Fall 1-31-2025

Advanced Construction Scheduling: Capturing Conditional and Stochastic Relationships

Amr Elsabagh
The American University in Cairo AUC, amr.elsabagh@aucegypt.edu

The American University in Cairo

Graduate Studies

Advanced Construction Scheduling: Capturing Conditional and Stochastic Relationships

A THESIS SUBMITTED BY

Amr ElSabagh

TO THE

Department of Construction Engineering

SUPERVISED BY

Dr. Ibrahim Abotaleb
Associate Professor, Construction Engineering Dep.,AUC

20 August 2024

in partial fulfillment of the requirements for the degree of
Master of Science in Construction Engineering

Declaration of Authorship

I, Amr ElSabagh, declare that this thesis titled, "Advanced Construction Scheduling: Capturing Conditional and Stochastic Relationships "and the work presented in it are my own. I confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where I have consulted the published work of others, this is always clearly attributed.
- Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work.
- I have acknowledged all main sources of help.
- Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself.

Signed:

Amr Elsabagh

Date:

20th August 2024

Abstract

Proper planning is a key factor in all business endeavors. This is especially important in the construction industry as a project is sensitive to countless factors that affect the project's time and cost. This created the need for agile and flexible schedules for close monitoring and dynamic planning for construction projects. The currently prevailing scheduling techniques are CPM and PERT, which professionals in the field widely use. These methods create static schedules that are vulnerable to any changes in the schedule logic, which often happens during construction projects. In addition, these Scheduling techniques do not capture the uncertainties and complex relationships well, which makes them susceptible to cost and schedule overruns. Usually, to overcome this, decision-makers develop a recovery plan that takes time and effort, and even these plans typically take tremendous effort to develop and implement. Instead, if decision-makers had access to a tool that provided them with different scenarios in order to prepare proper risk response techniques that provided dynamic responses to emergencies, this would save a lot of time and resources. In the literature, limited work was found to provide a scheduling method that considers different execution scenarios and the corresponding implications.

Hence, there is a need to develop an innovative method which can increase flexibility and provide adaptation and agility to schedules. The goal of this research is to develop a novel scheduling method based on conditional relationships and stochastic inter-activity associations, which tackles the shortcomings of current deterministic and limited stochastic scheduling techniques. This research utilizes the discrete event simulation on AnyLogic software to model the behavior of activities in stochastic networks to determine the overall project completion. In this model, each activity is simulated by an agent that has certain parameters duration, probability of occurrence, predecessor, and successor. Those agents are then evaluated in the constructed network and this process is repeated for 100 runs. The findings of the model have been compared against the findings of another simulation technique using a case study for historical rehabilitation. While other stochastic models estimated the project duration to be 24.14 days and deterministic models estimated 20.55 days, the model developed in this study estimated a project completion time of 26.4 days, which is the closest to the actual project duration of 35 days.

This research has various potential applications, both strategic and project-specific. Strategically,

they can offer top management a decision support tool, supplying sufficient information for making long-term strategic decisions and preparing for different scenarios. At the project level, this tool enables project managers to simulate the complexities of construction projects, which often necessitate a proactive approach. This allows project managers to anticipate different scenarios from the project's outset and develop mitigation plans accordingly.

Acknowledgments

All praise be to Allah who granted me the knowledge and guidance to complete this work. I would like to express my gratitude for the guidance of my supervisor Dr. Ibrahim Abotaleb who provided me encouragement and support throughout the journey. I also thank the esteemed committee Dr Khaled Nassar and Dr Dina Atef for their insight and feedback which has been crucial to enhance the work. Finally, I would like to extend my appreciation to my wife, parents and friends who has been supportive every step of the way and dedicate this work to my late grandparent who left us but their support is never forgotten every step of the way.

Contents

List of Figures

List of Tables

List of Abbreviations

AEC	Architecture, Engineering and Construction
AOA	Activity on Arrow
AON	Activity on Node
CP	Critical Path
CPM	Critical Path Method
DES	Discrete Event Simulation
EFT	Early Finish Time
EST	Early Start Time
GAN	Generalized Activity on Node
GERT	Graphical Evaluation and Review Technique
TF	Total Float
LFT	Late Finish Time
LST	Late Start Time
PDM	the Precedence Diagramming Method
PERT	Program Evaluation Review Technique
PtP	Point to Point Relationship
XOR	Exclusive OR

Chapter 1

Introduction

1.1 General

The Architectural Engineering and Construction (AEC) industry faces numerous uncertainties, making project planning and management a challenging task. Several factors, including political, economic, and natural influences, add complexity to these projects (Taroun, 2014). As the typical project includes a large number of activities that usually overlap in time of execution with varied duration and frequent delays, this causes instability in schedules. The aforementioned reasons create a need to have resilient schedules that can absorb uncertainties and risks faced by projects. (Lucko et al., 2021)

The current practice is that planners have to take into account all the uncertainties in the duration of activity through techniques such as Program Evaluation and Review Technique (PERT) and Critical Path Method (CPM). For PERT, the possible duration of each activity in the project is represented using a statistical distribution that is often beta distribution (Lee, 2005). This distribution often requires the definition of three values which are the pessimistic scenario which represents the longest duration of this individual activity, the optimistic scenario, which is the shortest possible duration of this activity, and the most likely duration which is somewhere between the two aforementioned values and represents a more realistic scenario (Nasir, et al, 2003). For the CPM, this is a more deterministic approach that is widely used by professionals in the industry because of its ease of use (Lee & Arditi, 2006). The project duration is calculated by constructing dependencies between the activities within the network while assigning a deterministic value to each activity duration. Certain parameters for each activity are calculated such as early start, late start, early finish, and late finish. These parameters are then used to calculate the total float for each activity and whether this activity lies on the critical path. The total duration of the activities on the critical path represents the shortest project duration (Khodakarami, et al., 2007). These two approaches have proven to be useful for planning and controlling construction projects; because these methods enable the project team to determine the slack in activities, referred to as activity float. These techniques can also help determine the impact of delays on the overall project duration and corresponding indirect costs. However, these methods either do not take uncertainty in the structure of the network and logical sequence, into account or tackle it in terms of the duration of the project; very few address the issue of different probable paths and a simple visualization method to enable an easy understanding of the project requirements (Lee,2005).

1.2 Problem Statement

There have been different attempts to overcome the common limitations of traditional scheduling techniques. Researchers have experimented with simulation modeling, fuzzy logic, and dynamic programming to integrate new parameters that govern the relationship between activities other than time. These other parameters include but are not limited to site limitations, resource availability, and priority of activities (Katsuragawa et al., 2021). However, very few studies have tried to provide a novel technique for introducing the probability of occurrence of relationships between activities. Also, there is a limited number of attempts in the literature to provide a new way to visualize activities other than activity on arrow (AOA). Even successful attempts were able to find the total time of the project without providing detailed execution scenarios necessary for the project manager (Radziszewska et al., 2017). Reviewing the literature shows a gap in finding a novel way to capture different execution scenarios for projects on a strategic or project level either by having an overview of the project's total duration. This study will attempt to fill this gap by exploring different planning and activity visualization techniques that take into consideration stochastic relationships.

1.3 Research Goal and Objectives

The goal of this research is to enhance the current scheduling methods based on conditional relationships and stochastic inter-activity associations, which tackles the shortcomings of current deterministic and limited stochastic scheduling techniques. To achieve this goal, the following objectives are set:

i. Adopting an enhanced visualization system for the representation of activities' interdependencies, and
ii. Developing an enhanced scheduling technique that takes into consideration conditional interdependencies and probabilistic scenarios.

The above points have many potential uses that are either strategic or project-level. In terms of strategy, it can provide top management with a decision support tool that can provide enough information for decision-makers to make long-term strategic decisions such as whether to move forward with the project or divide it into phases as well as prepare for alternative scenarios. On a project level, this tool can provide a way for project managers to simulate the complex nature of construction projects that often require a proactive approach from PMs in which they are aware of different scenarios at the beginning of the project to be able to prepare mitigation plans accordingly.

1.4 Research Methodology

The followed methodology is as follows:

1. **Literature Review**: In this step, we investigate the current scheduling techniques and their limitations. We also explore how fellow researchers tackled the problem so that the gap can be identified and tackled.
2. **Model Development**: In this step, development of a new scheduling technique that will capture the true nature of construction sequencing is undertaken. Also, heuristics that express useful insights and new matrices (i.e., aggregate ES, EF, LS, LF, and float) are developed.
3. **Model Verification/Validation**: The model is thoroughly examined to identify illogical sequences and paths. The model is also validated through a case study, where the results of the developed model are compared to those of other scheduling techniques.
4. **Development of Visualization System:** A simple visualization technique will be developed to demonstrate the obtained matrices from the developed scheduling technique (AON) with logical Emitter and Receivers.

1.5 Thesis Organization

This research is composed of the following chapters:

Chapter 1 Introduction

The chapter provides an overview of the importance of proper planning, research questions, research objectives, and the methodology of research.

Chapter 2 Literature Review

This chapter explores the current and previous developments in project planning not just in the AEC but also in operational research.

Chapter 3 Model Development and Implementation

This chapter dives into the details of the modeling methodology that has been utilized in this research and demonstrates the detailed blocks of each aspect of the model to allow for a better understanding of the model.

Chapter 4 Verification and Validation of the Model (Case Study)

This chapter demonstrates the verification steps used during the development of the model as well as the application of the aforementioned model in a real-life scenario from a case study to compare the findings of the model to the real scenario to validate the model.

Chapter 5 Conclusion

This chapter concludes the findings of the research, emphasizing the contribution to the body of knowledge and paving the way for future research on this topic.

Chapter 2

Literature Review

The topic of project planning has been thoroughly investigated in the literature. However, few work has been found with regard to the nature of relationships between the activities themselves and the types of constraints, whether related to time, such as Start to Start or Finish to Start, or related to resource availability or the integration of probability paths on schedule structure. The section below illustrates the traditional methods of scheduling as well as the current direction of research in terms of new scheduling techniques that take into consideration actual site dynamics, and resource limitations. Also, this research overlaps with the area of operational research as many of the concepts needed in developing this work are inspired by the ongoing work in operational research where the visualization method proposed in this research has its roots in the area of operational research.

2.1 Traditional Scheduling techniques

The most common scheduling methods are the CPM and PERT as Katsuragawa et al., (2021) provided the history and comparison between both in their paper titled "Fuzzy linear and repetitive scheduling for construction projects". The paper highlighted that CPM can be described as a deterministic approach to determine the project's duration. This is done by adding project duration according to the dependency paths that calculate the earliest start and finish date, which is considered an optimistic scenario. This method also calculates the latest start and finish date for the pessimistic scenario. The slack of the activity duration, often referred to as activity float, is also calculated to determine the buffer for each activity and critical path. This method is a simple approach that does not capture the consistently changing project dynamics. The Program Evaluation and Review Technique (PERT), on the other hand, includes the element of uncertainty in the calculations. The US Navy developed it to provide a realistic estimate for the completion of military projects during the Cold War (MacCrimmon & Ryavec, 1962). This method captures the element of uncertainty by defining probability functions with three-point approximations of the optimistic, most likely, and pessimistic durations (Hajdu et al., 2016). The previously mentioned duration is often weighted equally to calculate the mean and standard deviation of the activity durations. On the other hand, there are certain limitations to the method. Firstly, there are underlying similarities between PERT and CPM in terms of adding durations on the network paths; if all calculations were based on either pessimistic or optimistic durations, then this would result in the maximum or minimum duration of the project (Katsuragawa et al., 2021). On the other hand, if realistic durations are used, this is analogous to the durations in the CPM. Secondly, the probable duration of each activity is given equal weights, which is not realistic as the longest durations occur most frequently in real-life projects (Alzraiee, Hani, et al., 2015).

Miklos Hajdu provided a critique of the current relationships that have been serving the professionals for several years (Hajdu, 2018). He also noted that not much effort had been invested in developing new types of relationships despite the limitations of the Precedence Diagramming Method (PDM) which is well-known to most professionals in the field. Hajdu also highlights the potential use of Point to Point (PtP) relationships in overcoming the downfalls of traditional techniques. This method was developed during the 1990s by creating logical connections between different points of the activities. An example of that is demonstrated in Fig. 2 below in the connections between activities **A** and **B;** for instance, activity **B** can start after finishing 50m of activity **A** which is expressed as (A—(50m, 0 m, 0 day)) in terms of quantity of work needs to be done before B commence. Another way of expressing the relationship can be in terms of time units in the case that can start after 2 days are concluded as (A—(2 days, 0 days, 0 days). The activities also need to be connected using defined points and the minimal durations between them. For instance, in Fig. 1 there are four points defined between A and B.

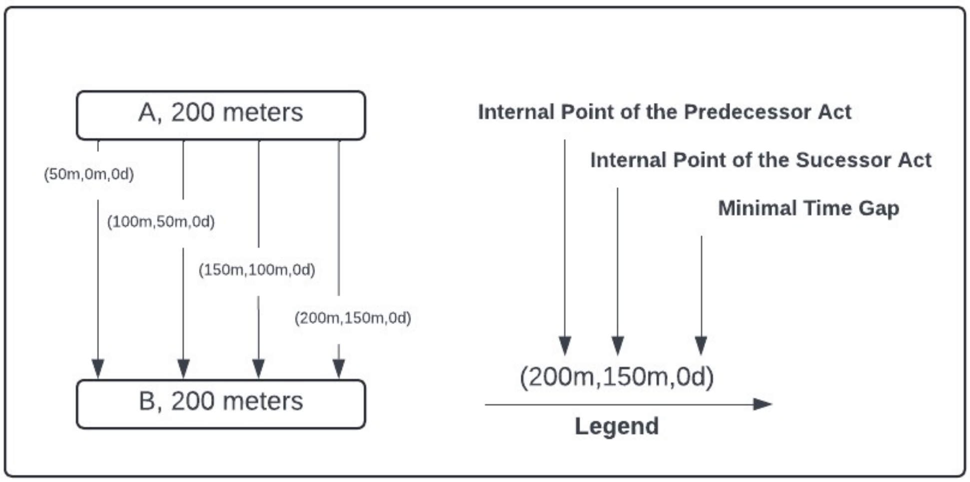


Figure 1 PtP relationship example work unit between A &B (Hajdu, 2018)

The author argues that this method has the same downfall as the traditional relationships; however, this can be overcome by increasing the number of points of connection between two activities, as this is defined by the user, to infinite and decreasing the length of segments to zero which gave rise to the “continuous relationships.” This is a type of relationship where all points of each activity are connected by either work or time status as seen in Fig. 2.

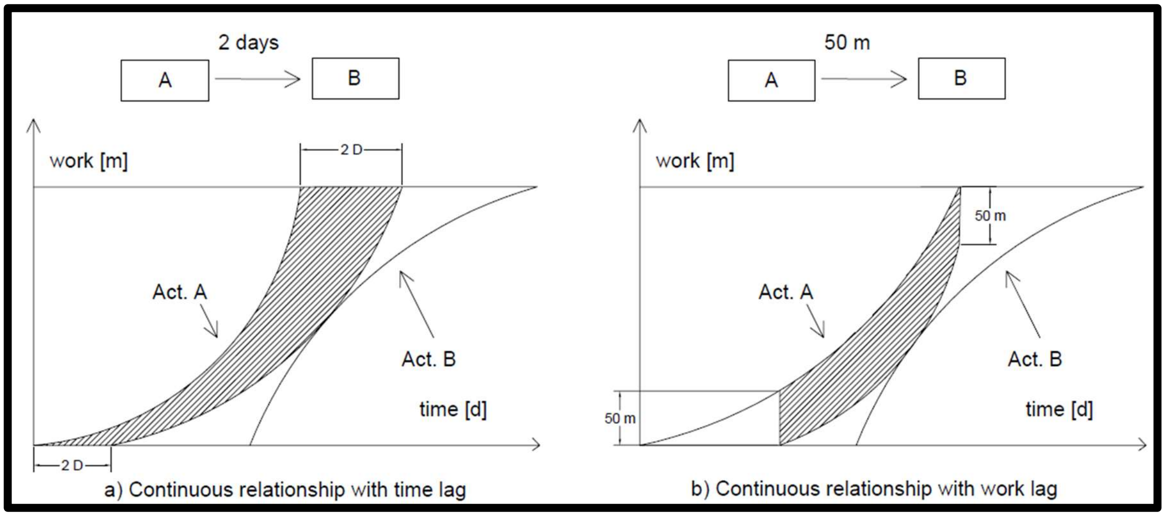


Figure 2 Continuous relationship between Activity A & B (Hajdu, 2018)

In theory, this can be a viable alternative to model overlapping activities as it gives better control over every point of the overlapping activities. Continuous relationships also can be used to define nonlinear activities. In this paper, the author also discussed an improved version of this technique by using logical switches, bidirectional relationships, and continuous precedence relationships for more control over dynamic activity changes.

2.2 Simulation and Optimization based scheduling

Simulation modeling is one of the techniques used in the literature for project planning to capture the stochastic nature of AEC industry activities and resource utilization. For example, The traditional CPM fails to properly capture the effect of material availability on the construction sequence and overall time schedule without changing the activity relationships. On the other hand, simulation modeling can provide a solution to the aforementioned limitation by only changing the number of sets of specific resources. Figure 3(a) illustrates two approaches to simulate activity scheduling in Critical Path Method (CPM) using different quantities of form sets. Figure 3(b) presents a similar model using simulation, which does not depend on the quantity of resources available (Chehayeb & AbouRizk, 1998). In the simulation model, the erection of formwork for a subsequent activity depends on the availability of formwork, rather than the dismantling of formwork from a prior task (Chehayeb & AbouRizk, 1998). This is represented in the simulation by requesting a form set to start an activity and then making it available again after use.

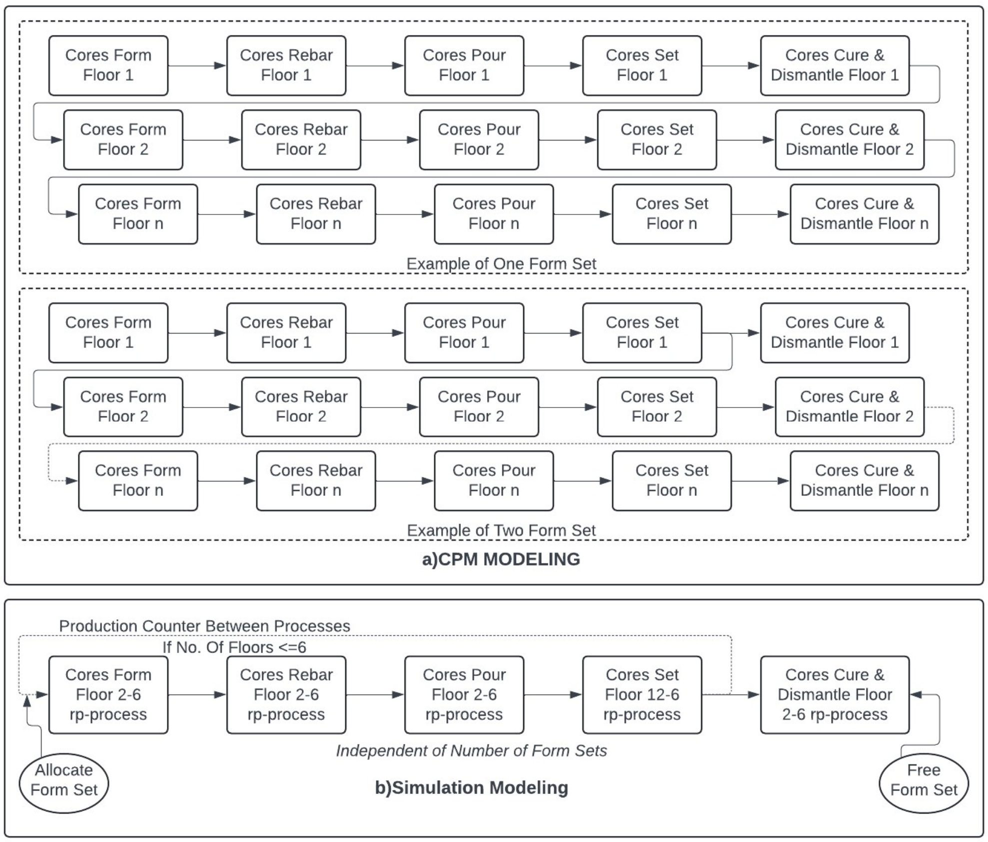


Figure 3 Comparison between Simulation and CPM Modeling (Chehayeb & AbouRizk, 1998)

Priority rules are also used to allow the user to allocate resources properly, and their importance is significant in simulating real-life processes in construction. For instance, the rule enables the allocation of a dismantled crew for formwork and another for erection when both activities can start simultaneously. In case only one crew is available, this rule enables the system to decide whether to dismantle the formwork in the current element or erect it for the next element, provided that there is available formwork and not only one is utilized for the two activities. The above shows advantages for simulation modeling, but there are some limitations. The first is the time consumed in preparation of the schedule as a detailed record of all activities, relationships, and resource utilization is needed for better control over activities. The second is that simulating cyclic processes could be hideous as relationships are not simple like CPM (Chehayeb & AbouRizk, 1998).

Isidore et al., (2002) introduce the Multiple Simulation Analysis Technique (MSAT) to integrate probabilistic scheduling aiming to better manage project risks by quantifying the relationship between project cost estimates and schedules. Initially, least-squares linear regression was considered for relating the data from these techniques. However, due to its limitations, it was found not to be the best method for connecting the results of range estimating and probabilistic scheduling. Instead, a new approach called the multiple simulation analysis technique was developed. The paper defined the term "range estimating and probabilistic scheduling" as techniques used to produce cost estimates and project schedules as probability distributions rather than fixed amounts. They involve defining activity costs and durations as probability distributions instead of static values. Once these distributions are established, a Monte Carlo simulation, which uses random number generation, is employed to sample from these distributions. For each simulation run, random values for costs and durations are generated and summed to calculate the overall project cost and schedule. Repeating this process numerous times produces probability distributions for both the total project cost and schedule. This method makes it easier to choose a cost estimate and schedule with a low chance of being exceeded. However, the output of the model is only a total project duration and a corresponding cost with no analysis to individual scenarios. (Isidore et al., 2002).

Hegazy & Kamarah (2008) also proposed a High-Rise Scheduling Model (HRSM) scheduling tool specifically developed for scheduling and cost optimization in high-rise construction it employs a genetic algorithm-based cost optimization to determine the best combination of construction methods, crew numbers, and work interruptions to meet three main scheduling constraints. Firstly, logical relationships within each floor (horizontal constraints. Secondly, logical relationships among floors (vertical constraints). Finally, work continuity constraints that ensure a smooth continuous workflow for the project. To ensure practical application, the model uniquely represents the activities forming the building's structural core, which must be managed carefully to avoid scheduling errors. Additionally, the model allows decision-makers to add extra work constraints related to resource and workflow requirements. A prototype program has been developed using the VBA language in Microsoft Project to automate HRSM functions. While this model considers time, cost, and crews in planning the project, it only provides the decision makers with overall it does not consider the stochastic nature of the project (Hegazy & Kamarah, 2008).

Zhou, J, et al. (2013) examined the current practices to address the construction schedule optimization (CSO) problem. The paper describes the problem as trying to optimally sequence activities and allocate resources in order to have the least duration, least cost, and highest profit. The methods of optimization are divided into three categories mathematical, heuristic, and metaheuristic. An example of mathematical methods are CPM, Integer Programming (IP), and Linear Programming (LP). The paper argued that the downfall of mathematical methods is formulating objective functions and constraints as this is a time-consuming and challenging task. Few construction professionals possess the necessary mathematical training to effectively perform such formulations. Consequently, the application of these techniques to construction and engineering project scheduling has been limited. The first step is to formulate the problem at hand by formulating the objective function and constraints. Also, some techniques like hill-climbing algorithms and CPM are single objective that might not capture the true nature of schedules. On the other hand, IP and LP have been applied to solve the time cost trade-off problem and tackle the limitations of the CPM and can be utilized for multi objective problems (Liu et al, 1995).

2.3 Dynamic programming

Another mathematical method is Dynamic Programming (DP) applicable to solving complex problems that can be broken down into some sub-problems (Zhou, J, et al., 2013). Moselhi and El-Rayes (1993) developed a DP model that introduces a cost variable into the optimization problem along with minimizing. which is divided into two steps. The first is the forward problem which tackles the time cost trade-off problem. The second is the backward process which includes scanning the processes and making sure that the minimal state is realized (Moselhi & El-Rayes, 1993). Also, DP has been used in the literature to develop a long-range facility management strategy by overcoming by incorporating uncertainties that might affect the decisions and constraints (Botros et al., 2010). Previously this technique has been used in long-range pipeline design as an optimization technique to determine either the least cost path for pipes or the most probable path i.e., the path with the least constraints. K.K. Botros discusses in his paper titled " Long-Range Facility Planning Based on Dynamic Programming for Optimum Combined Cost and Probability Paths" two new variations for the dynamic programming to constitute a multi-objective optimization problem where the goal is to find the most probable path for the pipes as well as reduce the cost of the project (Botros et al., 2010). This can be utilized in the project management domain as a network of activity for a project can be mimicked by a network of pipelines; hence, this method has the potential to address the limitation of the inclusion of probability in project planning. Robinson (1975) developed a DP model to tackle the time cost trade off problem and determine the allocation which minimizes the duration of the project (Robinson, 1975).

2.4 Fuzzy Linear and Repetitive Scheduling

Katsuragawa et al., (2021). presents previous efforts in the literature to capture the uncertainty element in the planning for construction activities. The paper titled "Fuzzy linear and repetitive scheduling for construction projects" introduces the Monte Carlo simulation as one method that generates multiple runs to quantify the behavior of individual activities. By randomly selecting activity duration according to the defined probability distribution, this method is able to overcome the limitation of the PERT method. On the other hand, the paper highlights the limitation of this method as it is only able to simulate discrete events and discrete links, which is not the reality of construction activity sequences.

The paper also discusses the potential usage of fuzzy logic in simulating projects. This is achieved by defining each activity duration using a membership function and fuzzy addition along the network paths to mirror the technique of CPM by using centroids in fuzzy scheduling and defining fuzzy criticality and fuzzy float analogous to the traditional CPM. This is achieved by modeling the duration for optimistic, realistic, and pessimistic durations of activities by a fuzzy cone. Also, the study considered the spatial and safety constraints between activities by representing continuous buffers. Finally, the paper aims to identify fuzzy critical segments or ones with fuzzy floats that might increase without increasing the overall likelihood of affecting project duration (Katsuragawa et al., 2021).

2.5 Activity Visualization:

Another aspect of this research is the visualization and graphical visualization of construction activities and sequences. There have been several mentions in the literature of the drawbacks of the current network representations for construction activities and attempts to enhance existing methods. GANT charts, Line of Balance (LOB), and CPM have been widely used as visual illustration methods. The LOB is used specifically for repetitive construction projects.

The CPM is more widely used for standard projects as it illustrates the activity predecessor, successors, the type of relationship between activities, and the critical activities that affect the overall project duration. However, these methods have several limitations. Firstly, the traditional relationships show only time-related constraints and not other factors such as resource availability and workspace. The second issue is the activity cluttering that occurs when the network size increases, so it becomes challenging to visually determine the relationships between activities. Thirdly the currently available tool does not provide ways of comparing different schedules to each other in an effective manner. While software like Primavera P6 can display different bars for different schedules, for instance, as-built vs as-planned, it provides minimal support for comparing many alternative schedules. Finally, the current scheduling methods fail to incorporate uncertainty in the logic of the network itself. For example, This situation occurs in cases where site conditions beneath the soil are not clear, the works need to commence, and appropriate action needs to be determined afterward during the execution. In this scenario, the project team constructs a schedule with a certain scenario, and if conditions are to be different from those that the team has planned, then the team has to construct a different schedule to accommodate the different scenarios. (Naticchia et al., 2019).

The field of operational research has explored the adaptation of different network representation techniques, such as Activity on Arrow (AOA) and Activity on Node (AON), well known activity representation techniques. Dawson & Dawson (1995) constructed the framework for the Generalized Activity on Node (GAN) network that allowed for stochastic network and probabilistic activity branching. According to the authors, there has not been much effort in the literature to represent stochastic networks using AON. Instead, most of the efforts were directed toward AOA. The authors argued that the AON is superior to AOA because of its compatibility with existing software tools and the ability to apply the PDM constraints. In their work, the authors introduced the rules of constructing a GAN with stochastic relationships such as deterministic nodes, stochastic independent OR Exclusive OR, Dependent OR, and cost/time dependencies. The author of this research adopted this framework in developing the visualization technique for this research.

2.6 Scenario Analysis for Uncertainty in GAN

The literature presents a few works that tackle the analysis of GANs, which are found in the area of operational research. In the early days of introducing the Generalized activity network, Elmaghraby provided a framework to represent the generalized activity network using AOA and developed the required algebra to perform the network analysis and conclude the estimated project time (1964). Pollack-Johnson & Liberatore (2005) provided a simplified approach for the analysis of uncertainty in probabilistic scenarios . The author's approach included the identification of each of the probable scenarios during the execution and the identification of the corresponding probability of occurrence. Afterward, CPM calculations are performed for each scenario separately and aggregated values for the duration, and project completion time are calculated.

2.7 Agent Based Modeling and Discrete Evenet Simulation

In the literature, exists efforts where Discrete Event Modelling (DES) has been utilized to simulate complex operations in different industries. It has been used to simulate complex and decentralized system environments in social and environmental sciences and economics. DES has several applications within the realm of the construction industry as it was used to mimic construction management procedures at the micro level and how incentives encourage safe behaviors on construction sites. Sadeghi et al (2016) highlight the shortcomings of other scheduling methods and proposed an enhanced approach to tackle this problem by integrating fuzzy logic into DES to enhance the accuracy of the activity durations. Sadeghi argued that DES simulation uses probability distributions that are obtained through human input by expressing linguistics of "high" or "low" probability instead of having precise probabilities. The use of fuzzy logic enables a better representation of these linguistics. DES has also been used to include resource constraints in CPM scheduling (Lu, Ming, et al, 2008). Lu Ming et al (2008) argued that the calculation of float in the critical path when resource constraints are applied is not guaranteed. Therefore, a combination of DES and particle swarm optimization is used to formulate a resource constained schedule with the shortest project duration. Lee, Dong-Eun (2010) introduced a simulation technique named Construction Operation and Project Scheduling (COPS) that is based primarily on DES. DES proved a useful tool to analyze productivity at the activity level and also can be utilized to model the stochastic duration of activity on project level. Lee provided a framework to integrate these two uses of DES. The model estimates the best probability distribution for each activity's duration, then the model provides the best resource allocation that meets the project constraints in terms of time and cost. Finally, the model calculates the project duration with a corresponding probability distribution. Agent-based modeling is used to capture the dynamic and uncertain nature of construction environments.ABM handles complex interactions and emerging behaviors by modeling the system as a collection of interacting agents, representing the construction environment. To further reduce computational complexity, graph embedding networks are utilized. These networks embed the graph structure of construction activities into the RL process, streamlining the computational effort required to establish activity sequences and work breakdown (Kedir et al, 2022). Cao et al (2015) utilized agent based modeling in facility management applications. The proposed framework uses agent-based modeling (ABM) to prioritize maintenance tasks by considering energy efficiency and occupant satisfaction alongside traditional factors like system failure and safety. Occupant satisfaction is quantified using data from current building maintenance work and is grounded in classical disconfirmation theory and post-occupancy evaluation (POE) research. A survey is conducted to collect data on occupant satisfaction, which is then used to develop an ABM that prioritizes maintenance work for increased satisfaction (Cao et al, 2015). Jabri and Zayed (2017) tackled the issue of planning the earthwork in their research that adopts agent-based modeling. They argued that ABM is a better alternative to Discrete Event Simulation (DES) as it can take into account the specification of different earthwork equipment to better simulate varying equipment and realistic earthwork operations (Jabri & Zayed, 2017).

2.8 Monte Carlo Simulation with GERT Network

The research also reviewed the current research in the area of operational research to adapt some of the ongoing work and implement it in construction industry applications. Kurihara et al., (2002) provided a framework to utilize Monte Carlo Simulation with Graphical Evaluation and Review Technique (GERT) for analysis of stochastic schedule with probabilistic branching in the paper titled "Efficient Monte Carlo GERT Type Network". The authors highlighted the ability of the GERT network model to handle probabilistic treatment of network activities and paths. According to the paper, they provided a frame to enhance the efficiency of Monte Carlo simulation methods when applying to GERT networks, while reducing the computational effort This method involves generating random samples to estimate the statistical properties of the network's performance. To improve the efficiency of Monte Carlo methods, the study proposes various techniques such as variance reduction methods and improved sampling techniques. These enhancements are designed to reduce computational time and resources while ensuring the accuracy of the simulations. The results of the study indicate significant improvements in the performance of the proposed methods.

Various GERT network models were tested, and the findings showed substantial gains in computational efficiency and accuracy. However, this approach does not address the potential means to analyze individual scenarios during execution and uses the AOA method to construct the networks to be analyzed which is not familiar to professionals in the domain of the construction industry.

2.9 Summary of the Literature Review

The above review of the literature demonstrated the current attempts towards developing project planning based on several aspects depending on the nature of the project. Some attempts were scheduled based on resources and material availability. Other attempts used principles of dynamic programming and modeling the activity network behavior similar to that of a pipeline that follows the path of least constraint and lease cost; this approach only considers the project's financial aspects. The literature also presents attempts to illustrate the activities and relationships based on new relationships that are dependent on the amount of progress of the predecessor.

The literature also explores the current network representation techniques, such as AOA and AON, which are used to demonstrate the project relationships and calculations for the critical path. However, no major research has explored the use of AON to represent construction activity with probabilistic branching that demonstrates different execution scenarios. Moreover, the attempt to simulate a probabilistic activity network has been conducted using only Fuzzy Logic simulation , which only demonstrates the behavior of the network overall (e.g., overall project duration, overall project cost). The use of discrete event simulation will provide the means to examine the whole network on the macro level and the parameters of each activity on the micro level. Table 1 summarizes the current attempts to consider different parameters when planning projects.

Table 1 Summary of Current Project Scheduling Techniques

Modeling technique	Considerations	Limitations
PERT	Random Activity duration with different statistical distributions	Does not consider logical relationships
CPM	Activity logical relationships such as FS, FF, SS	Does not consider multiple scenarios
Simulation Based Modeling with Continuous Relationships	Priority rules that allocate resources to critical activities	Does not consider stochastic durations or different scenarios
Monte Carlo Simulation with GERT	Variable project duration Visualization with GERT	No analysis in Individual Scenarios Does not include cost aspect
Dynamic Programming	The cost aspect of the project and finding the path with the least cost to be executed	Does not consider stochastic durations or different scenarios
Fuzzy Linear and Repetitive Scheduling	stochastic duration of activities through fuzzy logic and space constraints within the site to ensure safety	Does not consider different scenarios
Point to Point Relationships	Distance constraint between each activity prior to execution of the successor	Does not consider stochastic durations or different scenarios
Generalized Activity Network	Multiple scenarios during execution	Professionals are not familiar with AOA diagrams

Table 2 Review of Scheduling Techniques

Reference	Model Type	Deterministic Relationship	Stochastic Relationship	Probabilistic Cost	Probabilistic Duration	Network Visualization	Individual Scenario Analysis	Resource availability	Type of Construction
Tomczak et al. (2024)	Particle Swarm Optimization (PSO)	Y						Y	Repetitive Projects
Pollack & Liberatore (2005)	CPM		Y			Y	Y		General
Botros et al. (2010_	Dynamic Programming	Y		Y			Y	Y	Linear Projects
Katsuragawa et al. (2021)	Fuzzy Logic								Repetitive Projects
Kurihara, & Nobuyuki (2002)	Monte Carlo		Y			Y			General
Lu, Ming, et al (2008)	DES with Particle Swarm	Y			Y			Y	General
Lee, Dong-Eun et al. (2010)	Construction Operation and Project Scheduling with DES	Y			Y		Y	Y	General
This Research	Discrete Event Simulation with GERT	Y	Y		Y	Y	Y		General

Chapter 3 – Model Development and Implementation

This chapter of the research illustrates the details of the development of the model and the algorithms used to model stochastic network behavior with its agents. The construction of the model was conducted using AnyLogic software, as it has proved to be one of the most capable software for discrete event simulation because of its predefined blocks that enable accurate behavioral modeling along with the ability to add Java command lines within the model to constraint the behavior of the agents. The model views the activities within the project as agents, and each project run is an execution attempt for the whole project. This construct allows the model to consider the stochastic scenarios that occur whenever the project has unforeseen conditions that give rise to different probable scenarios during the execution. This is captured by introducing stochastic nodes where in each run different scenario is realized at each node. The duration of the activity of the activity is deterministic in order to decrease the analysis time needed for the study of the outcome of the model. This study also provides the framework through which the user can construct a new network that is a combination of both the Precedence Diagram Method (PDM) and the Generalized Activity Network (GAN) which provides a more user friendly to professionals in the field as it is closer to PDM. The aforementioned model is divided into 1) an input module for the user to insert the needed parameters, and the rest of the needed parameters are computed using Excel functions 2) Simulation execution in AnyLogic to calculate the parameters of each agent 3) Output of several project runs would be analyzed to get the different paths outcome.

3.1 Agent Parameter Definition

The first aspect of the model is choosing the parameters the user is required to input, such as:

- Activity
- Activity Description
- Predecessors
- Activity Receiver
- Activity Emitter
- Activity Duration
- Probability of Execution (in case of multiple scenarios)
- Activity code

All the above items are the required input from the user in the Excel file that later uses this input to calculate the other parameters that are needed for the initialization of the simulation using Excel functions such as:

- Number of predecessors
- Predecessors Codes
- Number of Successors
- Successor Code

All the above items are used as the parameters for each agent (activity) within the simulations as Fig. 4 illustrates.

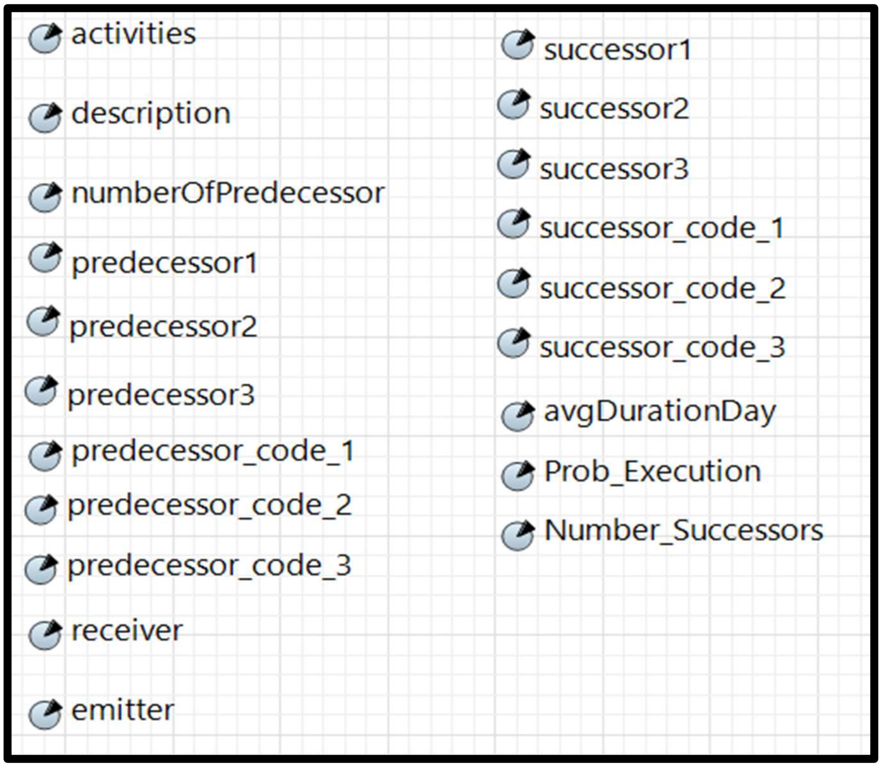


Figure 1 Initialization parameter for each agent at the beginning of the simulation

It is worth noting that the input for this model is only allowed for 3 successors and 3 predecessor activities as this model is a proof of concept. However, with minor adjustments to the code line, this can be easily increased

The values for the parameters above are introduced from the aforementioned input Excel sheet at the beginning of the simulation. The type of each parameter is demonstrated in Table 2.

Table 3 Input parameters for the model initialization data types

Parameter	**Date Type**
Activity	String
Description	String
Number of Predecessors	Integer
Predecessors	String
Predecessor Code	Integer
Receiver	String
Emitter	String
Successor	String

Successor Code	Integer
Number of Successors	Integer
Average Duration (days)	Double
Probability of Execution	Double

The activity parameter represents the short version of the activity that will later be used in the visualization of the network, while the description of the activity provides details of what actions would be performed during the duration of the activity, and both are defined with a string data type. On the other hand, the predecessors of each activity are input by the user in a string format in the database, and the Excel file automatically calculates the corresponding number of predecessors and predecessor code for each. The same principle applies to the successors and successor codes.

Fig.5 demonstrates a typical activity node within the project, where each has a receiver, which governs the relationship between this activity and its predecessor, and an emitter, which governs the relationship between the activity and its successor.

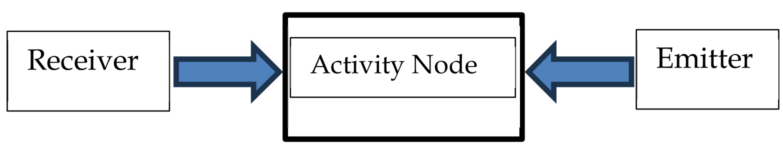
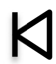
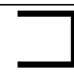


Figure 5 Typical Activity Node

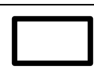
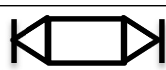
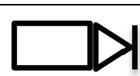
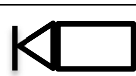
As discussed, one of the aims is to capture the stochastic relationship between activities, hence the need for stochastic receivers and emitters, as shown in Table 3. The Stochastic receiver/emitter represents the Exclusive OR relationship "XOR," which in the case of the receiver implies that only one of the predecessor activities will be completed while the other two remain idle, while in the case of the emitter implies that only one of the successors will be conducted. Based on this, it is concluded that these stochastic nodes would only be used in the project at the nodes with probabilistic branching for different scenarios. On the other hand, the deterministic receivers mandate that all the predecessors of the activity are to be completed before this activity can commence. Also, the deterministic "AND" emitter implies that all successors of the activity are to be undertaken.

Table 4 Different types of Activity Receiver and Emitters

Node	Stochastic	Deterministic
Receiver		
Emitter		

Different activity nodes might have different combinations of receivers and emitters, as Table 5 demonstrates, where the deterministic receiver and emitter constitute a deterministic node, and the stochastic receiver and emitter make the stochastic node. Also, a combination of both can also be found in projects where the probability branching is in the predecessor or the successor activities.

Table 5 Different types of activity Nodes

Node	Node	Type
1		Deterministic node with both the receiver and the emitter as "AND"
2		Stochastic node with both receiver and emitter as XOR
3		Combination of a deterministic receiver "AND" and stochastic emitter "XOR"
4		Combination of a stochastic receiver "XOR" and deterministic emitter "AND"

In order to simulate the interaction between the activities, additional parameters needed to be defined within the population of agents that are calculated and updated during the runs, as Fig. 6 demonstrates.

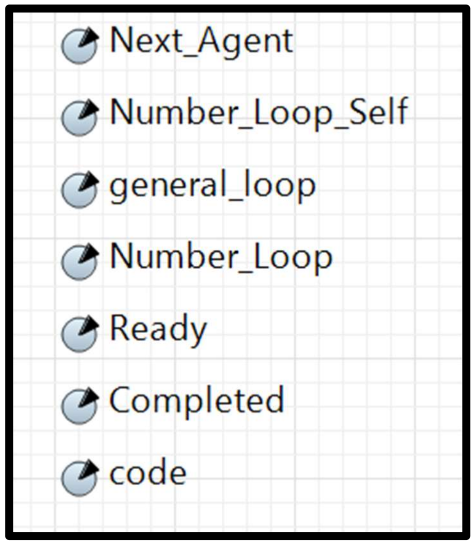


Figure 3 Activities parameters defined within the model for simulation of behavior

The type of each parameter is defined in Table 6.

Table 6 Data Types for Activity Parameters defined in the model.

Parameter	**Date Type**
Next Agent	Integer
Number of Self-Loop	Integer
General Loop	Boolean
Number loop	Integer
Ready	Double
Completed	Double
Code	Integer

The above parameters are used to determine the behavior of the networks in terms of activity precedence and succession. The "Next Agent" parameter is used as a means of communication between each agent and the successor agent. This is done by assigning the “activity code “of the successor as the “Next Agent” based on the probability of execution in nodes with stochastic emitter. The introduction of stochastic branching also allowed for the introduction of loops within the network. There are two types of loops, as Fig 7 demonstrates.

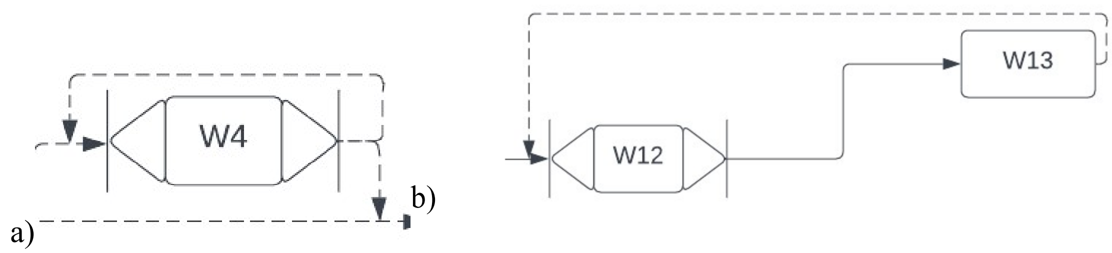


Figure 4 Example of Self loop a) and General Loop b)

The self-loop is self-explanatory, where the activity is conducted more than once, while the general loop is defined by the occurrence of a group of activities several times. The “General Loop” parameter is used to identify the activities that are part of the general loop by using Java command, which will be examined in later sections. The “Number Loop” and “Number of Self Loop” are counters within the simulation to record the iteration of each loop that will aid in the display of the output parameters. An example of the general loop would be the iteration of testing or inspection on site. There are several occasions where this loop is manifested. For example, water proofing of structures often undergoes testing to ensure that it is adequately applied. If not appropriately applied, then it is reapplied until its effectiveness is ensured. For the self looping activity, an example of that would be doing archaeological research for artifacts found during the excavation work. Multiple rounds of research might be needed depending on the significance of the findings. The “Ready” parameter is used to determine whether this activity is ready to be executed (i.e., all the predecessors are completed), with a value of either 0 for “not ready” or 1 for " ready”. With the same principle, the parameter “Completed” was defined to identify the activities that have been conducted and assign the value of 1 for "completed" and 0 for "not completed.".

In order to record the output of the model, the parameters below are defined to record the output values for each agent during each run, as Fig.8 demonstrates.

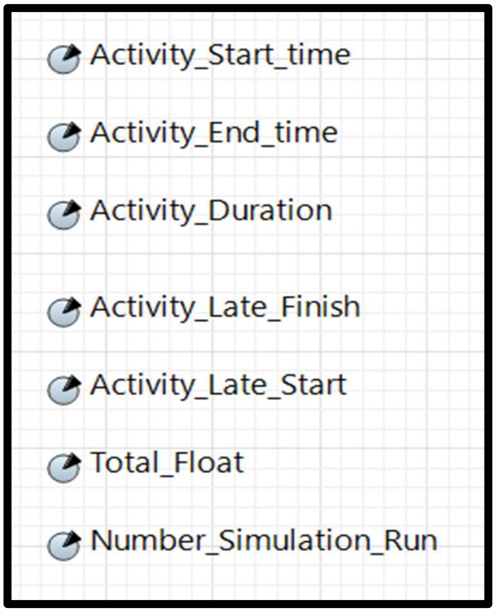


Figure 5 Output parameter setup

Each activity's start time is captured to calculate the ES and EF by following the equation.

$$EF = ES + Duration$$

The LS and LF are calculated by means of a backward pass using the following equation.

$$LS = \ LF - Duration$$

The total float is calculated using the following equation.

3.1 Simulation Module

3.1.1 Definition of Input

After the definition of the previously mentioned parameters that are either input by the user or calculated in the Excel spreadsheet, the Excel sheet is embedded within the software as a database reference for the initialization of the parameter values at the beginning of the simulation, which is done by adding the Excel file named “input_output” which contains the input parameter in a separate tab along with the output of the tab in another. The figure below illustrates the whole construct of blocks that constitute the simulation for the project run, which contains a logical loop and actions at each block that will. The flowchart in Fig.9 illustrates the general framework that is later translated into AnyLogic software. The first step is that the activity is generated and goes through several sequential steps. The first step is evaluating whether all the predecessors of this activity are completed or not, in the case the agent goes into the revaluation process. Once all predecessors are completed. The agent goes into the execution process where it is assigned a certain duration. Afterwards, the activity emitter is evaluated to identify whether it is an XOR or not. In case the successor of the activity has the same activity code, that implies that this activity is part of a self loop structure which means that this activity can be repeated more than once. If not, then the successor of the successor, assigned the name Next Agent, is also evaluated to determine whether the activity is part of the general loop or not. If the activity emitter is not OR then the activity goes directly to sink after being executed.

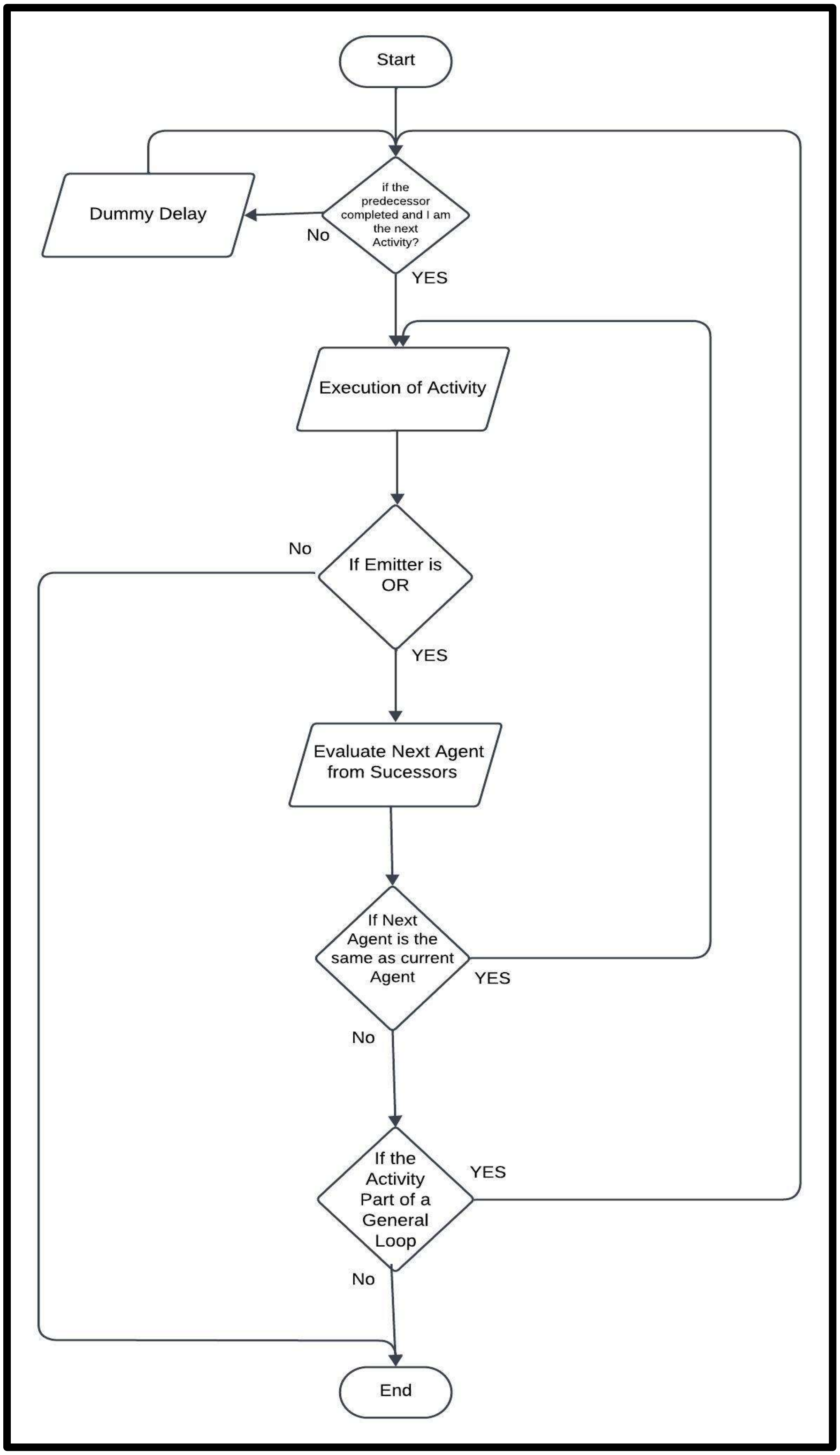


Figure 6 Model Logic Flow Chart

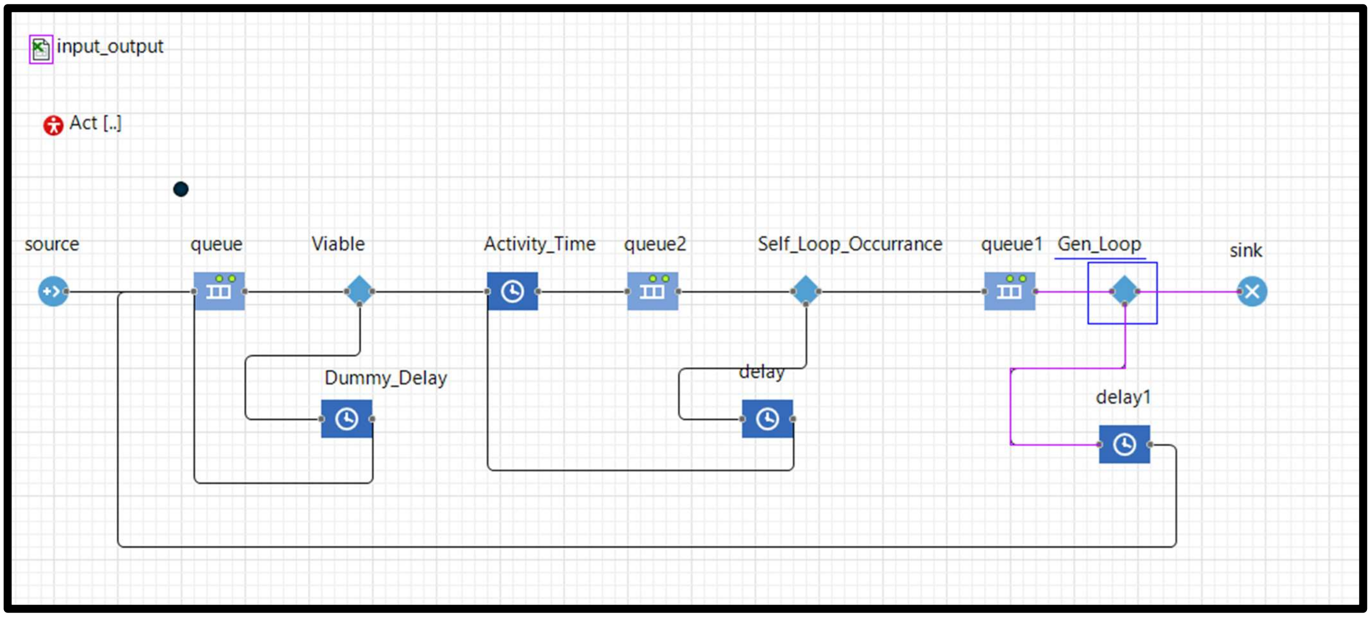


Figure 7 AnyLogic Model construct.

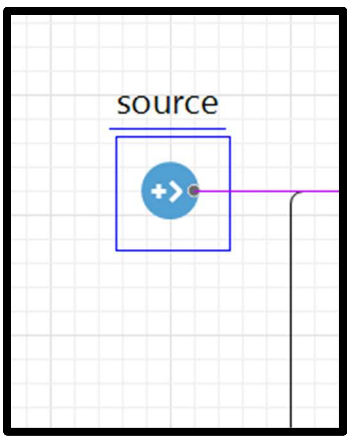


Figure 8 Source Block

3.1.2 Activity Generation

The first section of the model is the source block illustrated in Fig.11 which initializes the number of agents, which in return corresponds to the number of activities that are intended to be executed in the project's duration. The only action at this block is at the “On Exit” action using the function Excelfile.setCellValue(value, sheet index, row index, column index). This first check of the model is shown in Fig.12, which illustrates the code used to display the parameter value at the exit of the source to verify that the input parameters inserted in the Excel sheet are of the same value at the exit of the source.

On exit:

```
input_output.setCellValue(agent.number, 2, agent.number+1, 1);//
input_output.setCellValue(agent.activities, 2, agent.number+1, 2);//string
input_output.setCellValue(agent.description, 2, agent.number+1, 3);//string
input_output.setCellValue(agent.numberOfPredecessor, 2, agent.number+1, 4);//int
input_output.setCellValue(agent.predecessor1, 2, agent.number+1, 5); //string
input_output.setCellValue(agent.predecessor2, 2, agent.number+1, 6);//string
input_output.setCellValue(agent.predecessor3, 2, agent.number+1, 7);//string
input_output.setCellValue(agent.predecessor_code_1, 2, agent.number+1, 8);//int
input_output.setCellValue(agent.predecessor_code_2, 2, agent.number+1, 9);//int
input_output.setCellValue(agent.predecessor_code_3, 2, agent.number+1, 10);//int
input_output.setCellValue(agent.receiver, 2, agent.number+1, 11);//string
input_output.setCellValue(agent.emitter, 2, agent.number+1, 12);//string
input_output.setCellValue(agent.successor1, 2, agent.number+1, 13);//string
input_output.setCellValue(agent.successor2, 2, agent.number+1, 14);//string
input_output.setCellValue(agent.successor3, 2, agent.number+1, 15);//string
input_output.setCellValue(agent.successor_code_1, 2, agent.number+1, 16);//int
input_output.setCellValue(agent.successor_code_2, 2, agent.number+1, 17);//int
input_output.setCellValue(agent.successor_code_3, 2, agent.number+1, 18);//int
input_output.setCellValue(agent.avgDurationDay, 2, agent.number+1, 19);//double
input_output.setCellValue(agent.modeDay, 2, agent.number+1, 22);//double
input_output.setCellValue(agent.Loop_0_1, 2, agent.number+1, 23);//int
input_output.setCellValue(agent.Completed, 2, agent.number+1, 30);//int
input_output.setCellValue(agent.code, 2, agent.number+1, 31);//int
```

Figure 9 Java code line to print initial model parameters for verification.

3.1.3 Activity Viability Loop

Upon exiting the source block, the agents enter the first loop shown in Fig.13 which is the viable loop. This loop is set up in a way that allows only the activities that are ready for execution to proceed forward for the next process loop. If not, the agent keeps being reevaluated in the loop until it is ready to be executed and moves to the next evaluation step. The queue block is added at the beginning of the construct to arrange the entry of agents into the viability loop as the agents will keep navigating the loop until all the predecessors are executed and the agent is ready to move to the next loop of evaluation. The dummy delay is added in this loop to allow for the continuous navigation of the agents in this loop until the aforementioned condition is realized. The need for this block was realized during the verification of the model where the agents were not navigating the loop but rather idle. Finally, the viability block evaluates whether this activity is ready for execution or not.

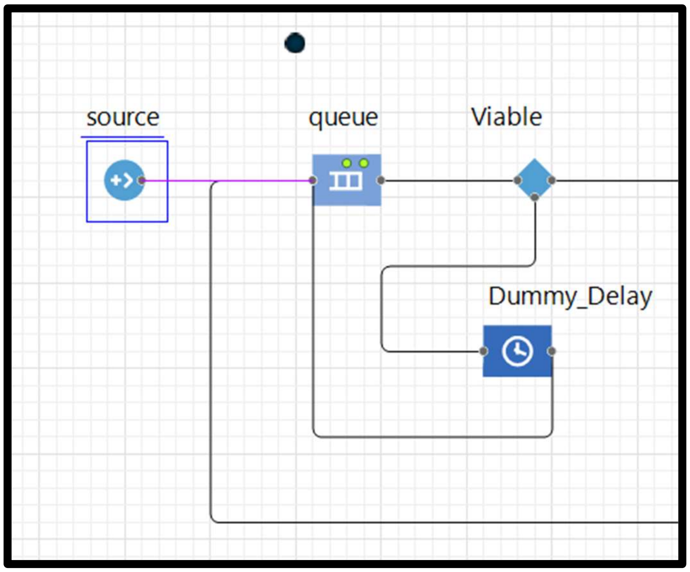


Figure 10 Activity Readiness Viability Loop

This model is constructed in a manner where all the evaluation of the relationships is conducted inside the queue with a series of instructions, as per the code in Fig. 14

```
int flag = 1;//inital condition of the flag variable

if(numberOfPredecessor==1)
{
if(Act(the first predecessor emitter is OR ))

if ( the Next Agent code of the first predecesosor != code of the agent)
{
    flag = 0;
}
}//Checks for the case of 1 predecessor

if(numberOfPredecessor==2)
{
if(Emiiter of one of predecessor is OR)
if (code of the Next Agent of either predecssors is not equal to the code of the agent)
{
    flag = 0;
}
}//Checks for the case of 2 predecessor

if(number of predecssors is 3 ){
if(the emitter of one of predecssors is OR )

if (code of the Next Agent of any predecssors is not equal to the code of the agent)
{
    flag = 0;
}
}

if(number of predecssors ==0){if(flag==1) ste Ready=1;}// to set the first activity ready by default
if((number of predecssors==1)&& predecessor 1 Completed==1){if(flag==1) set agent Ready=1;}// set ready if 1 predecessor

// If statment for AND Relationship
if(receiever of the activty node is AND)
{
if(predecessor 1 & predecessor 2 are completed {if(flag==1) set agent Ready=1;})// and relationship in case of 2 predecessors
if(predecessor 1 & predecessor 2 & predecessor 3 are completed ){if(flag==1)set agent Ready=1;}
}

// If statment for OR Relationship
if(receiever of the activty node is OR)
{
if(number of predecessors ==2) && (predecessor 1 or predecessor 2 are completed ){if(flag==1) set agent Ready=1;}// OR relationship in case of 2 predecessors
if((number of predecessors==3)&&(predecessor 1 or predecessor 2 or predecessorsare completed )){if(flag==1)agent.Ready=1;} //OR
}
```

Figure 11 Pseudo Code snippet for the queue block

The first section of the code is concerned with the stochastic branching nodes where there are several execution scenarios. The variable flag is used as the indication of whether this activity is to be executed in the case of the previous node being XOR. If the condition is met, the agent goes through the second evaluation process for each relationship type, as demonstrated in Fig. 14. Whereas whenever the receiver of the node is "AND" for the activity to be "Ready" for the execution, all the previous predecessors should have been “Completed.” On the contrary, if the receiver of the node is “OR” for the activity to be ready for execution, any of the predecessors should have been “Completed.”

After assigning a value to the parameter “Ready” with either 1 or 0, the viable block allows the activity to pass to the next block based on the assigned value for the “Ready” parameter as per Fig. 15

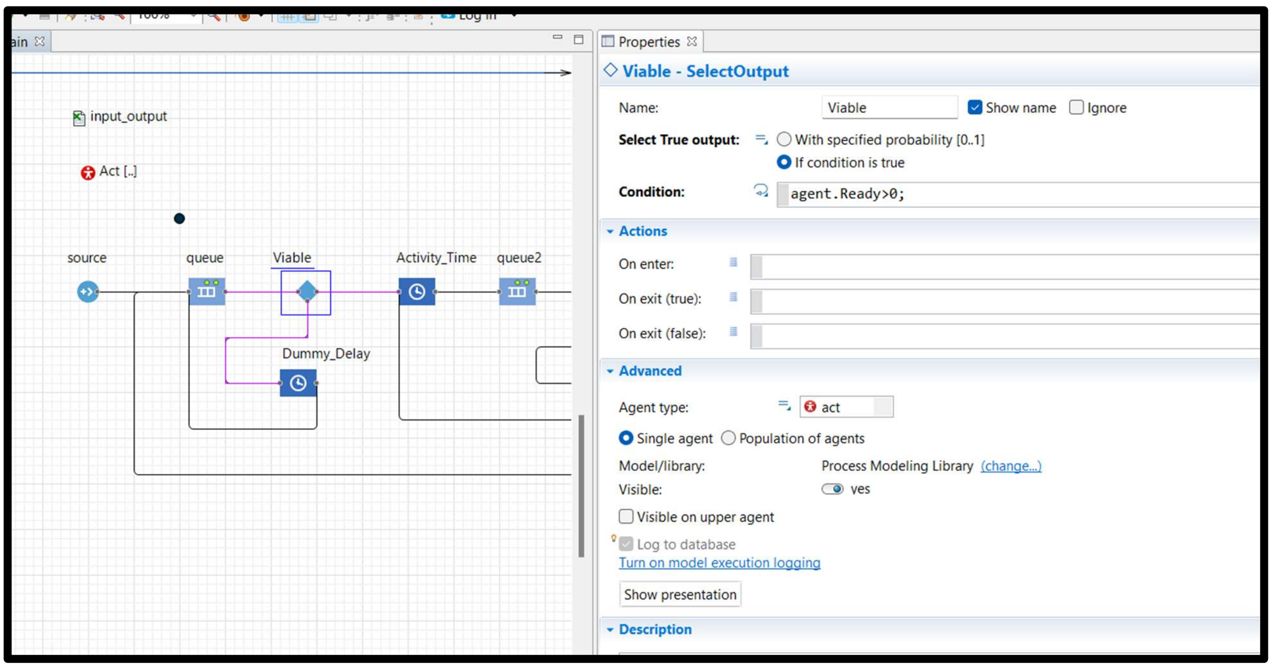


Figure 12 Condition for Activity viability loop.

3.1.4 Activity Execution Loop

If the activity is “Ready” for execution, then enter the execution loop demonstrated in Fig.16 in which the activity spends the duration assigned to it through the user in the input sheet.

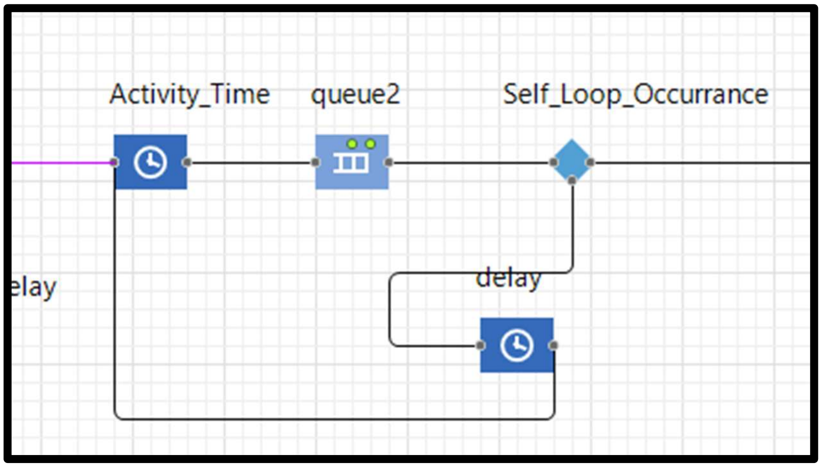


Figure 13 Activity Execution Loop

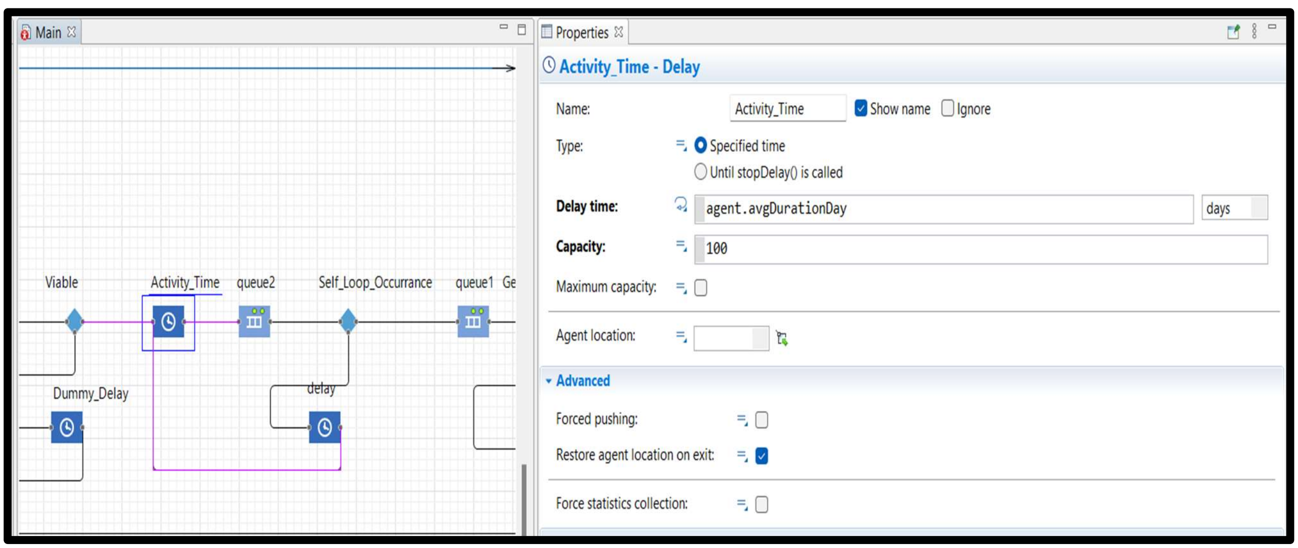


Figure 14 Activity Duration in the Execution Loop

The first of the needed output is captured upon entering this block in the "On enter action" by assigning the "Activity Start Time" parameter with the date() function that captures the time stamp of entry.

Four actions are executed after the agent spends the assigned time in the delay block that is obtained from input parameters as shown in Fig.16, and upon exit. The first is that the exit time is recorded by assigning the date() function to the “Activity End Time” parameter, which is the second output of the model. The second action is that the parameter “Complete” is set to “1” to indicate that his activity has been concluded to allow the execution of the successor activities. The third is the printing of the values of the activity End time, which corresponds to the EF, and the duration of the activity which is the difference between the Activity Start Time and Activity End Time using the differenceInCalendarUnits() function. The fourth action is the evaluation of which activity is to be executed according to the assigned probability of execution by evaluating the probability against a randomly generated number from 0-1 using the Math.random function. By recording the ES and EF values in this block, the forward pass calculations are concluded in this loop. The code in Fig.18 demonstrates the code used for the aforementioned functions in the Activity_Time block On Exit.

```
//sets activity complete parameter to 1
Set activity Complete = 1; // Sets the activity as completed when exit the Activity time delay block

Print the value of the complete parameter in the output sheet


//capture activity end time
Set Activity End Time parameter as the time of exit from the Activity time block;
Assign the Activity End Time parameter to the current Activity;
Set Activity duration as the difference between the Activity Start Time and Activity End time;
Set Activity duration parameter as the Activity Duration of current Activty;
print Activty End time in the Output Excel Sheet
Print Activity Duration in the Output Excel Sheet


//Assigns the sucessors of the current activity using Probability of Execution Variable

Assign the variable Random Number with a random number fromm 0-1;

if(Emitter of the Activity is OR)
{

if(Number of Sucessor is 2)
{
Assign the variable sum with the Probability of Execution value;
    if((randomNumber>= 0)&&(randomNumber<sum))
    {
        Assign Next Agent code as code of sucessor number 1;
    }
    else
    {
        Assign Next Agent code as code of sucessor number 2;
    }

}
else
if (Activity has 3 sucessors)
{
    Assign the variable sum with the Probability of Execution value of sucessor 1;
    Assign variable sum2 = Probability of Execution value of sucessor 2 +Probability of Execution value of sucessor 1;
    if((randomNumber>=0)&&(randomNumber<sum))
    {
        Assign code of Next Agent as code of sucessor 1;
    }
    else
    if((randomNumber>=sum)&&(randomNumber<sum2))
    {
        Assign code of Next Agent as code of sucessor 2;

    }
    else
    if(randomNumber>=sum2)
    {
        Assign code of Next Agent as code of sucessor 3;
    }
}
if(agent.Number_Successors == 2)
{
//checks whether we have general loop or not in case of 2 successors
if (activity code of the sucessor of the Next Agent is < code of the Next Agent for the two sucessors)
{
set the general loop variable of the activity and the Next Agent as false;
}
else
{
    set the general loop variable of the activity and the Next Agent as true;
}
else
//checks whether we have general loop or not in case of 3 successors
if(number of sucessors = 3)
{
if (activity code of the sucessor of the Next Agent is < code of the Next Agent for 3 sucessors){
set the general loop variable of the activity and the Next Agent as false;

}
else
{
    set the general loop variable of the activity and the Next Agent as true;
}
}
```

Figure 15 Code Snippet for Activity Duration Block

3.1.5 Activity Self Occurrence

Upon exiting the Activity_Time block, the select output block evaluates whether this activity will be executed several times, this is referred to as self-loop in Fig.19 by evaluating whether the Next Agent code to be executed is not equal to the current agent code being evaluated.

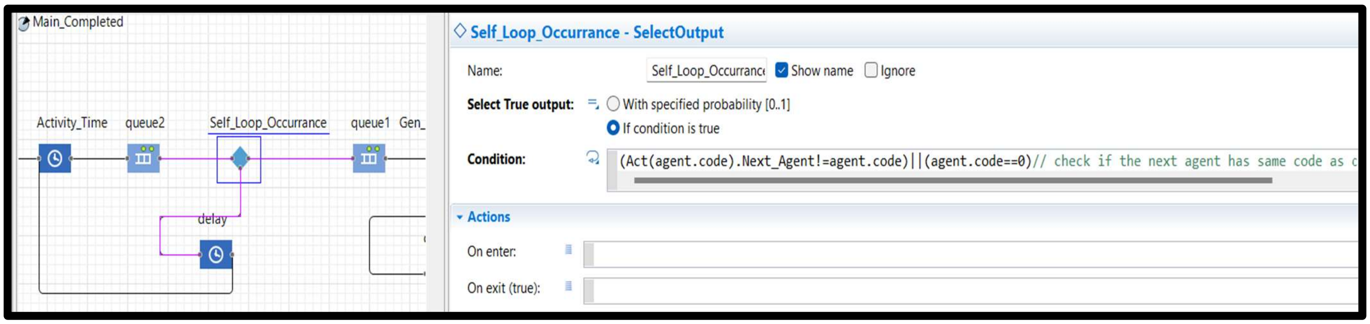


Figure 16 Condition of Self Loop Occurrence

For the CPM calculations, the model captures the ES of the self-looping activity as the start of the first iteration and EF as the exit time of the activity after all the activity iterations are completed, hence the new duration of the activity is always a multiple of the activity duration in case activity is carried out more than once

3.1.6 Activity General Loop

After the activity exits the previous node, the activity enters another select output node that evaluates whether the activity is part of a general loop referred to in Fig.20 by investigating whether the general loop boolean variable is True or False. If the activity is not a part of a general loop, it goes to the sink. If not, the activity goes back to the queue, which is the first block after the source, to start the activity evaluation process from the beginning. Similar to the case of the self-looping activity, the forward pass calculations in the case of the general loop are slightly modified, whereas the ES of the first activity of the loop is the start time of the first activity of the loop, and the EF of the time of completion of the first activity after the final iteration. In this manner, the first activity of the general loop represents the whole loop, where the duration of that activity is the duration of the whole loop, and calculations of all other activities in the loop are ignored.

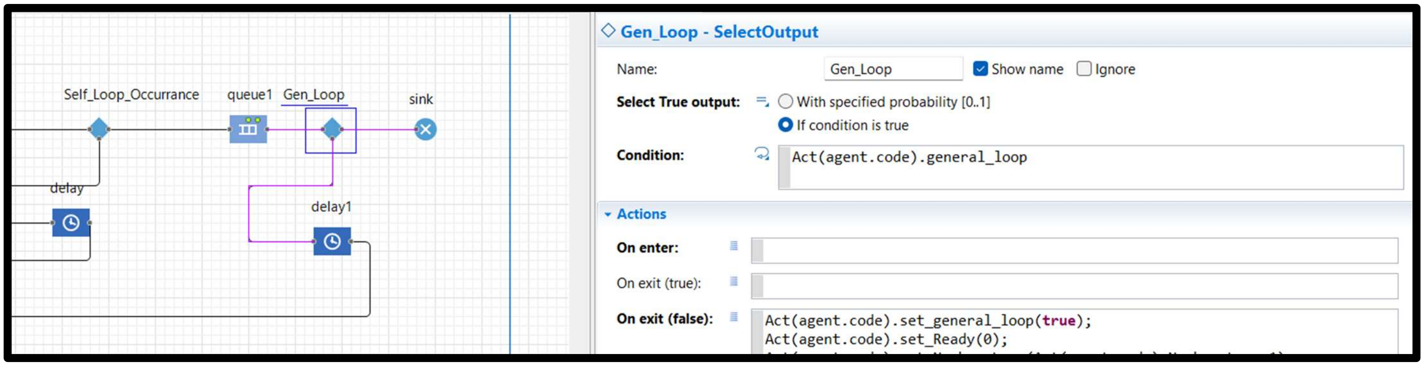


Figure 17 Condition for General Loop Occurrence

The final aspect of the model is the sink block, which is the last stop for each activity. This is the final destination for each activity after being performed and where the backward pass calculations for the CPM are performed after the simulation is concluded, as the code in Fig. 21 demonstrates.

```
if (last agent entered sink)
{
    finishSimulation();
    print Simulation time in output sheet;
    set late start of last agent= early start of last agent;
    set late finish of last agent= early finish of last agent;
    int Current_Agent = agent.code - 1;
    while (Current_Agent >= 0) //to loop through all the agents other than the last one
    {
        if (Act(Current_Agent).Completed == 1)// check if activity got executed
        {
             if(check for general loop case)
                    {
                        assign late finish and late start to early finish and early start of general loop activity
                        since its duration is already considered in its predecessor/successor
                    }
            else if (number of successor is 1)
            {
                assign late start of successor to late finish of current agent then subtract the duration to get the late start
            }
            if (if emitter is and)
            {
                if (number of successors is 2)
                {
                    assign the least late start from successors as this activity's late finish and then subtract the duration
                }
                if (number of successors = 3)
                {
                    assign the least late start from successors as this activity's late finish and then subtract the duration
                }
            }
            else if (emitter is OR)
            {
                if (number of successors is 2)
                {
                    if(only one successor is complete)
                    {
                        assign late start of this successor to current agent late finish and subtract duration to get late start
                    }
                    else if loop occurred ed take the late finish of the other task that is not equal to current task
                }
                   if (number of successors is 3)
                {
                    if(only one successor is complete)
                    {
                        assign late start of this successor to current agent late finish and subtract duration to get late start
                    }
                    else if loop occurred ed take the late finish of the other task that is not equal to current task
                }
            }
        }
        Current_Agent = Current_Agent - 1;
    }
}
```

Figure 18 Snippet from Pseudo code for Sink Block

Chapter 4 – Verification and Validation

4.1 Model Verification

Verification is a critical step in the simulation modeling process, ensuring that the model is correctly implemented and performs as intended. This chapter outlines the key steps involved in verifying the constructed model, with a focus on systematic approaches to identify and rectify errors.

Simulation model verification involves testing that the model's results accurately represent the conceptual model. This process ensures that the code and algorithms are correct and function as expected. Verification does not address whether the model accurately represents the real-world system, this is the role of validation through a case study, but rather focuses on the technical correctness of the model.

The first step in the verification process is a comprehensive review of the conceptual model. This step aims to ensure that the conceptual model is thoroughly understood and correctly translated into the simulation model. This involves checking the model's code for syntax errors and logical inconsistencies. AnyLogic has an embedded feature that displays errors, therefore through the development of the model, each error was resolved as the model progressed. Ensuring that the code executes correctly and adheres to the intended algorithms and rules is fundamental. Automated testing is also employed at this stage, where unit tests are used to verify the functionality of specific methods or modules. These tests confirm that each part of the code behaves as expected in isolation. This is conducted by examining and printing the parameters of agents before entering and upon exiting the block and examining the change in the parameters of the model. Values that underwent the examination are the activity code, duration, predecessors, and successors. These values were printed before the entry and on the exit in each block to isolate the effect of each block on the activity.

In order to ensure that the model behaves currently under extreme conditions, extreme input values were inserted in the model. This step observes how the model behaves in these scenarios and whether the outputs are reasonable or not. This allows the identification of weaknesses or limitations in the model during extreme conditions.

Sensitivity analysis is another verification technique that assesses the impact of changes in input parameters on the model's output. By systematically varying key input parameters and observing the resulting changes in the model's outputs, one can identify any unexpected behaviors or dependencies, indicating potential areas for further investigation. This analysis helps in understanding the model's sensitivity to different inputs and ensures its reliability.

Finally, trace analysis involves verifying the sequence of events and state changes in the model. Generating detailed traces of the model's execution, showing the sequence of events and state changes allows for a thorough review of these traces to ensure they align with the expected behavior of the model. This step is done by observing the output of the model and making sure that the activities of the project are conducted in a logical sequence in each scenario of the output scenarios.

4.2 Model Validation

After concluding the verification, the output of the model needs to be validated with a case study. The selected case study was a historical building where a retaining wall of a former seminary at the Wawel Royal Castle in Krakow, Poland was constructed in the 19th century(Radziszewska-Zielina et al., 2017).. Over time, micro piles were needed to support the structure's foundation along with other minor maintenance tasks. However, the wall's condition has worsened over the years and requires intervention. The initial scope work included the removal of the damaged section, applying appropriate waterproofing, and constructing new stairs that lead to the upper terrace, along with improvements to the pavement and exterior improvements to the wall. However, due to the nature of the project alternative scenarios where archaeological discoveries might occur during the digging phase might entail other technical requirements during the execution that might need to be taken into consideration during the project planning.

The selection of this case study was based on two factors. The first is that the reconstruction of a historical building is an example of a case where project planners have limited information during the planning phase. The second is the availability of a complete range of data for each regarding the project execution. This allows for model validation by comparing the predictions of the model with real life data from the project execution as well as the modeling techniques used by the authors (Radziszewska-Zielina et al., 2017).

The scope of the project included dismantling the damaged sections of the wall, installing a waterproofing layer for the structure, restoring its original geometry, constructing a staircase leading to the cellar and upper terrace, paving the surrounding area, and performing essential conservation work to refurbish the wall's exterior. At the beginning of the project Three activities are to begin simultaneously. The first is removing damaged bricks from the wall coping to determine the wall's condition in detail (W1). The second is performing manual earthworks for the stair foundation under archaeological supervision (W2). The third is preparing steel reinforcement for the stair foundations (W3). The successor of (W1) has three possible variants, depending on the wall's technical condition assessment. The first scenario involved reinforcing the wall with steel bars if interior delamination was found (W5). The second scenario, assuming the wall was in good condition, involved minor repairs to the wall coping (W6). If the wall was in very poor condition, the third scenario is for the wall to be demolished and rebuilt (W7). Following these activities, bricklaying was planned to raise the wall to the required height (W10), build smaller clinker brick walls on existing foundations (W20), and in parallel construct a vertical waterproofing layer on the main terrace side by injecting bentonite into the soil touching the wall (W11). After completing the waterproofing, its effectiveness would be tested by local uncovering (W12). Depending on the test results, the first variant involved making minor corrections if small breaches were found (W13), and repeating the process until the desired effect was achieved as Fig. 22 and 23 demonstrate, is an example of a general loop. The second occurs when the waterproofing is found to be adequate (W19), followed by a crystalline injection to create horizontal waterproofing (W22), with no further checks needed due to its high reliability. If the vertical waterproofing was ineffective, a decision would be made (W14) to use a different technology, which required manual excavation near the wall on the terrace side under archaeological supervision (W15), laying bitumen waterproofing (W17), and manually backfilling and compacting the excavation (W18). If archaeologists found remnants of earlier building foundations during the excavation, the waterproofing work would be paused for in-depth archaeological studies (W16). A similar issue with archaeological examinations (W4) might occur during the excavation for the stair foundation (W2) before creating the stair formwork (W8). Both Activities (W4) and (W16) are examples of self looping activities. Activities (W9) and (W21) represent placing reinforcement bars in formwork, pouring concrete, and backfilling the excavation near the foundations once the concrete is set, respectively. The rest of activities activities are concerned with the cobblestone pavement on the upper terrace near the retaining wall, where soil load-bearing capacity would be checked, and subsoil compacted for the planned subbase (W23). Depending on the results, two variants were considered: if the subsoil had sufficient load-bearing capacity, a standard aggregate subbase would be built (W25); if not, cellular geotextiles would reinforce the substructure (W24). Other soil reinforcement methods, like hydraulic binder stabilization or soil replacement, were deemed unsuitable due to the archaeological significance of the area. The final paving stage involved laying a limestone cobblestone surface on the subbase (W28). The last step included stonework: fastening sandstone cladding on the stairs (W26), mounting stone caps on the wall, and conserving the wall surface through sandblasting, or hydrophobic impregnation (W27).

Table 7 Activity input for case study activities

Activity Code	Activity Name	Description
0	Start	
1	W1	W1 Removal of Damaged stones & brick elements from coping wall
2	W2	W2 manual digging near the foundation of the stairs
3	W3	W3 Preparing steel reinforcement bars of the foundations
4	W4	W4 performing archaeological research
5	W5	W5 Reinforcing the wall with steel reinforcement bars
6	W6	W6 Small repairs to the top part of the wall
7	W7	W7 Dismantling the part of the wall above the foundations and performing bricklaying
8	*Dummy*	*Non construction activity is needed as a link for the sake of visual representation*
9	W8	W8 Erecting the formwork for the foundations and the structure of the stairs
10	W9	W9 Laying the previously prepared reinforcement bars of the foundations within the formwork followed by pouring the concrete mix
11	W10	W10 Laying additional masonry on the structure of the wall to bring it to the designed height
12	W11	W11 Forming a vertical waterproofing barrier from the side of the upper terrace by performing direct bentonite mass injection into the ground adjacent to the wall
13	W12	W12 Checking the effectiveness of the waterproofing made from bentonite mass by locally uncovering it

14	W13	W13 Small correction made to the bentonite mass waterproofing
15	W14	W14 Result of checking the bentonite mass waterproofing it was confirmed to be ineffective
16	W15	W15 Manual Excavation of the foundations of the wall performed under archeological supervision
17	W16	W16 Carrying out additional archaeological research
18	W17	W17 Erecting a vertical waterproofing layer of the wall from rolled materials
19	W18	W18 Filling in the excavations with manual compaction of the soil
20	W19	W19 The bentonite mass waterproofing has been properly made and should be left as is
21	W20	W20 Erecting smaller walls from clinker brick
22	W21	W21 Filling in the excavation, along with manual compaction of the soil
23	W22	W22 Erecting a horizontal waterproofing barrier within the wall through the use of crystalline injections
24	W23	W23 Analyze the load bearing capacity and compaction of the ground in order to design supports
25	W24	W24 Additional reinforcement of the subbase with the use of spatial geotextiles
26	W25	W25 Constructing a standard aggregate subbase as the ground has been determined to have sufficient load bearing capacity
27	W26	W26 fastening the stone cladding of the stairs
28	W27	W27 Fastening the stone coping of the wall and the conservation of its exterior through sandblasting, pointing a hydrophobic impregnation
29	W28	W28 Laying of Limestone pavement
30	W29	End

The same input in the case study by Radziszewska-Zielina et al (2017) is used in this model to compare the findings of this simulation technique with both the model used in their research as well as real execution time. The first aspect of the input is the definition of the activities and their description as per Table 7.

The next step is to construct the Generalized Activity on Node (GAN) network of the model based on the logical relationships and probable scenarios of execution as per Fig 21 that represent the GAN network adaptation, using the rules and guidelines discussed in the visualization aspect of this research, of the GERT network constructed by Radziszewska-Zielina et al. (2017) in their previous work that in Fig 22

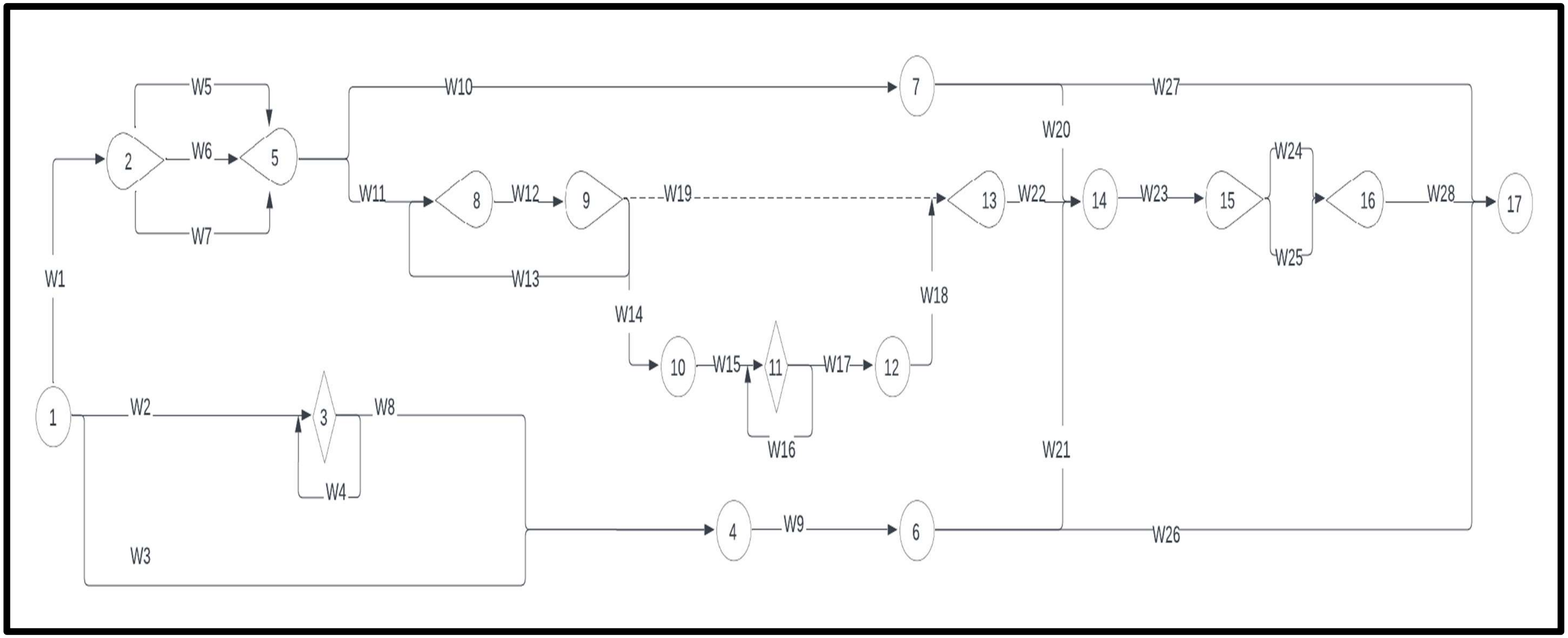


Figure 19 GERT representation for the case study (Radziszewska-Zielina et al. , 2017)

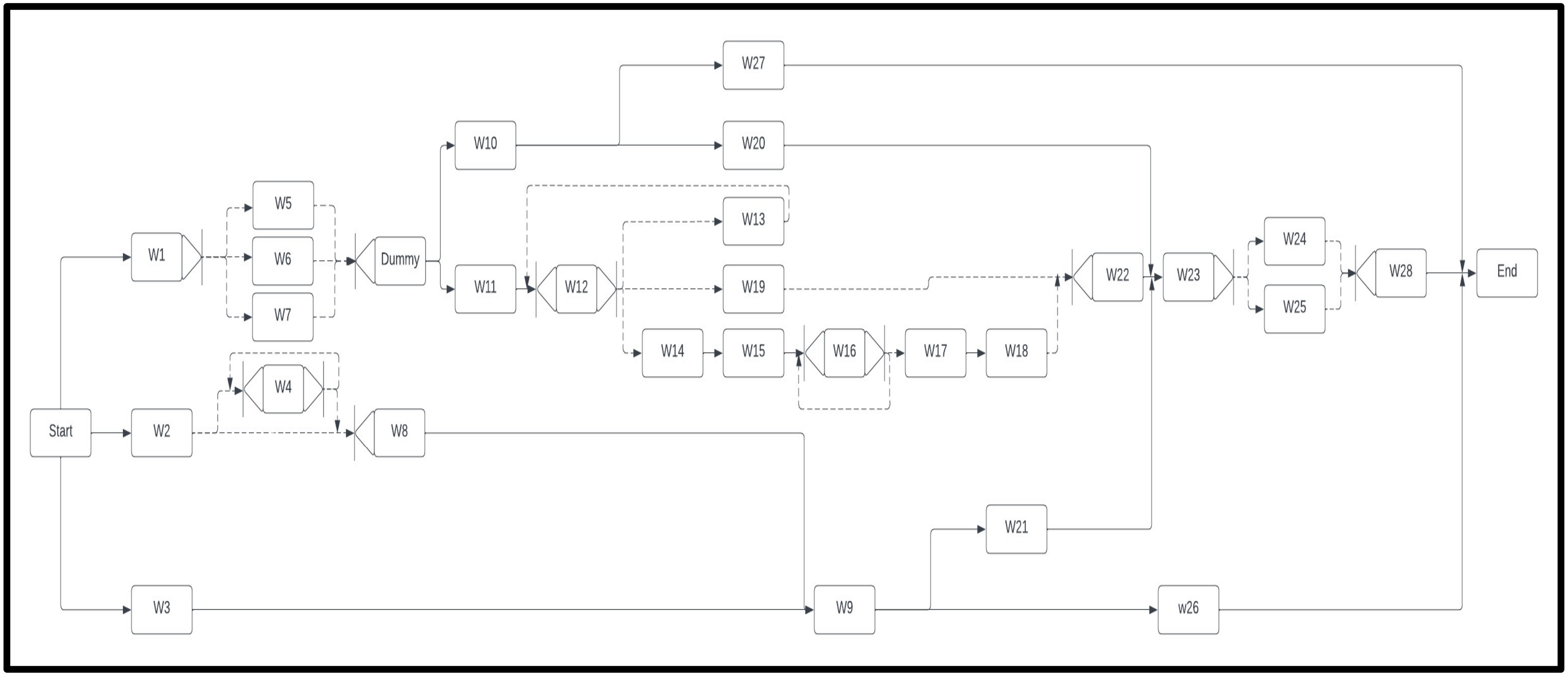


Figure 20 GAN/PDM network adaptation for the case study

4.3 Results Discussion

The findings of the simulation are discussed below, and it is divided into two sections. The first section examines each activity through all the simulations. The second section examines the possible durations of the project in each run and the corresponding scenario.

4.3.1 Analysis of Activity Parameters

After constructing the network logic through the GAN, the input duration of activity and probability of execution are obtained through a survey where ten experts with considerable experience in historical buildings provided their estimated value.

In order to compare different scenarios, the researcher obtained the results of 100 sequential runs, which were analyzed using Excel Pivot table feature to arrange the data and obtain the averages of the outputs in Table 8.

Table 8 Average values for the output parameters

Activity Name	ES	EF	LS	LF	Total Float
Start	0.0	0.0	0.0	0.0	0.0
W1	0.0	1.0	2.3	3.3	2.3
W2	0.0	2.3	6.9	9.2	6.9
W3	0.0	1.0	17.4	18.4	17.4
W4	1.8	10.3	6.0	14.5	4.2
W5	0.4	1.0	1.1	1.7	0.7
W6	0.2	0.6	0.7	1.1	0.5
W7	0.4	1.7	1.4	2.8	1.1
Dummy	3.3	3.3	5.6	5.6	2.3
W8	10.8	12.0	17.2	18.4	6.4
W9	12.0	13.2	18.4	19.6	6.4
W10	3.3	6.7	14.7	18.1	11.4
W11	3.3	4.3	6.1	7.1	2.8
W12	4.3	8.0	7.1	10.7	2.8
W13	-	-	-	-	-
W14	5.9	5.9	7.0	7.0	1.1
W15	5.9	7.4	7.0	8.6	1.1
W16	7.4	13.0	8.6	14.1	1.1
W17	13.0	14.0	14.1	15.1	1.1
W18	14.0	14.7	15.1	15.8	1.1
W19	2.1	2.1	3.7	3.7	1.6
W20	6.7	9.2	18.1	20.6	11.4

W21	13.2	14.2	19.6	20.6	6.4
W22	16.8	17.8	19.6	20.6	2.8
W23	20.6	21.6	20.6	21.6	0.0
W24	4.4	4.9	4.4	4.9	0.0
W25	17.2	18.6	17.2	18.6	0.0
W26	13.2	16.7	22.9	26.4	9.7
W27	6.7	11.9	21.2	26.4	14.5
W28	23.5	26.4	23.5	26.4	0.0
W29	26.4	26.4	26.4	26.4	0.0

Table 9 Project Completion time for 100 runs

Project Execution Time	Number Of Occurrences
14.6	4
15.3	4
17.1	3
18.1	6
19.7	1
20.5	5
21.0	7
21.8	4
22.5	4
22.9	4
23.0	3
23.1	2
23.4	1
24.5	7
25.9	3
27.0	6
28.6	5
29.0	4
30.6	7
31.7	1
33.1	3
35.5	7
39.7	1
40.7	1
43.3	2
45.7	1
46.6	1
48.5	1
52.0	1
62.6	1
26.4	**Average project execution time**

Table 9 demonstrates the project durations obtained from the 100 runs with the corresponding number of occurrences for each project duration. These values were used to obtain the average project execution in the 100 runs using the equation below as the weighted average.

$$\overline{D} = \frac{\sum_{i=1}^{n}(Di \cdot x_i)}{\sum_{i=1}^{n} x_i}$$

Were,

D_i = duration of project

X_i = number of occurrences

n = number of simulations

The findings above highlight the most critical activities with an average TF of zero are Start, W23, W24, W25, W28, and W29. The first activity is the Start of the project, which will always remain on the critical path by default. The activities from W23 to W25 represent the investigation of the load bearing of the compacted ground (W23) and, accordingly, determine whether additional reinforcement with geotextiles is needed (W24) or standard aggregate subbase will be used as the load bearing is sufficient (W25). Since these two are part of a stochastic node where either one of them will be executed and not the two together, and they have an average total float of zero, this means that whenever one of these two will be executed, it will lay on the critical path of the project. Other activities have an average float of 1.1, such as activities from W14 till W18 which is seen in Fig 20 are part of the stochastic branch where the checking of the bentonite waterproofing (W12) has proved to be ineffective and an alternative waterproofing method, which are activities W14-W18, is needed. The low average total float indicates that the activities that lay on this branch are part of the critical path whenever this branch is pursued. Even though these activities do not have an average TF of zero, the project managers need to consider these scenarios as they are considered near-critical activities that, when they occur, are part of the critical path. For activity W13, no CPM calculations were performed as this activity is part of the general loop, and all the parameters of the general loop are represented by W12 which takes into consideration the duration of the whole loop.

Table 10 Comparison of Project Completion time obtained using different simulation techniques.

Parameter	Deterministic model	Stochastic Fuzzy Logic Model Radziszewska-Zielina et al. (2017)	Discrete Event Modeling Average completion duration	Real world duration after completion
Project duration (Days)	20.55	24.14	26.4	35

The comparison in Table 10 demonstrates the effectiveness of the model in providing a close prediction to the actual real-world scenarios while also predicting the accurate real world duration of 35 days of execution, as shown in Table 8 where 35 days occurred seven times during the 100 runs.

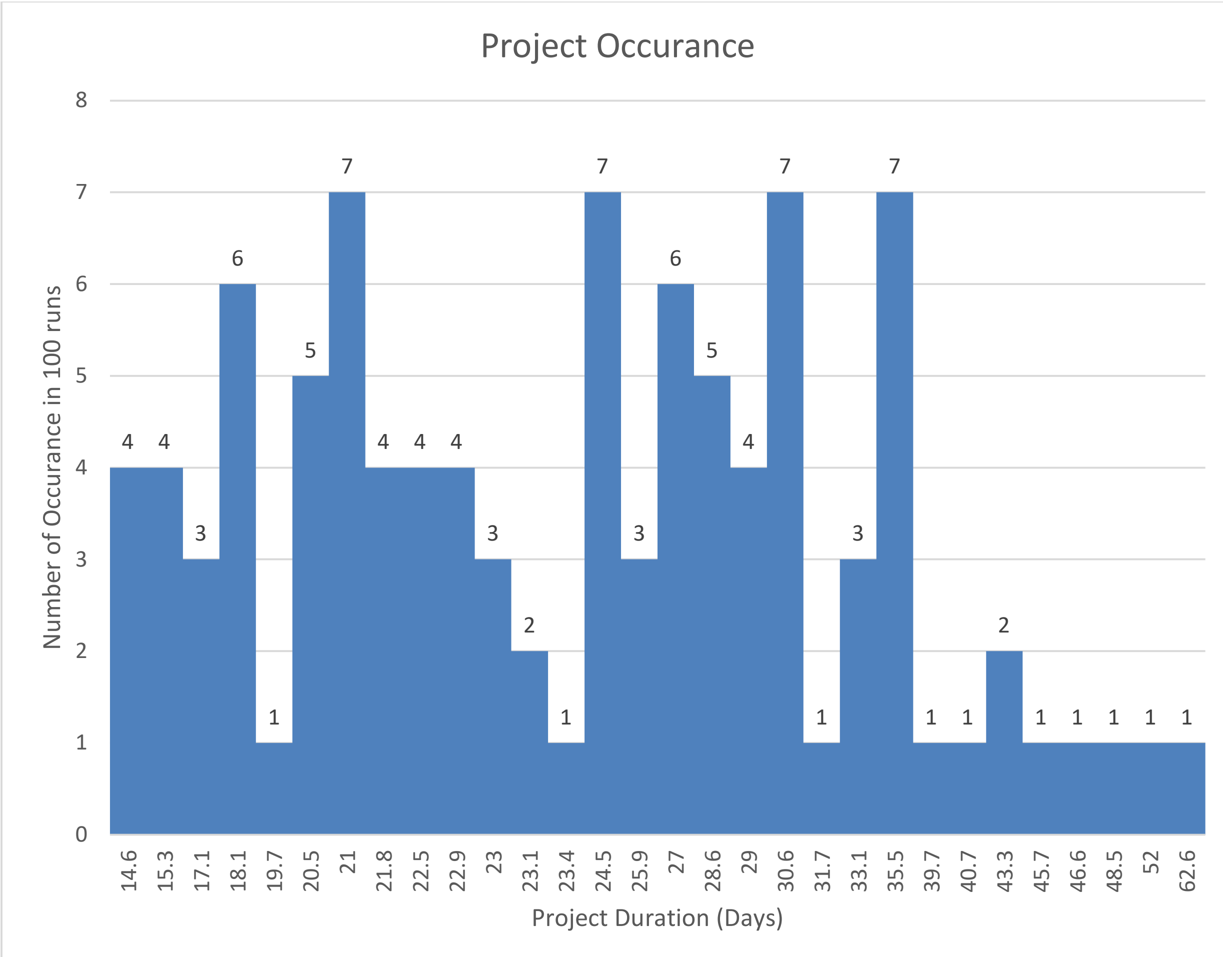


Figure 21 Number of Occurrence for each scenario in 100 Runs

4.3.2 Analysis of Different Scenarios of Execution

The advantage of this model is being able to analyze each probable scenario individually which would help project managers to analyze them which is not possible in other models due to the nature of the outputs. Fig.24 illustrates that during 100 runs 4 scenarios had occurred the most during the simulation. The analysis to be conducted will examine only the stochastic nodes where there are several methods of execution. The first scenario is the 21 days which is as in Fig.23

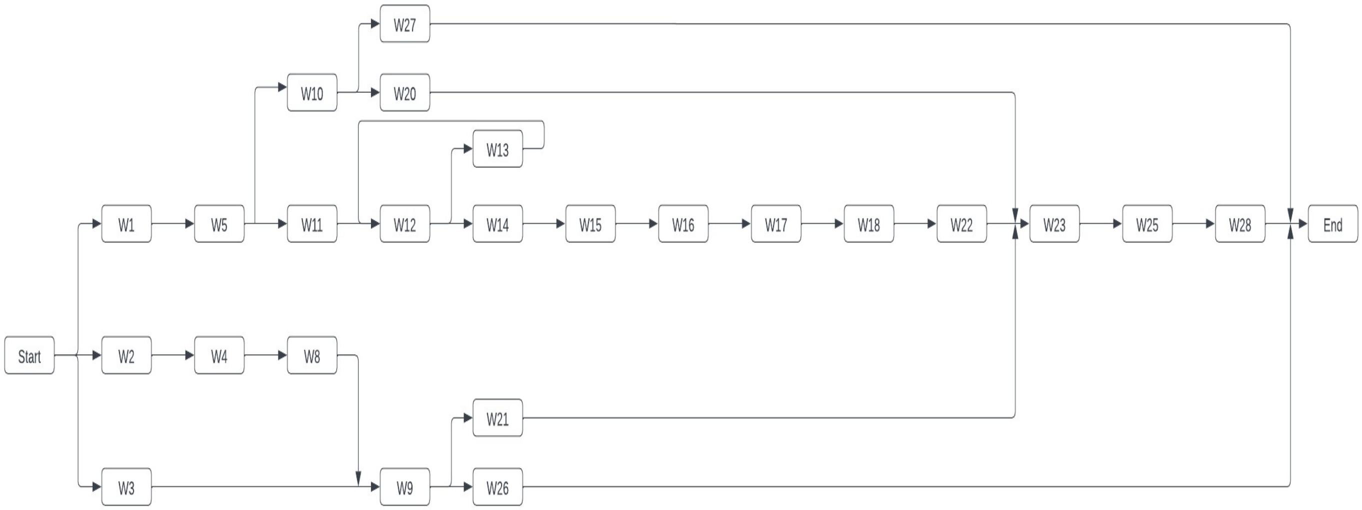


Figure 22 Scenario 1 with the highest occurrence in 100 simulation runs

When studying the first scenario in Fig. 25, it was observed that Activity (W5) is executed where additional bars are needed for the reinforced wall. Also, the general loop involving activities (W12) and (W13), where small corrections to the waterproofing needed to be executed, was realized before executing W14 which is an alternative scenario where another method of waterproofing is applied (W17). Also, during this scenario, W4 is realized where after digging near the foundation (W2), additional archaeological research was conducted before erecting the formwork for the foundation of the stairs (W8). The final stochastic node in this scenario is the realization of (W25)

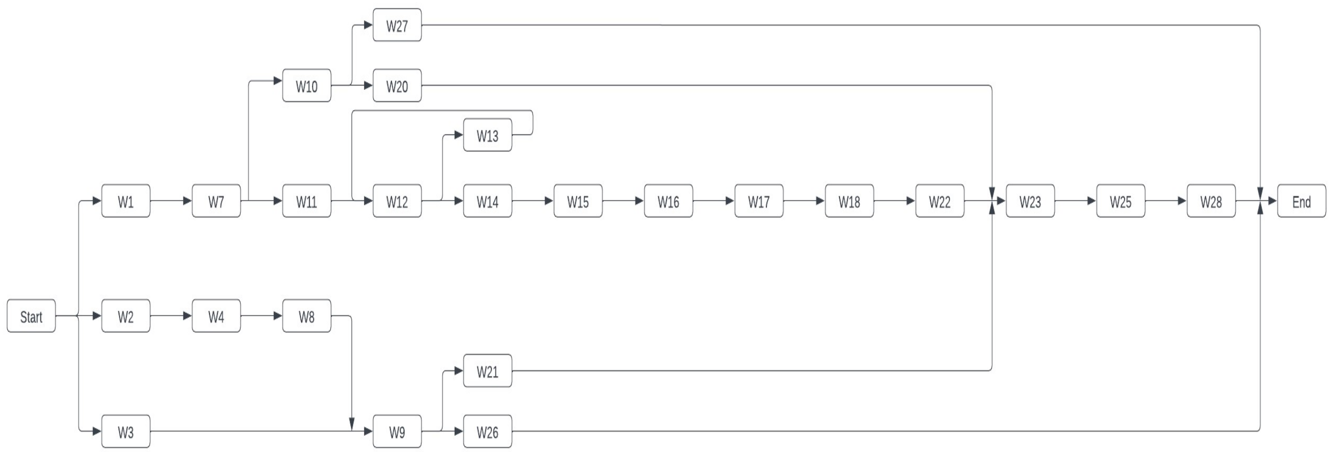


Figure 23 Scenario 2 with the highest occurrence in 100 simulation runs

The second scenario has a duration of 24.5 days demonstrated in Fig. 26 and is similar to the first scenario. However, in this scenario, instead of using additional reinforcement (W5), a part of the wall above the foundation had to be dismantled and bricklaying was performed (W7)

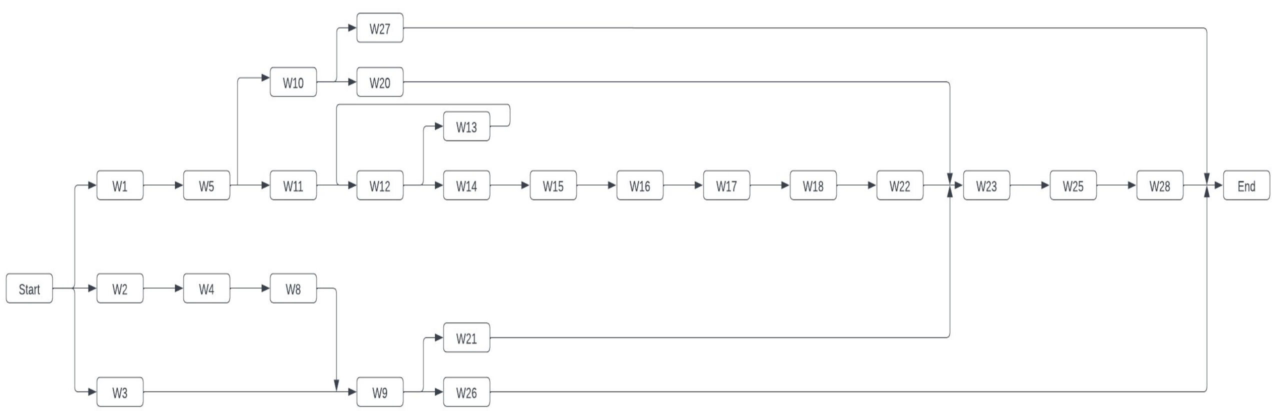


Figure 24 Scenario 3 with the highest occurrence in 100 simulation runs

The third scenario has a duration of 30.6 days demonstrated in Fig.27 is identical to the first scenario, however, the increase in the duration between those scenarios is attributed to the recurrence of the self-loop in (W4) which represents the archeological research that took longer duration that the first scenario.

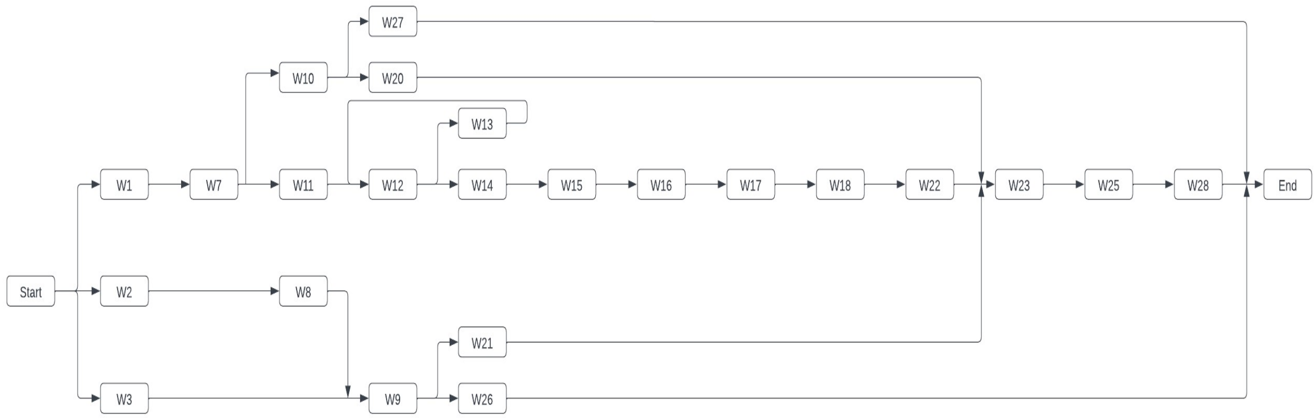


Figure 25 Scenario 4 with the highest occurrence in 100 simulation runs

The third scenario has a duration of 35.5 days demonstrated in Fig.28 is the closest duration to the actual project duration after execution and is similar to the first scenario. However, this increase in the duration of the total project is attributed to the increased duration of (W4) the additional archeological research as well as repeated attempts to apply the bentonite waterproofing material (W12) & (W13).

Chapter 5 Conclusion and Recommendations

5.1 Summary & Conclusion

This research has presented a scheduling technique that takes into consideration the probabilistic scenarios of execution through the integration of conditional relationships by utilizing DES where activities are represented by events that are being evaluated during the simulation through different blocks. This also allowed for the formation of new constituents of the activity network such as general loop and self looping activity, which allows for a more accurate and intuitive representation of networks. This research also demonstrates the use of GAN networks for construction projects that are widely familiar to project managers.

The novelty of this approach for modeling stochastic networks is that unlike other attempts to model, it provides more insight into other heuristics, such as the average values for the CPM forward and backward pass for each activity, and is not just concerned with the overall project completion durations. Moreover, the total float for the activity is calculated with reference to the results obtained in 100 runs and not just a certain simulation which provides the project managers with a view of criticality in different scenarios.

5.2 Research Outcomes & Contributions

The outcome of this research is highlighted in the ability to provide an enhanced prediction of the total project duration, where the other stochastic models estimated the project duration time as 24.14 days and deterministic models estimated 20.55 days, the model developed in this study estimated a project completion time of 26.4, the closest to the actual project duration of 35 days. Moreover, this research allowed for the closer examination of each activity’s criticality as well as being able to analyze all possible execution scenarios. The other aspect of this research is the demonstration of application for GANs networks in stochastic construction schedules, where this method allowed for a more familiar representation of the construction activities that are widely used.

5.3 Research Limitations and Future Work Recommendations

The primary drawback of this model is its restricted capacity for both predecessors and successors, with only three activities allowed in both cases. Moreover, the general loop can only consist of two activities. Another recommendation for future work is to explore the use of a combination of PDM relationships such as SS, FF, and the stochastic relationship introduced in this research to accurately capture complex relationships. Another parameter of this model that can be utilized is introducing the cost associated with each activity and analyzing it along with each scenario. The final recommendation is the introduction of stochastic activity durations (i.e., PERT) as the activity durations used in the project as deterministic.

References

Alzraiee, Hani, et al. (2015) Dynamic planning of construction activities using hybrid simulation. *Automation in Construction*, vol. 49, pp. 176–192, https://doi.org/10.1016/j.autcon.2014.08.011.

Botros, K. K., Tchir, W. J., Henderson, J. F., & Chmilar, B. (2010) Long-Range Facility Planning Based on Dynamic Programming for Optimum Combined Cost and Probability Paths. *Journal of Pipeline Systems Engineering and Practice*. https://doi.org/10.1061/(asce)ps.1949-1204.0000052

Cao, Yang, et al. (2015) n energy-aware, agent-based maintenance-scheduling framework to improve occupant satisfaction. *Automation in Construction*, *60*, 49–57. https://doi.org/10.1016/j.autcon.2015.09.002

Chehayeb, N. N., & AbouRizk, S. M. (1998). Simulation-Based Scheduling with Continuous Activity Relationships. *Journal of Construction Engineering and Management*, *124*(2), 107–115. https://doi.org/10.1061/(asce)0733-9364(1998)124:2(107)

Chen, S.-M., Griffis, F. H. (Bud)., Chen, P.-H., & Chang, L.-M. (2012). Simulation and analytical techniques for construction resource planning and scheduling. *Automation in Construction*, *21*, 99–113. https://doi.org/10.1016/j.autcon.2011.05.018.

Dawson, C. W., & Dawson, R. J. (1995). Generalised activity-on-the-node networks for managing uncertainty in projects. *International Journal of Project Management*, *13*(6), 353–362. https://doi.org/10.1016/0263-7863(95)00027-5

Elmaghraby, S. E. (1964). An Algebra for the Analysis of Generalized Activity Networks. *Management Science*, *10*(3), 494–514. https://doi.org/10.1287/mnsc.10.3.494

Hajdu, M., & Bokor, O. (2016). Sensitivity analysis in PERT networks: Does activity duration distribution matter? *Automation in Construction*, *65*, 1–8. https://doi.org/10.1016/j.autcon.2016.01.003

Hajdu, M. (2018). Survey of precedence relationships: Classification and algorithms. *Automation in Construction*, *95*, 245–259. https://doi.org/10.1016/j.autcon.2018.08.012

Hegazy, T., & Kamarah, E. (2008). Efficient Repetitive Scheduling for High-Rise Construction. *Journal of Construction Engineering and Management*, *134*(4), 253–264. https://doi.org/10.1061/(asce)0733-9364(2008)134:4(253)

Isidore, L. J., & Back, W. E. (2002). Multiple Simulation Analysis for Probabilistic Cost and Schedule Integration. *Journal of Construction Engineering and Management*, *128*(3), 211–219. https://doi.org/10.1061/(asce)0733-9364(2002)128:3(211)

Jabri, A., & Zayed, T. (2017). Agent-based modeling and simulation of earthmoving operations. *Automation in Construction*, *81*, 210–223. https://doi.org/10.1016/j.autcon.2017.06.017

Katsuragawa, C. M., Lucko, G., Isaac, S., & Su, Y. (2021). Fuzzy Linear and Repetitive Scheduling for Construction Projects. *Journal of Construction Engineering and Management*, *147*(3), 04021002. https://doi.org/10.1061/(asce)co.1943-7862.0001996

Kedir, N. S., Somi, S., Fayek, A. R., & Nguyen, P. H. D. (2022). Hybridization of reinforcement learning and agent-based modeling to optimize construction planning and scheduling. *Automation in Construction*, *142*, 104498. https://doi.org/10.1016/j.autcon.2022.104498

Khodakarami, V., Fenton, N., & Neil, M. (2007). Project Scheduling: Improved Approach to Incorporate Uncertainty Using Bayesian Networks. *Project Management Journal*, *38*(2), 39–49. https://doi.org/10.1177/875697280703800205

Kurihara, K., & Nishiuchi, N. (2002). Efficient Monte Carlo simulation method of GERT-type network for project management. *Computers & Industrial Engineering*, *42*(2-4), 521–531. https://doi.org/10.1016/s0360-8352(02)00050-5

Lee, D.-E. (2005). Probability of Project Completion Using Stochastic Project Scheduling Simulation. *Journal of Construction Engineering and Management*, *131*(3), 310–318. https://doi.org/10.1061/(asce)0733-9364(2005)131:3(310)

Lee, D.-E., & Arditi, D. (2006). Automated Statistical Analysis in Stochastic Project Scheduling Simulation. *Journal of Construction Engineering and Management*, *132*(3), 268–277. https://doi.org/10.1061/(asce)0733-9364(2006)132:3(268)

Lee, D.-E., Yi, C.-Y., Lim, T.-K., & Arditi, D. (2010). Integrated Simulation System for Construction Operation and Project Scheduling. *Journal of Computing in Civil Engineering*, *24*(6), 557–569. https://doi.org/10.1061/(asce)cp.1943-5487.0000061

Liu, L., Burns, S. A., & Feng, C.-W. (1995). Construction Time-Cost Trade-Off Analysis Using LP/IP Hybrid Method. *Journal of Construction Engineering and Management*, *121*(4), 446–454. https://doi.org/10.1061/(asce)0733-9364(1995)121:4(446)

Lu, M., Lam, H.-C., & Dai, F. (2008). Resource-constrained critical path analysis based on discrete event simulation and particle swarm optimization. *Automation in Construction*, *17*(6), 670–681. https://doi.org/10.1016/j.autcon.2007.11.004

Lucko, G., Su, Y., & Thompson, R. C. (2021). Theoretical quantification of ripple effect of delays in network schedules via activity cruciality. *Automation in Construction*, *129*, 103789. https://doi.org/10.1016/j.autcon.2021.103789

Ma, G., Hao, K., Xiao, Y., & Zhu, T. (2019). Critical Chain Design Structure Matrix Method for Construction Project Scheduling under Rework Scenarios. *Mathematical Problems in Engineering*, *2019*, 1–14. https://doi.org/10.1155/2019/1595628

MacCrimmon, K. R., & Ryavec, C. A. (1964). An Analytical Study of the PERT Assumptions. *Operations Research*, *12*(1), 16–37. https://doi.org/10.1287/opre.12.1.16

Moselhi, O., & El-Rayes, K. (1993). Scheduling of Repetitive Projects with Cost Optimization. *Journal of Construction Engineering and Management*, *119*(4), 681–697. https://doi.org/10.1061/(asce)0733-9364(1993)119:4(681)

Nasir, D., McCabe, B., & Hartono, L. (2003). Evaluating Risk in Construction–Schedule Model (ERIC–S): Construction Schedule Risk Model. *Journal of Construction Engineering and Management*, *129*(5), 518–527. https://doi.org/10.1061/(asce)0733-9364(2003)129:5(518)

Naticchia, B., Carbonari, A., Vaccarini, M., & Giorgi, R. (2019). Holonic execution system for real-time construction management. *Automation in Construction*, *104*, 179–196. https://doi.org/10.1016/j.autcon.2019.04.018

Pollack-Johnson, B., & Liberatore, M. J. (2005). Project Planning under Uncertainty Using Scenario Analysis. *Project Management Journal*, *36*(1), 15–26. https://doi.org/10.1177/875697280503600103

Radziszewska-Zielina, E., Śladowski, G., & Sibielak, M. (2017). Planning the reconstruction of a historical building by using a fuzzy stochastic network. *Automation in Construction*, *84*, 242–257. https://doi.org/10.1016/j.autcon.2017.08.003

Robinson, D. R. (1975). A Dynamic Programming Solution to Cost-Time Tradeoff for CPM. *Management Science*, *22*(2), 158–166. https://doi.org/10.1287/mnsc.22.2.158

Sadeghi, N., Fayek, A. R., & Gerami Seresht, N. (2016). A Fuzzy Discrete Event Simulation Framework for Construction Applications: Improving the Simulation Time Advancement. *Journal of Construction Engineering and Management*, *142*(12), 04016071. https://doi.org/10.1061/(asce)co.1943-7862.0001195

Su, Y., Isaac, S., & Lucko, G. (2018). Integrated Temporal-Spatial Model for Construction Plans with Boolean Logic Operators. *Journal of Construction Engineering and Management*, *144*(4). https://doi.org/10.1061/(asce)co.1943-7862.0001450

Taroun, A. (2014). Towards a better modeling and assessment of construction risk: Insights from a literature review. *International Journal of Project Management, 32*(1), 101–115. https://doi.org/10.1016/j.ijproman.2013.03.004

Tomczak, M., & Piotr Jaśkowski. (2024). Customized particle swarm optimization for harmonizing multi-section construction projects. *Automation in Construction*, *162*, 105359–105359. https://doi.org/10.1016/j.autcon.2024.105359

Zhou, J., Love, P. E. D., Wang, X., Teo, K. L., & Irani, Z. (2013). A review of methods and algorithms for optimizing construction scheduling. *Journal of the Operational Research Society*, 64(8), 1091–1105. https://doi.org/10.1057/jors.2012.174