly identical outcome to the original one. However, a deeper analysis reveals a critical hidden difference. While the final project durations were similar, Figure 10 showed that in 24% of the iterations, the optimal mitigation strategy from the original tool was different from the one selected by the resource-constrained model. This implies that in nearly a quarter of the possible scenarios, the optimal strategy from the original tool was, in fact, infeasible because it violated one or more resource constraints, and the new model was forced to find a different and feasible solution

To further investigate the last insight, the optimal strategies from original Mit-C were checked for compliance with the resource constraints. This analysis found that a total of 1730 constraints were not complied with. Figure 14 shows the percentage distribution of these violations by resource, while the raw data can be found in Table 1 of Appendix D. As seen in the figure, the "Crane" was the resource with the highest number of violations, while "high-early-strength concrete" and the "laser-drying equipment" had none.

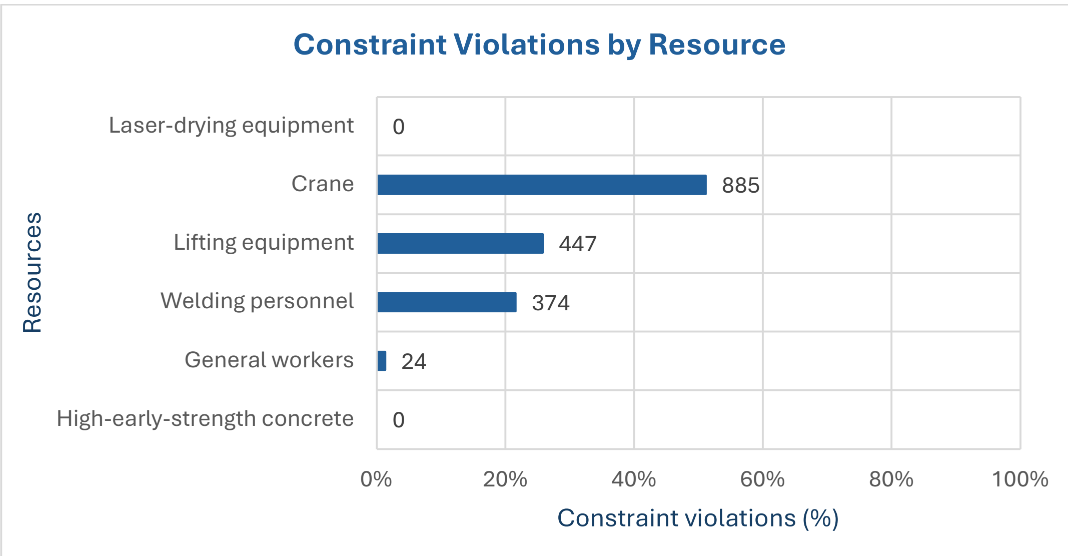

Figure 14: Percentage distribution of constraint violations (first validation exercise)

These results were presented to the project manager from the real project and then an interview was conducted to gather his insights. The most important point raised by the interviewee is that the modeling of the resources is not very detailed in this exercise. He suggested adding more resources to differentiate between specialized worker types, noting that in practice workers are capacitated for different tasks.

6.3 Final results

In this sub-section, the results of the second validation, which used the more detailed project data, are presented. The program was also run for 100 iterations.

The most obvious impact of the increased data detail was on computational performance. The execution time for 100 iterations was 325 minutes (approximately 5.4 hours). This is a dramatic increase compared to the 9 minutes required for 100 iterations in the first exercise. This performance difference highlights the computational cost of increased model complexity. This is logical, considering that for each added mitigation measure, $1 + 3 . T$ extra variables are created, and for each renewable resource, T extra constraints are introduced, where T is the duration of the project in the critical path.

A second significant finding was the sharp decline in the "matching result rate". In this validation, the optimal solutions from the two tools matched in only 37% of the iterations. This is a significant drop from the 76% rate in the first validation exercise. This change is a direct consequence of the more detailed data. By introducing more specific resources and a more complex schedule, the number of potential resource conflicts increases, causing the unconstrained and constrained models to diverge more frequently.

Figure 15, which displays the critical index of mitigation measures, visualizes this divergence. There is now a clear difference between the strategies selected by the original and resource-constrained models. While both tools still use some measures with similar frequencies, the resource-constrained tool now selects a wider variety of mitigation measures. In fact, the new mitigation controller included all the mitigation measures in at least 5% to 10% of the iterations. This suggests that, under more realistic constrained conditions, the optimal strategy is not to rely on a few dominant measures, but to employ a more diverse combination of mitigation measures.

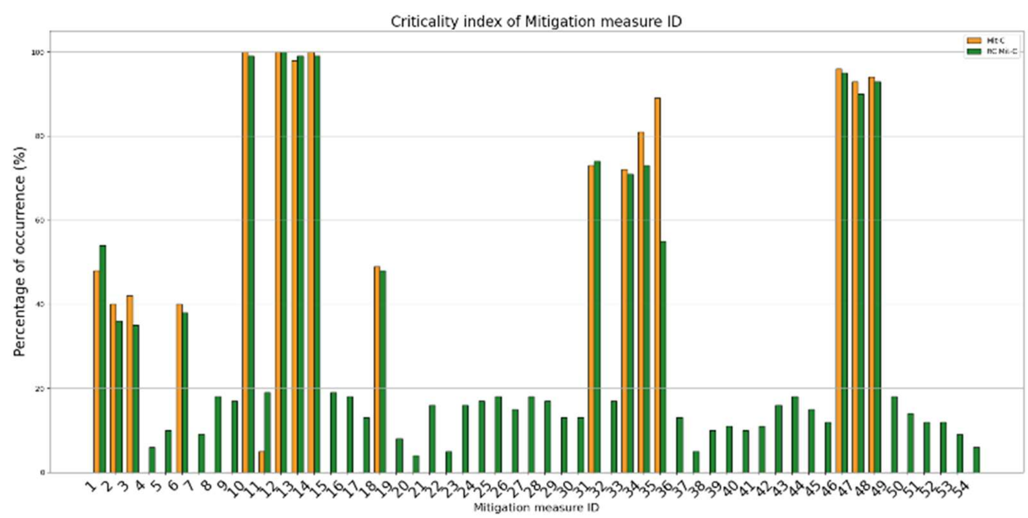

Figure 15: Critical index of mitigation measures (second validation exercise)

The impact on the project duration is shown in the s-curves of Figure 16. The probability of finishing on target decreased for all optimized scenarios compared to the first validation. The Permanent and original Mit-C scenarios both dropped to around 80% probability of finishing on the target. More drastically, the probability of the resource-constrained scenario fell to 60%. This general decline can be attributed to the more detailed risk register, which introduced greater project variability into the possible project durations. The 20% difference between the two Tentative scenarios, however, is a direct result of the resource constraints restricting the use of certain mitigation measures. From a project management perspective, the 60% outcome is still a favorable result. According to the principle suggested by Van Gunsteren et al. (2011), this is a sufficient and acceptable probability for project control.

Another relevant insight from Figure 16 is that the s-curve for the resource-constrained model extends further to the right of the target line. This is confirmed in Figure 17, where the right tail of its probability distribution is longer than the others. This indicates that under realistic resource constraints, the project faces a greater risk of significant delays and it's a logical consequence. The constraints prevent the use of some mitigation measures that are available in the more optimistic and unconstrained scenarios, leading to a wider range of possible negative outcomes with longer project durations. This finding highlights a key contribution of the enhanced tool: while the resource-constrained model is more restrictive and produces less optimistic results, its realism, achieved by including the resources, makes it a more accurate and reliable forecasting tool.

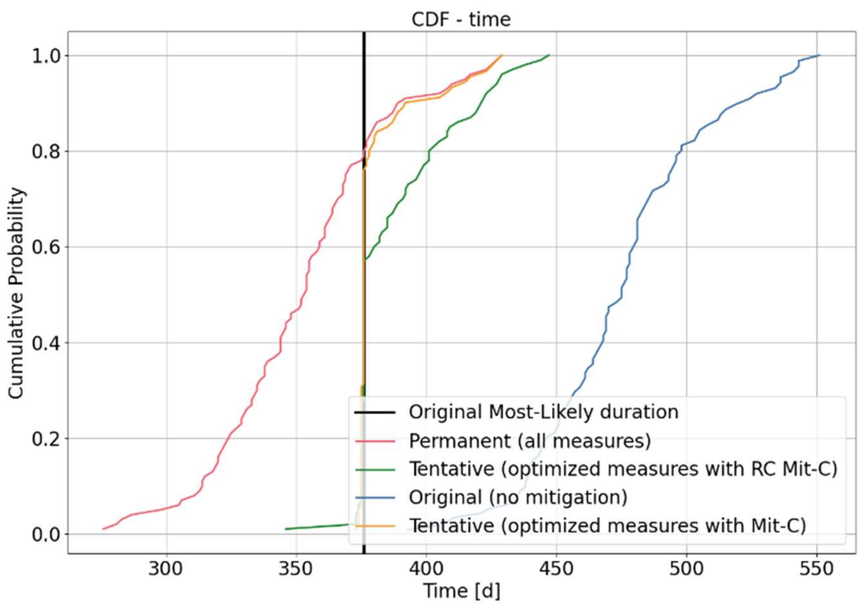

Figure 16: Cumulative distribution function of the project duration (second validation exercise)

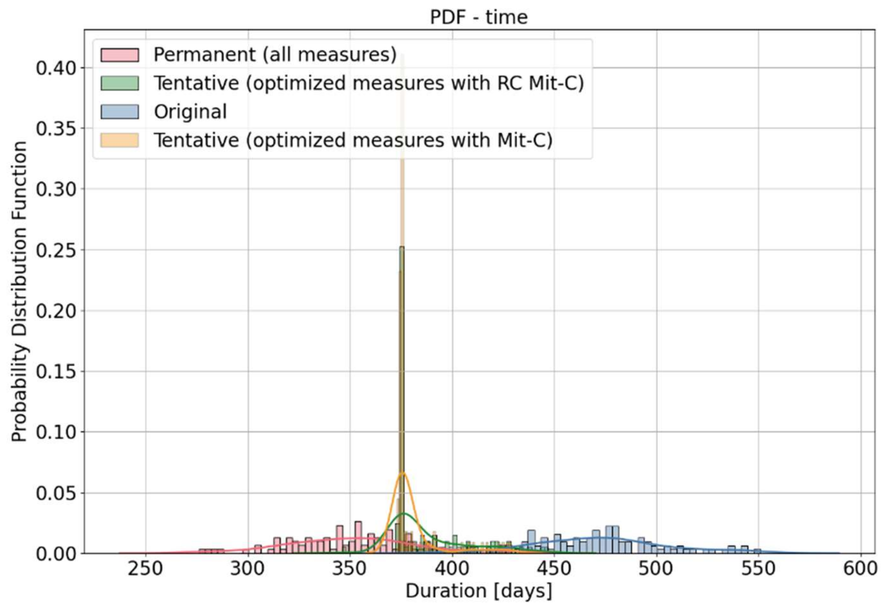

Figure 17: Probability distribution function of the project duration (second validation exercise)

The non-compliance of the original model's output was again quantified. The optimal mitigation strategies from the original Mit-C violated 2685 resource constraints in this validation exercise. Figure 18 shows the percentage distribution of these violations by resource (the raw data can be found in Table 2 of Appendix D). Consistent with the first validation, no conflicts occurred for "high-early-strength concrete" or "laser-drying equipment". Additionally, no violations occurred for several other personnel and equipment resources. This suggests that a more detailed schedule provides a more realistic representation of resource demand over time. By splitting these into a more detailed sequence of activities, the resource demand is spread out more realistically over makespan, thus avoiding artificial violations. This did not eliminate all conflicts, but rather focused them on the truly critical resources.

Consequently, violations were limited to three key resources: the "crane" (a conflictive resource in both validation exercises) and the "earthworks personnel" and "earthworks equipment fleet" (introduced to make the resources more task-specific). The high number of violations associated with these resources highlights the value of using detailed input data.

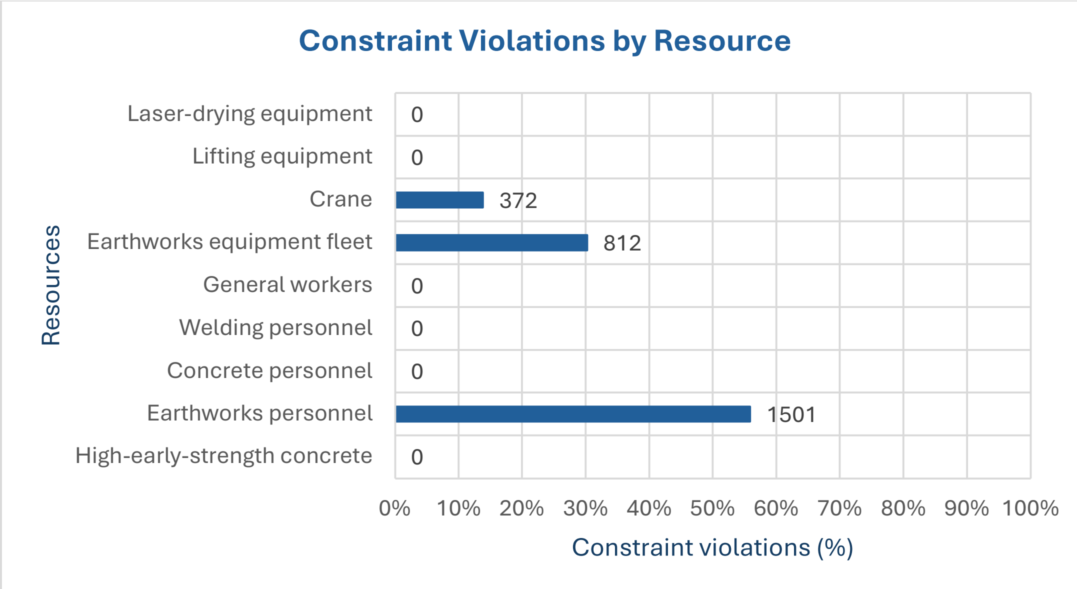

Figure 18: Percentage distribution of constraint violations (second validation exercise)

Finally, the results of this second validation were presented to the project manager and site manager, after which an interview was conducted. Their main comments included the following:

- The interviewees highlighted that the input data format was well structured and transparent.
- The interviewees mentioned that the graphical results were clear and easy to understand. Moreover, they identified the cumulative distribution function (s-curve) of the project duration as the most useful output, as it clearly illustrates the impact of both implementing mitigation measures and including resource constraints.
- The interviewees highlighted the tool's potential to significantly improve the selection process of mitigation measures compared to current practices. They explained that their current approach consists of following their "gut feeling" or intuition, whereas the tool provides a suggestion of optimal measures based on a data-driven analysis of all possible scenarios. Following the tool's strategy is more trustworthy and cost-efficient. Moreover, they mentioned the tool's value for proactive planning, allowing them to prepare for potential delays even before a project is behind schedule.
- The execution time of the program was initially perceived as lengthy. However, this was not considered a problem for the adoption of the tool, since they expect to use it for periodic analysis and not every day.

7. Conclusion

Delays and cost overruns are common problems in the construction industry. Despite extensive research on their causes and mitigations, these problems persist, suggesting that the challenge is not identifying possible mitigation measures, but rather selecting the right combination. Recently, researchers found out that the current way of selecting mitigation measures, by employing Monte Carlo scheduling to evaluate the measures' effect on the project duration, was lacking the project manager goal-oriented control behavior. Consequently, they developed a decision-support tool called Mit-C that combines Monte Carlo simulation with mathematical optimization. The aim of this optimization is to minimize the net cost to find the optimal mitigation strategy. In their research, they proved that by following the optimal strategy the costs of a project could be reduced. Subsequent research has further refined the tool, introducing various modifications. Despite these contributions, some limitations remained. One of these limitations was that the model did not account for resource constraints when determining the optimal strategy. This omission could lead to unrealistic solutions, such as using a mitigation measure in an unlimited way, or creating conflicts between measures competing for shared resources.

Consequently, this master thesis further developed the project management decision-support tool Mit-C by including the availability of and demand for resources required by the mitigation measures. This development primarily involved modifying the existing Mit-C code to incorporate the new resource constraints. While the objective function remained unchanged, several new variables and constraints were introduced throughout the rest of the model to manage the inclusion of resource constraints.

The results demonstrate that including resource constraints has a significant impact on the model's output. Firstly, it alters the optimal mitigation strategy. While some measures are selected in similar ways between the new and the original versions of the tool, the resource-constrained model frequently selects a different optimal strategy than the original. This difference becomes more noticeable when using more detailed input data.

This sensitivity to level of detail in data is a key finding. By analyzing the same project with two different levels of detail, this thesis effectively demonstrates that the optimal mitigation strategy is highly dependent on the project's specific context. Therefore, it can be concluded that the core finding, that resource constraints will always impact the optimal mitigation strategy, is generalizable, while the specific numerical results and the magnitude of that impact are case-dependent. For instance, the more detailed schedule in the second validation, with a higher number of parallel activities, created more resource conflicts and thus forced a more significant divergence from the unconstrained model's strategy.

The change in the optimal strategy directly leads to the second major finding: the inclusion of constraints reduces the probability of finishing the project on the target duration. By narrowing the feasible region, the constraints reduce the possible combinations of measures and limit their use. As a result, the project cannot be mitigated on the same level as with the original model, which in turn lowers the probability of finishing on time. Therefore, while the s-curve generated using the optimal strategies suggested by the resource-constrained tool appears more pessimistic, it represents a more accurate and reliable forecast, as the selected strategies are assured to be feasible in terms of resources.

7.1 Contributions

The main contribution of this master thesis is the conceptual and practical improvement of the Mit-C decision-support tool through the incorporation of resource constraints, ensuring the recommended mitigation strategies are not just optimal, but also feasible.

This was achieved through the adaptation of the Resource Constrained Project Scheduling Problem (RCPSP) theory. The constraints of the RCPSP, which traditionally apply to project activities, were successfully adapted to model the resource consumption of mitigation measures. While multi-mode literature served as key inspiration, significant changes were necessary. The adaptation had to account not only for the shift in focus from activities to mitigation measures but also for the fact that mitigation capacities are sampled from continuous distributions, which is incompatible with the discrete pre-defined choices of a standard modes method.

Furthermore, incorporating both renewable and non-renewable resource constraints was a significant challenge. This is because the original model did not track the start and finish times of activities or mitigations, which is essential for managing time-dependent renewable resources. This was solved by introducing time-indexed variables, which in turn required the addition of several new constraints.

All these modifications were implemented while preserving the model's global optimality. By using auxiliary binary variables, the model was kept linear. This is a relevant achievement, since it means that the new tool maintains one of its most important features: the feasible solution found is the best globally optimal strategy.

The validation of the improved tool demonstrated the importance of this contribution. It proved that accounting for resource availability directly impacts the optimal mitigation strategy selected, which in turn provides a more realistic and reliable forecast of the probability of finishing the project on time.

Ultimately, this thesis delivers an improved version of the mitigation controller that provides project managers with a higher degree of confidence. The resulting tool is a more trustworthy instrument for project management, as its recommendations are assured to be implementable within the available resources.

7.2. Further development

Even though a significant contribution has been made in this master thesis, there are still some aspects to improve in the mitigation controller. Some key areas for future work are the following:

- **Resource-capacity relationship:** Future research could improve the realism of the model by replacing the linear relationship between mitigation capacity and resource requirements. The current linear assumption, while computationally simple, is an oversimplification.
  To address this limitation, a dedicated study could investigate the relationship between mitigation capacity and resource consumption for different resource types, with the goal of deriving specific functions that describe these relationships. These new functions could then be incorporated into the mitigation controller model in a similar way to the current model but differentiating for each type of resource.
  This level of detailed modeling was beyond the scope of the current thesis due to the lack of existing research on the functions and the complexity to model them. For example, in the case of human resources, to increase the mitigation capacity extra workers are needed, which is aligned with a near-linear relationship. However, beyond a certain point, effects such as workspace congestion may reduce their efficiency and consequently the mitigation capacity. Moreover, a low number of workers could also significantly affect the mitigation capacity and not in a linear way because of the effect of workers' fatigue. This behavior is complex to model, but it would allow the model to better reflect real projects.
- **Incorporation of activities resources:** Since the focus of the tool is to determine the optimal mitigation strategy, this research only included the resources required by mitigation measures. However, a valuable extension would be to incorporate the resources demanded by activities and to allow the share of resources.
  From a modeling perspective, the implementation is straightforward. It could be done by creating an activity-resource matrix, similar to the existing one for mitigations, and defining extra constraints. A significant advantage is that the current model already tracks the start and finish times of activities as part of the optimization, which facilitates the implementation of the activities' renewable resource constraints.

However, two significant challenges need to be addressed. The primary barrier is the extensive data input required from the user. The model would need detailed resource consumption data for every activity which can be a substantial data collection effort. A second challenge is modeling the relationship between the duration of activities and resource consumption. As with mitigation measures, assuming a linear relationship is a simplification.

- **Precedence relationships:** The current mitigation controller only considers strict finish-to-start precedence relationships between activities. The tool could be further developed to support all types of logical dependencies, that is, to also allow: finish-to-finish, start-to-finish and start-to-start relationships. Also, incorporating lead and lag times into these relationships could be a relevant improvement, creating a more realistic representation of project schedules.

  While this extension would require simple modifications to the data input format, the primary challenge lies within the mathematical optimization. Since the start and finish times of activities are determined within the optimization, the constraints that define these variables would need to be reformulated. This would involve two key changes. Firstly, the simple binary precedence matrix would need to be redesigned to allow all types of precedence and lead or lag times. Secondly, the model would need to dynamically apply the correct equation for each activity's start and finish times based on the specific type of relationship it has with each of its predecessors.
- **Multi-project management:** Considering that sometimes projects share resources between each other, future research could focus on extending the Mit-C model to simultaneously analyze a portfolio of projects.

  The implementation would require extending the data model to handle multiple independent projects (schedules, risks, and mitigation measures) simultaneously. The key would be to keep the previous type of data separate while restraining all projects to a shared set of resource constraints. The current constraints would probably need to be reformulated since at the moment only consider the mitigation measures related to a single project. Moreover, there is another bigger challenge: defining the optimization goal. The model would probably need to handle project prioritization. This would require a new objective function that can weigh the relative importance of the projects and make trade-offs between them. For this purpose, the IMAP methodology could be very beneficial.

References

- Chaudhary, A., & Meshram, S. (2025). Comparison of Various Techniques for Project Scheduling under Resource Constraints. *International Journal of Engineering Trends and Technology*, *73*(2), 57–72. https://doi.org/10.14445/22315381/ijett-v73i2p106
- Forrester, R. J., & Waddell, L. A. (2022). Modeling using logical constraints. In Encyclopedia of Optimization (pp. 1–5). https://doi.org/10.1007/978-3-030-54621-2_834-1
- Hartmann, S., & Briskorn, D. (2009). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, *207*(1), 1–14. https://doi.org/10.1016/j.ejor.2009.11.005
- Hartmann, S., & Briskorn, D. (2021). An updated survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, *297*(1), 1–14. https://doi.org/10.1016/j.ejor.2021.05.004
- Hillier, F. S., & Lieberman, G. J. (2001). Introduction to Operations Research. McGraw-Hill Companies.
- Kammouh, O., Nogal, M., Binnekamp, R., & Wolfert, A. R. M. (2021). Mitigation Controller: Adaptive simulation approach for planning control measures in large construction projects. *Journal of Construction Engineering and Management*, *147*(8). https://doi.org/10.1061/(asce)co.1943-7862.0002126
- Kammouh, O., Nogal, M., Binnekamp, R., & Wolfert, A. (2022). Dynamic control for construction project scheduling on-the-run. *Automation in Construction*, *141*, 104450. https://doi.org/10.1016/j.autcon.2022.104450
- Khajesaeedi, S., Sadjadi, S. J., Barzinpour, F., & Moghaddam, R. T. (2024). Resource-constrained project scheduling problem: Review of recent developments. *Journal of Project Management*, *10*(1), 1–26. https://doi.org/10.5267/j.jpm.2024.12.002
- Khalifé, A. (2022). *Mitigation Controller: A Multi-Criteria Optimization & Simulations Approach to Keep Construction Projects Within Budget*. https://resolver.tudelft.nl/uuid:dc1c6dc1-52a5-4f2e-85cd-b0d2c574172c
- Luenberger, D. G., & Ye, Y. (2021). Linear and nonlinear programming. Springer. https://doi.org/10.1007/978-3-030-85450-8
- Manoj Philip, A. (2022). *“MitC-GERT”-An alternate network distribution in mitigation controller*. https://resolver.tudelft.nl/uuid:fe4d6d49-521f-46c9-9cd6-8b4d5a9e7010
- Nocedal, J., & Wright, S. J. (2006). Numerical Optimization. Springer Science & Business Media.
- Ramos, A. S., Miranda-Gonzalez, P. A., Nucamendi-Guillén, S., & Olivares-Benitez, E. (2023). A Formulation for the Stochastic Multi-Mode Resource-Constrained Project Scheduling Problem Solved with a Multi-Start Iterated Local Search Metaheuristic. Mathematics, 11(2), 337. https://doi.org/10.3390/math11020337
- Rivera, A., Le, N., Kashiwagi, J., & Kashiwagi, D. (2017). Identifying the global performance of the construction industry. Journal for the Advancement of Performance Information and Value, 8(2), 7. https://doi.org/10.37265/japiv.v8i2.61
- Schwindt, C., & Zimmermann, J. (2015). Handbook on Project Management and Scheduling Vol.1. In Springer eBooks. https://doi.org/10.1007/978-3-319-05443-8
- Teuber, L. G., J, V. H. H., & Wolfert, A. R. M. (2024a). Advancing strategic planning and dynamic control of complex projects. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2408.12422
- Van Gunsteren, L., Binnekamp, R., & De Graaf, R. (2011). Stakeholder-oriented project management: Tools and concepts. IOS Press. http://ebookcentral.proquest.com/lib/delft/detail.action?docID=682729
- Williams, H. P. (2013). Model building in mathematical programming. John Wiley & Sons.

Appendix A: MitC subsequent developments

Mit-C with budget optimization and impacts' multi-criteria assessment

As introduced earlier, Khalifé (2022) further develops the Mit-C by optimizing the project's budget instead of the duration. Moreover, the author incorporates a "multi-criteria assessment of negative impacts". The process for finding the best mitigation measures is very similar to the first model, the main changes are done in the optimization equations.

The objective function is modified to only two terms. The first term, like the Kammouh et al. (2021) model, searches for the mitigation measures strategy that reduces the project cost to be within the budget while maximizing effectiveness. In Khalifé (2022), the effectiveness is defined as "the ratio of the cost reduction and the aggregated average of negative impact of each mitigation measure". The author explains that mitigation measures besides carrying cost reduction, they also have negative effects. Thus, he introduces the concept of "average of negative impact" as an aggregated average computed using preference-based modelling software to take into account the effect of the measures under specific criteria (e.g. the author uses for the studied case: time delays, environmental impact and noise disturbance).

The second term of the objective function, in a similar way to the third term of the original model, tries to minimize the cost overrun when the mitigation measures are not enough to reach the project's budget.

Mit-C with activities correlation and contractual project completion performance scheme

One of the limitations that the Mit-C introduced by Kammouh et al. (2021) has is that it only considers the paths with a duration longer than the target one, in other words, those paths with delay. Thus, the original model is not able to consider the possible benefits of finishing the project before the target time. Kammouh et al. (2022) further develops the original model to include a different "contractual project completion performance scheme" in which penalties and rewards for the project's duration are considered. In order to do this, the objective function is changed since now the aim is to reduce the total net cost. In other words, the objective is to find the mitigation strategy that minimizes the net cost which includes the cost of the mitigation measures and the penalties or rewards. The problem as stated is non-linear, but some modifications are made to the equations to transform it into a linear problem and, consequently, be able to find the global optimal solution.

Another limitation of the original model is that it assumes that activities' durations are independent from each other. Kammouh et al. (2022) explain that construction activities can be stochastically correlated since they are exposed to similar conditions, such as the weather, and these shared factors introduce uncertainty in the activities' duration. Thus, an activity's duration can have two types of uncertainty. First, there is an independent uncertainty which only influences the specific activity (this was included in the original Mit-C model). Secondly, there is a shared uncertainty which affects several activities. Consequently, the authors propose to compute the random variable "activity duration" as the summation of the random variable "uncorrelated duration" and the random variable "shared duration".

Mit-C - GERT

The model of Kammouh et al. (2022) was later further developed by Manoj Philip (2022). This new research focused in studying the influence of the project network structure on the mitigation controller. The main modification implemented by Manoj Philip (2022) is the use of the Graphical Evaluation and Review Technique (GERT) instead of the PERT. As explained by the authors, this technique allows to tackle the limitations of the PERT. For instance, while PERT follows a deterministic branching, GERT follows a probabilistic one. Also, while in the original approach the activities are executed in a linear way, the GERT allows to repeat activities by "feedback loops". The authors state that this model could be applied to more complex projects but also mention that it requires higher computational power and to be knowledgeable on the GERT concept.

Odycon

Further research was done by Teuber et al. (2024) based on Kammouh et al.(2022) model. The authors present a new model called Open Design and Dynamic Control (Odycon), "a pure a-priori stochastic simulation & optimization methodology integrating the capability of the project (technical domain), the human goal-oriented behavior (human domain), and the association of stakeholder-oriented behavior (social domain)" (Teuber et al., 2024). Odycon integrates the MCS with the Integrative Maximization of Aggregated Preferences (IMAP) optimization method. As explained by the authors, the latter method is implemented to integrate the interests of the stakeholders with the technical design and, consequently, reach a "best-fit for common purpose project management" (Teuber et al., 2024). This means that with this approach the interest of all the stakeholders are taken into account. Also, considering that stakeholders may have more than one interest, Odycon allows the inclusion of several objectives, broadening the original mitigation controller tool which only focuses on the duration or the budget of the project.

The concept of Odycon works in a similar way to Mit-C. One of the main differences is that Teuber et al. (2024a) propose two ways of using the tool, as project planning optimization or as project control optimization. The user needs to choose whether to use it for planning or for control, since this defines the types of variables. Nevertheless, the focus of this proposal is on project control, thus the explanation will continue with this approach.

As mentioned before, the optimization is carried out using the IMAP method. Teuber et al. (2024a) explain that the aim is to find the highest aggregated score to define the optimal measures. In order to do this, first it is necessary to define the objective functions, the preference functions of the stakeholders and the weights. To compute the aggregated score, the first step is to calculate the values of the objective functions and the related preference scores using the preference functions, then these scores per stakeholder are aggregated using the weights and finally the overall aggregated preference score is computed employing the weighted least squares method.

Appendix B: RCPSP

This appendix provides an overview of the types of resources encountered in the Resource Constrained Project Scheduling Problem (RCPSP) literature.

Chaudhary and Meshram (2025) review project scheduling techniques in the context of the resource-constrained problem, but first they identify the most important type of resource constraints. According to the authors the constraints can be of the following types: human resources (workers availability and skills), equipment availability, material constraints, financial constraints (budget limitations or funding availability) and IT resources.

Based on the research done by Hartmann and Briskorn (2009), the classification of resources is a little bit different. They recognize various types of resources and, thus, different variants of the RCPSP. Moreover, the authors state that the original RCPSP considers only renewable resources, that is, resources that are “available in each period with its full capacity” (Hartmann & Briskorn, 2009). However, there are variants that consider other scenarios.

One of these cases mentioned by the authors, and perhaps the most relevant to the current development, is the Multi-Mode scheduling problem (MRCPSP or MMRCPSP). This problem may consider different types of resources: renewable, non-renewable and doubly-constrained resources. Hartmann and Briskorn (2009) explain that the non-renewable resources are those that have a capacity for the whole project and, thus, cannot be restored after a period is finalized, while the doubly-constrained ones are “limited both for each period and for the whole project” (Hartmann & Briskorn, 2009). The authors also highlight that doubly-constrained resources can be included using a renewable and a non-renewable resource. Moreover, some researchers consider a type of resource in the RCPSP problem that is similar to the latter one: partially renewable resources. Hartmann and Briskorn (2021) define this type of resources as those that have a “set of subsets of periods” and that the capacity is available in each subset.

Furthermore, Hartmann and Briskorn (2009) point out that while the base RCPSP considers discrete resources, such as labor and equipment, some authors focus on solving the problem by implementing continuous resources, such as material.

Another type of resource found in the literature is cumulative resources. Typical of production processes, Hartmann and Briskorn (2009) describes this type of resource as those that can be both used and generated by an activity. In the construction industry, this variant of the RCPSP could help to get insight into how to consider resource constraints in prefab construction projects.

Some other literature focuses on solving the RCPSP taking into account time-dependent resources. As the name implies, the resource availability changes as time passes. These may sound similar to the non-renewable resources. However, the availability of non-renewable resources changes because they are consumed, but the one of time-dependent resources varies for other reasons. Hartmann and Briskorn (2009) suggest that this type of resource is useful for taking into account equipment maintenance or workers vacations.

Hartmann and Briskorn (2021) updated the research on the RCPSP variants and included the Multiple-Skills resources (MS-RCPSP). The authors explain that in this type of problem activities need resources with specific skills while the resources have one or more skills, and the objective is not only to minimize the makespan but also to allocate correctly the resources. This type of problem can be significant in a construction project since it can reflect the different skills of workers.

In the current development, the most relevant type of resources to be considered are the human resources, equipment and materials, following the Chaudhary and Meshram (2025) classification. The financial constraint is somewhat already taken into account in the existing tool, and the IT resource is not fully relevant for the model. Following the Hartmann and Briskorn (2021) classification, the type of resources that will be included in the development are renewable and non-renewable. Other resource variants mentioned above, such as multiple-skill resources, may be relevant in the construction industry. However, they would probably increase the complexity of the tool, thus, their inclusion should be investigated in future research.

Appendix C: Case study data

Activity ID	Activity description	Optimistic	Most-Likely	Pessimistic	Predecessor activity
1	Start date	0	0	0	
2	Design phase	58	65	102	1
3	Earthworks	38	46	68	2
4	Foundation	60	69	102	2
5	Building internal metal structure	74	89	128	4
6	Building perimetral walls	86	113	150	3,4
7	Roof set up	112	146	192	5,6
8	Building slab on grade	39	43	56	5,6
9	Mechanical installations	125	153	203	3,4
10	Electrical installations	119	145	193	3,4
11	Office area finishes	103	120	172	5,6
12	Exterior works	74	86	123	5,6
13	Delivery	0	0	0	12

Table 1: Case study activities of validation exercise 1

Activity ID	Activity description	Optimistic	Most-Likely	Pessimistic	Predecessor activity
1	Start date	0	0	0	
2	Design Phase	52	65	102	1
3	EARTHWORKS - Soil Striping	5	7	10	2
4	EARTHWORKS - Cut and Scarification	12	15	23	3
5	EARTHWORKS - Filling and Compaction	30	40	60	4
6	EARTHWORKS - Crushed Base	9	12	18	5
7	EARTHWORKS - Sand Filling for Casting Beds	5	7	10	10,6
8	FOUNDATION - Spread Footings	17	22	33	5
9	FOUNDATION - Continuos Footing	18	22	34	5
10	TILT UP WALLS - Concrete Block For Anchoring	13	16	24	5
11	TILT UP WALLS - Casting Beds	14	17	26	10,6
12	TILT UP WALLS - Rebar Enabling	18	23	35	10,6
13	TILT UP WALLS - Tilt Up Pouring	24	30	45	10,6
14	TILT UP WALLS - Lifting	17	21	32	13
15	TILT UP WALLS - Sack and Patch	34	42	63	14
16	TILT UP WALLS - Painting	34	43	65	14
17	SLAB ON GRADE - Concrete Pouring	18	22	33	25,6
18	SLAB ON GRADE - Joint Sealing	14	18	27	17
19	METAL STRUCTURE - Main Structure	48	60	90	5
20	METAL STRUCTURE - Joists	64	80	120	5
21	METAL STRUCTURE - Central Structure	40	50	75	5
22	METAL STRUCTURE - Perimeter Structure	41	50	76	13,19
23	METAL STRUCTURE - Paint	48	60	90	21
24	ROOFING SYSTEM - Insulation	17	21	32	23
25	ROOFING SYSTEM - KR18 Metal Sheet Installation	16	21	31	24
26	ROOFING SYSTEM - Skylights	24	30	45	25
27	ROOFING SYSTEM - Accessories and Detailing	22	27	42	25,26
28	EQUIPMENT AND DOORS - Dock Levelers	30	38	57	17
29	EQUIPMENT AND DOORS - Ramp Doors	12	15	22	16
30	EQUIPMENT AND DOORS - Man Doors	13	15	23	16
31	EQUIPMENT AND DOORS - Concrete Ramps	26	30	45	17
32	EQUIPMENT AND DOORS - Bollards	5	7	10	17
33	FIRE PROTECTION SYSTEM - FPS Installations	36	45	68	25
34	FIRE PROTECTION SYSTEM - Fire Alarm System	21	27	41	33
35	ELECTRICAL INSTALLATIONS - Main Feeders	40	50	75	25
36	ELECTRICAL INSTALLATIONS - Lighting	38	47	71	35
37	ELECTRICAL INSTALLATIONS - Outlets	37	47	70	35
38	PLUMBING - Sanitary System	17	21	32	14
39	PLUMBING - Domestic Water Installations	24	30	45	25
40	EXTERIORS - Pavement Construction	44	55	83	22
41	EXTERIORS - Landscape and Exterior Rooms	12	15	23	40
42	Delibery	0	0	0	41

Table 2: Case study activities of validation exercise 2

Risk event ID	Risk event description	Minimum	Most likely	Maximum	Affected activities	Risk probability
1	Design rejection	8	12	20	2	0.05
2	Design errors (e.g.:collisions)	5	8	15	2	0.1
3	Unexpected groundwater table discovery	10	18	30	3	0.1
4	Soil supply shortage	7	12	18	3	0.15
5	Material supply shortage	8	13	22	4	0.2
6	Material supply shortage	8	13	22	6	0.2
7	Material supply shortage	8	13	22	8	0.2
8	Steel elements supply shortage	10	18	35	5	0.15
9	Steel elements supply shortage	10	18	35	7	0.15
10	Wall collapse	14	24	34	6	0.02
11	Crane/heavy equipment breakdown	5	10	20	6	0.1
12	Personnel heat stroke	3	7	12	7	0.15
13	Personnel fall accident	5	12	25	7	0.08
14	Incorrect cement quality	7	14	25	4	0.05
15	Incorrect cement quality	7	14	25	6	0.05
16	Incorrect cement quality	7	14	25	8	0.05
17	Power outage	1	3	7	9	0.1
18	External power infrastructure connection issues	7	12	20	10	0.12
19	Weather conditon	4	7	13	3	0.3
20	Weather conditon	4	7	13	4	0.3
21	Weather conditon	4	7	13	5	0.3
22	Weather conditon	4	7	13	6	0.3
23	Weather conditon	4	7	13	7	0.3
24	Weather conditon	4	7	13	12	0.3

Table 3: Case study risks of validation exercise 1

Risk event ID	Risk event description	Minimum	Most likely	Maximum	Affected activities	Risk probability
1	Design rejection	8	12	20	2	0.05
2	Design errors (e.g.:collisions)	5	8	15	2	0.1
3	Unexpected groundwater table discovery	10	18	30	3	0.1
4	Soil supply shortage	7	12	18	5	0.15
5	Soil supply shortage	7	12	18	7	0.15
6	Material supply shortage	8	13	22	8	0.2
7	Material supply shortage	8	13	22	9	0.2
8	Material supply shortage	8	13	22	10	0.2
9	Material supply shortage	8	13	22	13	0.2
10	Material supply shortage	8	13	22	17	0.2
11	Steel elements supply shortage	10	18	35	19	0.15
12	Steel elements supply shortage	10	18	25	21	0.15
13	Steel elements supply shortage	10	18	25	22	0.15
14	Wall collapse	14	24	34	14	0.02
15	Crane/heavy equipment breakdown	5	10	20	14	0.1
16	Personnel heat stroke	3	7	12	24	0.15
17	Personnel heat stroke	3	7	12	25	0.15
18	Personnel heat stroke	3	7	12	26	0.15
19	Personnel fall accident	5	12	25	24	0.08
20	Personnel fall accident	5	12	25	25	0.08
21	Personnel fall accident	5	12	25	26	0.08
22	Incorrect cement quality	7	14	25	8	0.05
23	Incorrect cement quality	7	14	25	9	0.05
24	Incorrect cement quality	7	14	25	13	0.05
25	Incorrect cement quality	7	14	25	17	0.05
26	Power outage	1	3	7	33	0.1
27	Power outage	1	3	7	34	0.1
28	Power outage	1	3	7	35	0.1
29	Power outage	1	3	7	36	0.1
30	Power outage	1	3	7	37	0.1
31	External power infrastructure connection issues	7	12	20	35	0.12

Table 4: Case study risks of validation exercise 2

SUF ID	SUF description	Minimum	Most likely	Maximum	Relations
1	Weather condition	-15	0	25	3,4,5
2	Weather condition	-15	0	25	8,9
3	Weather condition	-15	0	25	11,13
4	Weather condition	-15	0	25	19,20,21
5	Weather condition	-15	0	25	24,25,26

Table 5: Shared uncertainty factors of validation exercise 2

Mitigation ID	Mitigation measure	Minimum time	Most likely time	Maximum time	Relations	Dependency factor (eta)	Minimum cost	Most likely cost	Maximum cost
1	Additional heavy earthworking fleet	10	15	20	3	0.5	62500.00	75000.00	87500.00
2	Extra personnel	15	22	30	4	0.7	62181.82	80000.00	100363.64
3	Add large crane for material lifting	11	15	19	5	0.5	65000.00	75000.00	85000.00
4	Add aerial lifts for personnel access	8	12	16	5	0.5	10000.00	12000.00	14000.00
5	Extra steel erection crew	18	25	32	5	0.7	100500.00	125000.00	149500.00
6	Extra general workers	15	25	35	6	0.7	79200.00	110000.00	140800.00
7	Add large crane for material lifting	10	14	18	6	0.5	60000.00	70000.00	80000.00
8	Employ high-early-strength concrete	8	12	16	6	0.2	42000.00	45000.00	48000.00
9	Extra steel erection crew	25	35	45	7	0.7	140000.00	175000.00	210000.00
10	Extra general workers	15	22	30	7	0.7	34977.27	45000.00	56454.55
11	Extra general workers	8	12	16	8	0.7	30666.67	40000.00	49333.33
12	Employ laser drying equipment	4	6	8	8	0.2	14000.00	15000.00	16000.00
13	Employ high-early-strength concrete	5	8	11	8	0.2	83250.00	90000.00	96750.00
14	Add aerial lifts for personnel access	18	25	32	9	0.5	21500.00	25000.00	28500.00
15	Extra general workers	22	30	40	9	0.7	48800.00	60000.00	74000.00
16	Add aerial lifts for personnel access	18	25	32	10	0.5	21500.00	25000.00	28500.00
17	Extra general workers	22	30	40	10	0.7	48800.00	60000.00	74000.00
18	Employ high-early-strength concrete	10	14	18	12	0.2	80142.86	85000.00	89857.14
19	Employ laser drying equipment	5	8	11	12	0.2	16650.00	18000.00	19350.00

Table 6: Case study mitigation measures of validation exercise 1

		Objective Time					Objective cost		
Mitigation ID	Mitigation measure	Minimum time	Most likely time	Maximum time	Relations	Dependency factor (beta)	Minimum cost	Most likely cost	Maximum cost
1	Additional heavy earthworking fleet	2	3	4	3	0.5	20000.00	24000.00	28000.00
2	Additional heavy earthworking fleet	4	6	8	4	0.5	40000.00	48000.00	56000.00
3	Additional heavy earthworking fleet	11	15	19	5	0.5	104000.00	120000.00	136000.00
4	Add large crane for material lifting	7	9	11	14	0.5	64000.00	72000.00	80000.00
5	Add large crane for material lifting	11	15	19	19	0.5	104000.00	120000.00	136000.00
6	Add large crane for material lifting	7	10	13	21	0.5	68000.00	80000.00	92000.00
7	Add large crane for material lifting	7	10	13	22	0.5	68000.00	80000.00	92000.00
8	Add aerial lifts for personnel access	6	8	10	19	0.5	8750.00	10000.00	11250.00
9	Add aerial lifts for personnel access	7	10	13	20	0.5	10625.00	12500.00	14375.00
10	Add aerial lifts for personnel access	5	7	9	21	0.5	7500.00	8750.00	10000.00
11	Add aerial lifts for personnel access	5	7	9	22	0.5	7500.00	8750.00	10000.00
12	Add aerial lifts for personnel access	6	8	10	23	0.5	8750.00	10000.00	11250.00
13	Add aerial lifts for personnel access	3	4	5	24	0.5	4375.00	5000.00	5625.00
14	Add aerial lifts for personnel access	3	4	5	25	0.5	4375.00	5000.00	5625.00
15	Add aerial lifts for personnel access	4	5	6	26	0.5	5625.00	6250.00	6875.00
16	Add aerial lifts for personnel access	4	5	6	27	0.5	5625.00	6250.00	6875.00
17	Add aerial lifts for personnel access	5	7	9	33	0.5	7500.00	8750.00	10000.00
18	Add aerial lifts for personnel access	5	7	9	36	0.5	7500.00	8750.00	10000.00
19	Employ high-early-strength concrete	4	5	6	8	0.2	43200.00	45000.00	46800.00
20	Employ high-early-strength concrete	4	5	6	9	0.2	43200.00	45000.00	46800.00
21	Employ high-early-strength concrete	4	6	8	13	0.2	50400.00	54000.00	57600.00
22	Employ high-early-strength concrete	4	6	8	17	0.2	74666.67	80000.00	85333.33
23	Employ high-early-strength concrete	5	7	9	40	0.2	99000.00	105000.00	111000.00
24	Employ high-early-strength concrete	2	3	4	41	0.2	14000.00	15000.00	16000.00
25	Employ laser drying equipment	3	4	5	13	0.2	11400.00	12000.00	12600.00
26	Employ laser drying equipment	3	4	5	17	0.2	11400.00	12000.00	12600.00
27	Employ laser drying equipment	4	6	8	40	0.2	16800.00	18000.00	19200.00
28	Employ laser drying equipment	2	3	4	41	0.2	8400.00	9000.00	9600.00
29	Extra steel erection crew	14	18	23	19	0.7	76000.00	90000.00	107500.00
30	Extra steel erection crew	18	24	30	20	0.7	99000.00	120000.00	141000.00
31	Extra steel erection crew	11	15	19	21	0.7	61000.00	75000.00	89000.00
32	Extra steel erection crew	11	15	19	22	0.7	61000.00	75000.00	89000.00
33	Extra earthworks crew	2	2	3	3	0.7	6000.00	6000.00	8100.00
34	Extra earthworks crew	3	4	5	4	0.7	9900.00	12000.00	14100.00
35	Extra earthworks crew	9	12	15	5	0.7	29700.00	36000.00	42300.00
36	Extra earthworks crew	3	4	5	6	0.7	9900.00	12000.00	14100.00
37	Extra earthworks crew	2	2	3	7	0.7	6000.00	6000.00	8100.00
38	Extra concrete crew	5	7	9	8	0.7	22400.00	28000.00	33600.00
39	Extra concrete crew	5	7	9	9	0.7	22400.00	28000.00	33600.00
40	Extra concrete crew	4	5	6	10	0.7	17200.00	20000.00	22800.00
41	Extra concrete crew	4	5	6	11	0.7	17200.00	20000.00	22800.00
42	Extra concrete crew	7	9	11	13	0.7	30400.00	36000.00	41600.00
43	Extra concrete crew	5	7	9	17	0.7	22400.00	28000.00	33600.00
44	Extra general workers	9	12	15	15	0.7	19800.00	24000.00	28200.00
45	Extra general workers	9	12	15	16	0.7	19800.00	24000.00	28200.00
46	Extra general workers	12	15	19	23	0.7	25800.00	30000.00	35600.00
47	Extra general workers	5	6	8	24	0.7	10600.00	12000.00	14800.00
48	Extra general workers	5	6	8	25	0.7	10600.00	12000.00	14800.00
49	Extra general workers	7	9	11	26	0.7	15200.00	18000.00	20800.00
50	Extra general workers	6	8	10	27	0.7	13200.00	16000.00	18800.00
51	Extra general workers	9	11	14	28	0.7	19200.00	22000.00	26200.00
52	Extra general workers	3	4	5	29	0.7	6600.00	8000.00	9400.00
53	Extra general workers	3	4	5	30	0.7	6600.00	8000.00	9400.00
54	Extra general workers	2	2	3	32	0.7	5000.00	5000.00	6750.00

Table 7: Case study mitigation measures of validation exercise 2

	General workers	Welding personnel	Lifting equipment	Crane	Laser-drying equipment
Mitigation ID	resource_1	resource_2	resource_3	resource_4	resource_5
1	0	0	0	0	0
2	1	0	0	0	0
3	0	0	0	1	0
4	0	0	1	0	0
5	0	1	0	0	0
6	1	0	0	0	0
7	0	0	0	1	0
8	0	0	0	0	0
9	0	1	0	0	0
10	1	0	0	0	0
11	1	0	0	0	0
12	0	0	0	0	1
13	0	0	0	0	0
14	0	0	1	0	0
15	1	0	0	0	0
16	0	0	1	0	0
17	1	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	1
Capacity	2	1	1	1	1

Table 8: Case study renewable resources of validation exercise 1

	Earthworks personnel	Concrete personnel	Welding personnel	General workers	Earthworks equipment fleet	Crane	Lifting equipment	Laser-drying equipment
Mitigation ID	resource_1	resource_2	resource_3	resource_4	resource_5	resource_6	resource_7	resource_8
1	0	0	0	0	1	0	0	0
2	0	0	0	0	1	0	0	0
3	0	0	0	0	1	0	0	0
4	0	0	0	0	0	1	0	0
5	0	0	0	0	0	1	0	0
6	0	0	0	0	0	1	0	0
7	0	0	0	0	0	1	0	0
8	0	0	0	0	0	0	1	0
9	0	0	0	0	0	0	1	0
10	0	0	0	0	0	0	1	0
11	0	0	0	0	0	0	1	0
12	0	0	0	0	0	0	1	0
13	0	0	0	0	0	0	1	0
14	0	0	0	0	0	0	1	0
15	0	0	0	0	0	0	1	0
16	0	0	0	0	0	0	1	0
17	0	0	0	0	0	0	1	0
18	0	0	0	0	0	0	1	0
19	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0
21	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0
23	0	0	0	0	0	0	0	0
24	0	0	0	0	0	0	0	0
25	0	0	0	0	0	0	0	1
26	0	0	0	0	0	0	0	1
27	0	0	0	0	0	0	0	1
28	0	0	0	0	0	0	0	1
29	0	0	1	0	0	0	0	0
30	0	0	1	0	0	0	0	0
31	0	0	1	0	0	0	0	0
32	0	0	1	0	0	0	0	0
33	1	0	0	0	0	0	0	0
34	1	0	0	0	0	0	0	0
35	1	0	0	0	0	0	0	0
36	1	0	0	0	0	0	0	0
37	1	0	0	0	0	0	0	0
38	0	1	0	0	0	0	0	0
39	0	1	0	0	0	0	0	0
40	0	1	0	0	0	0	0	0
41	0	1	0	0	0	0	0	0
42	0	1	0	0	0	0	0	0
43	0	1	0	0	0	0	0	0
44	0	0	0	1	0	0	0	0
45	0	0	0	1	0	0	0	0
46	0	0	0	1	0	0	0	0
47	0	0	0	1	0	0	0	0
48	0	0	0	1	0	0	0	0
49	0	0	0	1	0	0	0	0
50	0	0	0	1	0	0	0	0
51	0	0	0	1	0	0	0	0
52	0	0	0	1	0	0	0	0
53	0	0	0	1	0	0	0	0
54	0	0	0	1	0	0	0	0
Capacity	2	3	2	3	1	1	5	2

Table 9: Case study renewable resources of validation exercise 2

	High-early-strength concrete
Mitigation ID	**resource_1**
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	1000
9	0
10	0
11	0
12	0
13	2165
14	0
15	0
16	0
17	0
18	2000
19	0
Capacity	4500

Table 10: Case study non-renewable resources of validation exercise 1

	High-early-strength concrete
Mitigation ID	resource_1
1	0
2	0
3	0
4	0
5	0
6	0
7	0
8	0
9	0
10	0
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	350
20	400
21	1000
22	2165
23	2000
24	250
25	0
26	0
27	0
28	0
29	0
30	0
31	0
32	0
33	0
34	0
35	0
36	0
37	0
38	0
39	0
40	0
41	0
42	0
43	0
44	0
45	0
46	0
47	0
48	0
49	0
50	0
51	0
52	0
53	0
54	0
Capacity	4500

Table 11: Case study non-renewable resources of validation exercise 2

Appendix D: Constraints violations

Resource	Number of constraints
High-early-strength concrete	0
General workers	24
Welding personnel	374
Lifting equipment	447
Crane	885
Laser-drying equipment	0
Total	1730

Table 1: Resource-constraint non-compliance (first validation)

Resource	Number of constraints
High-early-strength concrete	0
Earthworks personnel	1501
Concrete personnel	0
Welding personnel	0
General workers	0
Earthworks equipment fleet	812
Crane	372
Lifting equipment	0
Laser-drying equipment	0
Total	2685

Table 2: Resource-constraint non-compliance (second validation)