Using Monte Carlo Simulation to Evaluate Performance of Forecasting Models in Project Control

by

Rana Ead

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

in

Construction Engineering and Management

Department of Civil and Environmental Engineering
University of Alberta

ABSTRACT

The construction sector is one of the largest in the world economy. Billions of dollars are spent every year to execute and deliver projects serving the public. However, internal and external factors – such as unexpected weather, financial difficulties, manpower shortages, and excessive change orders – can impact the performance of these projects and cause deviations from planned performance. Consequently, researchers are constantly developing methods to improve project performance during execution. Although the models are developed using advanced techniques, the validation techniques applied to assess their accuracy are flawed. One example of this can be found in schedule- and cost-forecasting models. These models are crucial in ensuring that project progress is regularly tracked and reviewed, areas requiring changes are identified, and the subsequent changes are initiated accordingly. However, most researchers use validation techniques that rely on comparing the developed method to other well-known methods using actual completed project data. This has several drawbacks that render the validation experiments inconclusive.

Accordingly, this thesis focuses on designing a Monte Carlo simulation-based empirical experiment to improve the process of assessing and comparing the effectiveness of different methods, with a focus on forecasting methods. The experiment involves generating a large sample set of random activity-on-node networks that resemble actual construction projects. The actual cost and schedule performance of these projects is simulated using a Markov chain approach that mimics the uncertainty affecting the execution of these projects. To test the experiment, four forecasting methods are applied to predict the final project cost and duration given the randomly-generated planned and actual performance data: earned value analysis (EVA), earned schedule (ES), non-linear regression Gompertz growth model (NLR-GGM), and the Kalman filter forecasting model (KFFM). These methods are evaluated in terms of forecasting accuracy,

timeliness, and stability under varying project complexities and lengths. By providing an in-depth analysis, researchers and practitioners can develop an enhanced understanding and insight into these methods. This research contributes to the current state-of-the-art by developing a means for validating and assessing cost- and schedule-forecasting models in a controlled, unbiased, and simulated environment.

ACKNOWLEDGMENTS

First and foremost, I would like to thank God Almighty for giving me the knowledge, strength, and opportunity to undertake this research study and to persevere and complete it satisfactorily. Without his blessings, this achievement would not have been possible.

I owe a great deal of gratitude to my supervisor, Dr. Simaan AbouRizk, for his constant guidance and support throughout these two years and for being an impeccable role model. I have learned a great deal from him, on both a professional and personal level, and for that I am extremely grateful. I would like to extend my appreciation to Stephen Hague for always offering any help or assistance that I could need. His advice has proven monumental towards the success of my research, and it is whole-heartedly appreciated. I wish to also express my gratitude to Mickey Richards for her effort and assistance with manuscript editing.

The physical contributions of the National Science and Engineering Research Council (NSERC) and Alberta Innovates are truly appreciated. Without their funding and support, this research would not have achieved its goal.

I am forever indebted to my family without whom this research would not have been possible. To my loving husband, Hady, for supporting me and for keeping me sane throughout this process. I feel lucky to have gone through this journey with him by my side every step of the way. To my siblings, Maha and Ahmed, for always inspiring me and keeping me on the track to complete when I needed the motivation. Finally, to my parents, for being my mentors and for always believing in me more than I ever have myself. Thank you for your constant motivation, encouragement, and prayers, and for always showing me the light at the end of the tunnel.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Explanation
AC	Actual cost
ACWP	Actual cost of work performed
AON	Activity-on-node
AP	Actual progress
BAC	Budget at completion
BAF	Bayesian adaptive forecasting
BCWP	Budgeted cost of work performed
BCWS	Budgeted cost of work scheduled
CDF	Cumulative density function
CPI	Cost performance index
CPM	Critical path method
CV	Coefficient of variation
DB	Design build
EAC	Estimate at completion
EDAC	Estimated duration at completion
EPC	Engineering procurement construction
ES	Earned Schedule
EV	Earned value
EVA	Earned value analysis
FF	Finish-to-finish
FL	Fuzzy logic
FS	Finish-to-start
GA	Genetic algorithms
GEAC	Gompertz estimate at completion
GGM	Gompertz growth model

ISM	Interpretive structural modeling
KFFM	Kalman filter forecasting model
MAE	Mean absolute error
MAPE	Mean absolute percentage error
MC	Monte Carlo
MPE	Mean percentage error
MSE	Mean square error
NLR-GGM	Non-linear regression Gompertz growth model
NN	Neural network
PERT	Program evaluation and review technique
PMBOK	Project Management Body of Knowledge
PV	Planed value
RCPSP	Resource-constrained multi-project scheduling problem
RMSE	Root mean square error
SF	Start-to-finish
SPI	Schedule performance index
SPI_t	Schedule performance index time
SPSS	Stochastic project scheduling simulations
SS	Start-to-start
SSC	Stochastic S-curves
SV	Schedule variance
TPC	Total project cost
TPD	Total project duration
TV	Time variance

CHAPTER 1: INTRODUCTION

1.1 Background and Problem Statement

The construction sector is one of the largest in the world economy. Billions of dollars are spent every year to execute and deliver projects serving the public. However, internal and external factors – such as unexpected weather, safety factors, financial difficulties, manpower shortages, and excessive change orders – can impact every aspect of a construction project (Sweis et al. 2014). Consequently, productivity and actual performance in terms of schedule, cost, quality, safety, etc. push researchers to develop methods that improve performance during the execution phase. Identifying deviations from planned performance in a timely manner can control and mitigate their effects, allowing project objectives to be achieved. Although these methods are developed using advanced techniques, the validation processes applied to assess their efficacy are often unreliable. One example can be found in project control models. These models are crucial in ensuring that project progress is regularly tracked and reviewed, areas requiring changes are identified, and the subsequent changes are initiated accordingly. The Project Management Body of Knowledge (PMBOK) Guide describes several monitoring and control activities that can be used to achieve the above. These include: (1) comparing actual outcomes with planned outcomes and reporting performance; (2) forecasting project outcomes, assessing overall performance, and determining whether and what action needs to be taken; (3) ensuring deliverables agree with previously approved requirements and acquiring sign offs on these deliverables; (4) managing risks; (5) managing vendors and contracts. This thesis is concerned with the second activity: forecasting models and the validation processes involved in the development of these models.

Through forecasting, project managers translate actual execution data from the construction site into knowledge that can be used in making correct management decisions. Several forecasting methods have been developed over the years to improve the accuracy of predictions and provide more timely forecasts. Earned Value Analysis (EVA), a well-established forecasting method, is the most commonly known and used in practice for cost- and schedule-forecasting. However, the deterministic nature of this technique and its assumption that all activities, critical and non-critical, will equally affect performance can often provide misleading outcomes (Lipke, Earned Schedule, 2006). Thus, researchers have proposed extensions to traditional EVA or have used other techniques including Bayesian inference, Markov chains, regression, and artificial intelligence to overcome these limitations. Although researchers often conclude that the developed methods theoretically provide more accurate, stable, or timely forecasts when compared to other methods, these conclusions are based on inductive reasoning. In other words, broad generalizations on the performance of these methods are made from specific observations using a small sample of actual projects. Although the outcomes are true for the tested sample, the main drawback of this approach is that when expanded to larger dataset, the conclusions do not necessarily hold true. In addition, the use of real project data for validation and comparison purposes has been criticized as having several shortcomings. The number of real projects available are limited, and more importantly, project outcomes are affected by managerial decisions during execution (B. Kim 2007), rendering unreliable validation experiments.

Accordingly, this research proposes an empirical Monte Carlo (MC) simulation-based experiment to validate forecasting methods in construction management. MC simulation is a type of simulation that computes the results of an experiment using statistical analysis with repeated random sampling. It is particularly useful in analyzing scenarios with uncertainty and providing a probabilistic investigation of different situations. Hence, it is closely associated with random experiments in which the exact outcomes are unknown. MC simulation has been used in several domains including mathematics, engineering, finance, etc. In construction, it has proven particularly useful in risk management applications. The methodology involved in this research applies MC simulation by generating a set of random construction project baselines using several input variables represented by statistical distributions. As discussed above, actual performance during project execution often deviates from planned performance; this randomness and uncertainty in performance is modeled using a Markov process to create actual progress variations of each baseline plan. As a result, a large sample dataset has been created that is representative of the entire solution space and can be used in validating forecasting methods empirically.

1.2 Objectives

The main objectives of this research are:

1. Providing a method for creating random project networks and actual progress variations using MC simulation and Markov chains.

2. Developing an empirical experiment to validate different applications in construction management in a reliable and efficient way.

3. Understanding the limitations of current practices in evaluating the performance of schedule- and cost-forecasting models in the construction domain and applying the developed experiment to overcome these limitations

1.3 Expected Contributions

The expected academic contributions of the proposed research can be summarized as follows:

- Developing a methodology for creating project baseline schedules that resemble actual construction projects using MC simulation, and generating realistic sample progress variations using Markov chains to mimic the randomness and uncertainty affecting the execution of these projects.

- Developing a means for researchers to validate and asses the accuracy of developed cost and schedule forecasting models in a controlled, unbiased, and simulated environment that can be expanded by adding resources, risks, interruptions etc. for use in other applications.

The expected industrial contributions of the proposed research can be summarized as follows:

- Providing project managers with enhanced understanding and better insight into forecasting models used to predict final schedule and cost deviations.

- Improving the overall execution and performance of construction projects by advancing the use of forecasting models in practice.

1.4 Research Methodology

The research methodology was conducted in the following manner:

- Conduct a literature review to identify the various forecasting methods and techniques developed by researchers, the methods used to evaluate the performance of these forecasting techniques and their limitations, and the application of MC simulation in different fields.

- Design a MC simulation experiment to evaluate the performance of different forecasting theories in terms of accuracy, timeliness, and stability. This includes an outline of the hypothesis to be tested.

- Validate the methodology for creating random project networks by relying on construction experts' opinions, and refine it accordingly.

- Automate the experiment on four forecasting methods: EVA, Earned Schedule (ES), Kalman filter forecasting model (KFFM), and non-linear regression model.

- Analyze the sample data and results statistically, and assess the plausibility of the stated hypotheses.

- Provide a method for researchers to evaluate developed prediction models and provide recommendations to enhance the use of forecasting in the construction industry.

1.5 Thesis Organization

The thesis is organized as follows: Chapter 2 provides a summary of the literature. Chapter 3 provides a detailed design of the MC experiment, including the procedure for generating network topology randomly, simulating different variations of actual progress, the dependent and independent variables involved, and the overall automation of the experiment. Chapter 4 provides a case study where the designed experiment is tested with several forecasting methods, and the results are statistically analyzed. Chapter 5 provides conclusions and recommendations for future researchers and practitioners.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

Project monitoring in construction involves comparing plans to actual progress on the construction site, forecasting future progress, identifying deviations, and analyzing these deviations to pinpoint their causes. Accurate forecasting of project performance is critical to the success of construction projects. The earlier deviations from planned performance are detected, the easier and less costly it is to reduce their impact on project goals. Accordingly, several researchers have focused on developing models to improve project performance in terms of time, cost, quality, safety, etc. Over the years, traditional EVA indices (e.g. cost performance index (CPI), schedule performance index (SPI), or schedule performance index time (SPI_t)) have been commonly used for performance evaluation and forecasting and are known to improve schedule, cost, and technical performance of projects (F. Anbari 2012). However, EVA has been criticized as having a number of shortcomings. Accordingly, other techniques and frameworks have been proposed to overcome the existing limitations of EVA for more accurate forecasting of project duration and estimate at completion (EAC).

With the emergence of several forecasting methods, developing a method to accurately and effectively evaluate the performance of these methods becomes important. For practitioners to be able to rely on a certain method during the execution of their project, its advantages and disadvantages must be clearly and objectively known beforehand. This chapter reviews a number of existing forecasting techniques used to predict schedule and cost performance in construction projects. Next, the practices previously relied on to evaluate the performance of these prediction frameworks are examined and the existing limitations are highlighted. Finally, by summarizing the current literature and identifying the gaps in the existing knowledge, the basis for this research is laid out.

2.2 Project Performance Forecasting

Forecasting is the process of using information about the current status of the project and extrapolating to visualize performance at the end of the project. Forecasts are generally used in construction project management for predicting overall project cost or duration, quality of project deliverables, risk occurrence, resources required, or any combination of these factors. Forecasting provides the project management team with valuable knowledge that allows them to proactively manage their project and resources, allowing for better overall performance. Understanding schedule performance and risks related to activities that lie on the critical path of the project is essential to forecasting project duration and cost. However, resource-constrained projects are often difficult to predict since lack of resources can causes changes in the critical path (Shaikh 2017). Accordingly, expert judgement is an important factor for the accuracy of these forecasts.

Several frameworks have been developed to predict performance in terms of some or all of the factors mentioned above, and these are based on one or more techniques, including earned-value based methods, Markov chains, Bayesian inference, fuzzy logic, artificial intelligence, curve fitting and regression analysis, and miscellaneous techniques, as discussed in detail below.

2.2.1 Earned Value Analysis-Based Methods

The EVA technique was developed as a method for monitoring project progress during execution. The three main elements of project management – scope, cost, and schedule management – are integrated through this method (F. T. Anbari 2003). EVA calculates cost and schedule indices and variances and uses these values to forecast the performance of the project at the estimated completion date. By highlighting the possible need for corrective actions, project managers can enhance productivity and adjust strategies based on the calculated trends. Despite EVA's popularity, one of its biggest limitations is that the schedule variance (SV) and SPI are equal to 0 and 1, respectively, when evaluated at the completion of a project that is behind schedule (Lipke, Earned Schedule, 2006). Furthermore, EVA's assumption that performance is "static" results in inaccurate forecasts (Du, Kim, and Zhao 2016).

Over the years, there have been several extensions to the calculation of the cost EAC using EVA (Vignesh and Sowmya 2013). Each method is based on a number of assumptions. These methods are summarized below, and EVA will be discussed further in Chapter 4.

1. The first method assumes that the rate of spending until this point will continue into the future, and there will be no variance in the budget at completion (BAC). The EAC is

calculated using the cumulative value of the CPI (measures the cost performance by comparing the budget for the amount of work completed to the actual cost. A value greater than one indicates underrun and a value less than one indicates overrun).

$$EAC = \frac{BAC}{CPI_{cum}} \tag{1}$$

2. The next method assumes that the average rate of spending in the previous three months will continue into the future. EAC is calculated using the average CPI from the previous three months, the actual cost (AC), and the earned value (EV). In some cases, the six-month average cost performance is used.

$$EAC = AC + \frac{BAC - EV}{CPI_{3\ months}} \tag{2}$$

3. Similar to the previous calculation, this equation assumes that the cumulative average rate of spending will continue for the remaining duration.

$$EAC = AC + \frac{BAC - EV}{CPI_{cumulative}} \tag{3}$$

4. Another method assumes that work from the current point in the project will progress as previously planned. The cost variance (CV) that was previously recorded will not materialize in the future.

$$EAC = BAC + CV \tag{4}$$

5. As cost and schedule performance are both assumed to have an effect on the cost at completion, this method uses the CPI and SPI (measures the schedule performance by comparing the budget for the amount of work completed to the budget for the work planned to be completed. A value greater than one indicates underrun and a value less than one indicates overrun). The user selects the weight (α, β) to be placed on each, where the sum of the weights is equal to 1.

$$EAC = AC + \frac{BAC - EV}{\alpha \times CPI + \beta \times SPI} \quad (5)$$

6. In this method, cost EAC is also impacted by a combination of the cost and schedule performance indices. Similar to method 3, rather than using the cumulative values of the CPI and SPI, the average six-month cost and schedule performance indices are used.

$$EAC = AC + \frac{BAC - EV}{CPI_{cum} + SPI_{cum}} \quad (6)$$

2.2.2 Markov Chains

A stochastic process is a mathematical model that evolves probabilistically over time. Markov chains, named after the Russian mathematician A. A. Markov (1856–1922), are a special type of stochastic process in which the outcome of an experiment depends only on the current state of the system and not on any of the past states (Markov 2003). Applications of Markov chains are especially common in infrastructure management. For example, Robelin and Madanat (2007) used Markov chains for bridge deck maintenance and replacement optimization. Furthermore, Markov chains have been used for deterioration predictions of wastewater systems (Baik, Jeong, and Abraham 2006), timber bridge elements (Ranjith et al. 2013), pavement networks (Li, Xie, and Haas 1996; Ortiz-García, Costello, and Snaith 2006), concrete bridges, and sewers (Wirahadikusumah, Abraham, and Castello 1999). Markov chains have also been widely used in other disciplines such as medicine and chemistry (Markov 2003).

Several researchers have also applied Markov chains in project performance prediction. Nasser (2005) developed a probablistic forecasting model using dynamic Markov chains to address the stochastic nature of project performance and incorporate past performance, future corrective actions, and expert judgement into the forecasts. In his model, he assumed that project performance can occupy five possible states based on the performance index: outstanding, exceeds target, within target, below target, and poor performance. He relied on the "memoryless" property of the Markov process to develop the transition matrix from one state to another. The project was divided into equal time increments that reperesent the times when the performance is evaluated. The initial state matrix was calculated based on historical project data, and the transition matrix was calculated with the assumption that the performance index is proportional to the project S-curves, by revenue, man-hours, or cost. The results indicated that the model is able to accurately predict the stochastic performance of a project due to its capability of capturing uncertainity and time-dependence. However, it required a large amount of data regarding actual project performance to produce reliable results.

Du et al. (2016) combined Markov chain and MC simulation for cost projection. The probability distribution of the CPI for each project period was simulated using the model, and the final cost was calculated by aggregating all simulated period costs. A case study about a coal fire power plant in the U.S. was presented. Historical cost performance data from 60 engineering, procurement construction (EPC) projects was used to fit a distribution of period performance factors (PF). Based on the generated distribution, 20 percentile values were chosen to calculate the transition matrix, i.e. the probability of transitioning from one period PF to another, and consequently the cumulative density functions (CDF) were obtained. Random numbers were generated using inverse transform methods following the CDFs, and the MC simulation was repeated 1000 times to improve statistical results. Finally, the results were compared with the results of EVA and indicated that Markov chains provided a more stable EAC prediction for the tested case study. The prediction converged to the actual value at completion more quickly than EVA, which wavered erratically from month to month.

2.2.3 Bayesian Statistics

Bayesian inference is based on Bayes' theorem and can be used to deduce the parameters of a probability distribution, as well as update these parameters when new data becomes available (Bartlett, 2018). Bayes' theorem is represented by equation (7), with theta (θ) representing the distribution parameters to be calculated, and $y = \{y_1, y_2, y_3, \ldots y_n)$ representing the data or set of observations available (Bartlett, 2018).

$$P(\theta|Y) = \frac{P(Y|\theta)P(\theta)}{P(Y)} \tag{7}$$

Where,

$P(\theta|Y)$ = the conditional probability that event A occurs given that event B has occurred, also called the posterior distribution.

$P(Y|\theta)$ = the likelihood distribution, represents the "conditional density of the data given the parameters". (Bartlett, 2018)

$P(\theta)$ = represents the prior distribution, i.e. the belief about the true values of the parameters, which may be based on expert judgement or historical data.

The Bayesian concept can be used in updating the posterior distribution in real-time as new data becomes available, as well as in integrating actual and historical performance data for more accurate forecasting. The iterative process uses the posterior distribution as the new prior distribution, which is updated at each iteration with the likelihood distribution derived from the new data. This process can be useful in updating forecasts of cost and duration at completion to ensure successful delivery of construction projects.

Kim and Reinschmidt (2009) developed a probabilistic forecasting method based on Bayesian inference with the beta distribution that provides confidence bounds on project duration and EAC. The methodology involved generating the beta S-curve parameters (α, m and *T*) as prior distributions based on the prior information available and the level of confidence of that information (determined by the decision maker). Then those parameters were updated using Bayesian inference. A numerical example was presented to validate the framework. The Bayesian betaS-curve method can be used to integrate planned estimates of duration with actual performance, resulting in more accurate forecasts compared to the EVA and earned schedule methods. This method does not require detailed activity level performance, unlike the critical path method (CPM).

Later, Kim and Reinschmidt (2011) proposed a probabilistic cost forecasting model that uses Bayesian inference and Bayesian model averaging techniques for adaptively combining historical project data (outside view) with detailed information about the specific project (inside view). The framework consisted of two phases, and the first was cost Bayesian adaptive forecasting (cost-

BAF). In this phase, the forecasted EAC was continuously updated during execution, by feeding actual performance data into the model. In the second phase, Bayesian mode averaging technique was used to combine the cost-BAF model predictions with pre-project estimates.

In other research, Caron et al. (2013) used Bayesian statistics to improve the calculation of the EAC in EVM. The model was based on two main assumptions: independence of the indices CPI_m and $SPI(t)_m$ (the subscript m represents the monthly value) and a log-normal distribution. Their proposed framework was comprised of three phases, the first being "data analysis and logarithmic transformation of the indices' values" (Caron, Ruggeri, and Merli 2013). The second phase encompassed using expert opinion to construct the prior distribution, focusing on minimizing biases to avoid using incorrect parameters as inputs to the model. Finally, the posterior distributions of the indices, CPI_f and $SPI(t)_f$ (the subscript f represents the future value), were calculated. The model was tested on a project from the oil and gas industry to demonstrate its effectiveness and applicability. By combining expert judgement in a formal way with previous project data, more accurate forecasting of the EAC is achieved, and consequently, project controls can make better informed decisions.

2.2.4 Fuzzy Logic

Another technique used to improve the accuracy of performance evaluation and prediction models is fuzzy logic. Fuzzy logic and fuzzy expert systems have become more popular in the construction industry and are being used to model problems where available deterministic data are limited and measurement of factors affecting the model is subjective (Knight and Robinson Fayek 2002).

Robinson Fayek and Sun (2001) developed a fuzzy expert system to evaluate and predict performance on design projects. Factors that impact design project performance (e.g. overall size of design firm and level of competition in the market) and factors that measure design performance (e.g. level of accuracy of design documents and level of performance against cost of design) were established as input and output factors for the model, respectively. Membership functions were generated using a new approach based on objective data, and input factors were related using expert rules. Model validation took place with a survey, and the results showed that the model provided accurate design performance in the form of linguistic predictions – i.e. factors are subjective in nature and are thus described using linguistic terms (e.g., small, average, large). Limitations of the fuzzy expert system included a low success rate for numerical predictions, lack of sufficient data resulting in the failure of some membership functions, and difficulty in application due to complexity.

Knight and Robinson Fayek (2002) developed a fuzzy logic model, to be used during the bidding stage, that assesses risk on engineering design projects and forecasts potential cost overruns to determine the likelihood of making profit. The authors relied on fuzzy binary relations to model the relationships between 13 project characteristics and eight risk events to determine the cost overrun caused by each combination of factors. Model validation was performed using case studies and a sample project is presented in the paper. The output of the model is useful in for design firms to use in determining the appropriate fee based on likely project performance or modifying controllable and existing conditions to enhance the performance.

2.2.5 Artificial Intelligence

Research into artificial intelligence and its application in the construction industry has emerged in the past decade. Neural networks (NN), a research area that falls under artificial intelligence, attempt to mimic the human brain's capabilities of thinking, storing, processing, and retrieving information (Moselhi, Hegazy, and Fazio 1991). Some construction management applications that used NN include financial forecasting, optimum markup estimation, task sequencing optimization, prediction of cost change indices, project performance predictions (Boussabaine 1996).

Ling and Liu (2004) used artificial neural networks (ANN) to construct a design build (DB) project performance prediction model and tested it using data from five projects. The authors specified DB project success using 11 performance indices (e.g. construction and delivery speeds, turnover, and project intensity) and determined 65 factors that may affect the success of these projects. Models to predict specific project performance were developed using correlation analysis and ANN on data from 33 DB projects, and they were subsequently tested using data from five new projects. The results indicated that six performance factors could be predicted with a reasonable degree of accuracy using the developed framework. The study concluded that in order to ensure DB project success, contractor selection and project variables should be given more attention. The limitations of this research is that it has been applied to DB projects only and the dataset from which the model was built included 33 projects, which may be insufficient.

Ko and Cheng (2007) attempted to combine genetic algorithms (GA), fuzzy logic (FL), NN to develop an evolutionary project success prediction model (EPSPM) for continuous monitoring of a project. Their model used GA for optimization, FL for dealing with uncertainty in predictions, and NN for capturing complex relationships between success factors as input and project outcomes as output. The combination of NN and FL allows for an integrated system that accurately captures qualities of the human brain. The model consisted of four main components: a project database storing information about previous projects, factors that are useful in predicting project success, an adaptive engine, and a predictive solution component. The validated results showed that project managers can use the model in real-time to make proper decisions and accomplish projects successfully.

2.2.6 Curve Fitting and Regression Analysis

The financial performance of a project can be planned and controlled using curve fitting techniques. The advantage of these methods over traditional EVA is that they provide an estimate at completion and can predict the cash inflows and outflows throughout the entire project duration (Willems and Vanhoucke 2015). Regression-based techniques are popular due to their ability to describe linear and nonlinear statistical relationships between input and output variables. Narbaev and Marco (2014) attempted to overcome the limitations of traditional EVM in calculating the Cost Estimate at Completion (CEAC) using a modified index-based formula to predict the estimate to completion. The framework also indicates the expected duration at completion by integrating earned schedule concepts. The methodology is divided into three steps. The first step is calculation of the Gompertz growth model (GGM) parameters, that describe phenomena inherent to data with a growth pattern, using non-linear regression curve fitting. Secondly, a new formula for CEAC is introduced integrating the evaluated GGM parameters. Finally, the equation is further modified to account for the effect of schedule progress on cost performance. The framework is tested using a eight sample projects and provides more accurate and precise estimates when compared with traditional index-based formulas in early, middle and late stages of the project.

2.2.7 Stochastic Methods

S-curves are used in construction projects as plots to represent the cumulative budget and planned duration against project progress. The name is derived from the S-like shape of the curve that most projects follow, i.e. flatter performance at the beginning and end of the project and steeper performance otherwise. Barraza et. al (2004) relied on the concept of stochastic S-curves (SSC) for evaluating project performance status at completion. The method was based on the variable nature of duration and cost of individual activities in the project. Using a simulation approach, forecasted SSC were generated at intermediate points throughout the execution of the project. These were compared with planned SSC to determine the variation at completion in terms of cost and duration. The approach overcame the inherent limitations of deterministic methods and curve fitting techniques that fail to provide a confidence interval around the estimates (Willems and Vanhoucke 2015). To demonstrate its flexibility, the method was applied on an example project.

Lee (2005) introduced Stochastic Project Scheduling Simulations (SPSS) to be used mainly during the bidding phase of a project. The software predicted the probability of completing a project at a certain date using MC simulation by accounting for the variability in the activities' durations and the longest path in the network. The study claimed to overcome the limitations of program evaluation and review technique (PERT) while retaining the CPM modeling environment with stochastic activity durations. The limitations of the model included its inability to transfer data from commercially available software.

Acebes et. al (2015) proposed an integrated methodology using MC simulation and statistical learning methods for monitoring projects under uncertainty. First, information about the expected behavior of the project was explored, and simulations in combination with Anamoly Detection algorithms were used to detect whether performance deviations were due to expected variability. If this was the case, classification and regression techniques were used to estimate probabilities of success in terms of time, cost, expected total duration, and cost of the project. The study was a refinement of previous methodologies based on EVA and risk analysis.

2.2.8 Miscellaneous Methods

There is limited literature on combinations of EVA with methodologies such as Kalman filter, critical chains, GA, dynamic programming, and other optimization models (Willems and Vanhoucke 2015). The probabilistic forecasting model developed Kim and Reinschmidt (2010) for predicting project duration at completion Kalman filter and EVA will be reviewed in detail in Chapter 4.

2.3 Evaluating the Performance of Forecasting Methods

Despite extensive literature on forecasting models for the successful delivery of construction projects, research on methods to evaluate the performance of these models is limited. Teicholz (1994) proposed a computational approach, the sliding moving average, to forecast the final cost and budget variance of construction projects. The forecast of the final cost of the project reflects overruns and underruns recorded earlier in the project. The study relied on 121 real construction projects completed over 15 years to compare the proposed methodology to two other methods. The methods were compared based on a number of criteria including accuracy, timeliness, and consistency. Analysis of the results indicated that the proposed methodology was superior in the criteria studied. However, limitations include the use of projects all from the same construction company. Other researchers have also used actual data from completed construction projects to evaluate the performance of their developed models. Robinson Fayek and Sun (2001) proposed a fuzzy expert system for design project performance prediction and evaluation. They validated the model using seven case studies of design projects ranging from 5 to 18 months. Although the study concluded with reasonable results, the main limitation included the inadequacy of the dataset from which these conclusions were drawn. Furthermore, Abdel Azeem et. al (2014) attempted to validate the performance of two deterministic forecasting methods, EVA and ES, against a probabilistic method, the Kalman Filter Forecasting Model (KFFM). A case study of an actual project is used for the validation, and the lowest average percentage error is the criterion that the methods were ranked on. According to the results, the KFFM outperformed the other two models by providing probabilistic prediction bounds of the project duration at completion with lower errors than EVA and ES.

To overcome the limitations of a limited dataset for validation, Vanhoucke and Vandervoode (2007) relied on artificial projects and used simulation to evaluate different earned value (EV) metrics in forecasting project duration. The proposed methodology generated 3100 diverse project networks by varying the topological structure of each network. The projects were the subject to nine uncertainty scenarios to generate actual performance by simulating them using MC principals. The purpose of the study was to draw objective and extensive conclusions about these methods that could be generalized to a large set of real-life scenarios. Kim (2007) also used the idea of generating artificial datasets to test the performance of forecasting models against a wide range of possible solutions. The projects were generated using a simulation approach based on the schedule of activities. Four forecasting methods were tested – EVA, CPM, KFFM, and the Bayesian Adaptive Forecasting Model (BAFM) – after random executions of the project produced a large dataset of diverse projects. The methods were compared on accuracy, timeliness, and reliability, and a number of hypotheses were tested on the dataset of 6000 projects with planned total durations ranging from 64 to 91 weeks. The study concluded that comparisons among the different methods cannot be deterministic, i.e. no one method exclusively outperformed the others in all aspects. In another study, Khafri (2018) developed a Bayesian-Markov forecasting model that used historical performance data to improve the EAC. The model was validated using a randomly-generated set of 100 projects and tested for accuracy against EVA. For each month, the planned value (PV), EV, and actual cost (AC) were randomly produced, and a diversified set of projects were created by generating projects of different planned value types, namely linear, S-curve, front-loaded, and back-loaded. Different accuracy measures were used for comparison including the mean absolute percentage error (MAPE), mean absolute deviation (MAD), and root mean squared deviation (RMSD). The research claimed that through substantial testing, the proposed method generated more accurate forecasts than EVA, especially at earlier stages of the project.

2.3.1 Limitations of Proposed Methods

In the discussed frameworks, the developed models demonstrated more accurate and timely forecasts through testing on actual project data and comparing the results with other methods, most commonly EVA. However, using a number of case studies where one model outperforms others fails to prove the superiority of that model because in another situation that may not be the case (B. Kim 2007). Accordingly, the use of actual project data for validation purposes has been criticized as having several shortcomings. For the results from process validation experiments to be reliable and meaningful, the sample size has to be large enough to reflect the process accurately. The larger the sample size, the more precise the results obtained from the experiment are in terms of determining the mean and identifying outliers. However, actual project data is often limited in amount, and to be able to use historical data for comparison purposes, a complete record of all planned and actual performance data must be available. This is very difficult to achieve in the real world (M. Vanhoucke and Vandevoorde 2007). Furthermore, with forecasting being an integral part of any construction project, actual data is often affected by the forecasting method applied during the execution of the project. For the comparison to be reliable, the project data used should be independent of any forecasting methods being tested (B. Kim 2007).

Thus, artificial project data are more suited for comparison purposes as it allows for the generation of larger datasets from which conclusions can be more reliably drawn. However, these data can also provide misleading results if experiments are not carefully designed to ensure that the dataset is free from bias and diverse enough to accurately reflect reality. This highlights one of the main limitations of the previous research – planned and actual progress are generated from the same probability distributions. Consequently, the results of the experiment are biased, especially when the same distributions are used as inputs to the forecasting model, i.e. the generated projects and forecasts are not independent.

2.4 Discussion and Summary

The above review of forecasting methods is not exhaustive, but it shows the importance of forecasting for successful project management. Monitoring the progress of a project throughout execution by estimating future progress is crucial for keeping projects under control. A variety of methodologies have been used by researchers to achieve a range of objectives, from improving forecasting accuracy to calculating the performance necessary during the remainder of a project to be able to complete it successfully (Willems and Vanhoucke 2015). Although most studies address time and cost improvements and control systems, the validation methods applied using actual performance data indicate a substantial need for providing an empirical method for validation.

The recent growth in the number of techniques and models for forecasting has produced a need for a clearly defined, objective, and flexible method to evaluate the performance of these models. Accordingly, research in this thesis focuses on developing an empirical experiment for testing the performance of forecasting techniques. By using a MC simulation, thousands of fictitious planned projects can be created with randomly generated tasks, relationships, durations, and costs. To create a diverse dataset, the complexity and lengths of the projects are varied. Actual progress variations are randomly generated using a Markov chain method that will be discussed in the following chapter. Relying on Markov chains ensures that the executions are indeed random, unbiased, and a true reflection of reality. Several criteria are defined to compare the predictions by different forecasting methods and are used to test a number of hypothesis. An example is provided to show the implementation of the experiment.

CHAPTER 3: MC SIMULATION-BASED EXPERIMENT DESIGN

3.1 MC Simulation

MC simulation is a technique used to solve a variety of mathematical problems. The fundamental idea of MC is modeling and understanding a system using random sampling (Amelin 2013). The technique is used in a variety of disciplines including engineering, physical sciences, computing (i.e. machine learning), finance, statistics, among others. Random experiments, in which the outcome of the experiments are unknown, are at the core of MC simulation (Rubinstein and Kroese 2016). A typical mathematical model involves inputs, model formulation, and outputs as shown in Figure 1.

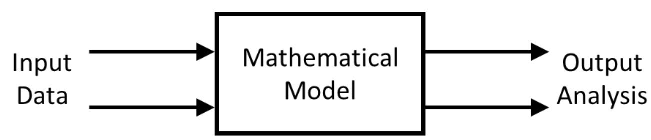


Figure 1 Mathematical models

The inputs are processed through the mathematical model to produce outputs. In most cases, the inputs to the model are affected by a number of external factors (Amelin 2013). In deterministic models, the most likely value of the inputs parameters are used, representing the base case scenario. However, using a single value for the input parameters ignores the effect of risks associated with the input parameters, resulting in an ineffective model. In other cases, the best case and worst case scenarios for the input parameter values are also used. Although this provides more realistic outputs, it still has several disadvantages. First, it may not be possible to determine the values of the input parameters corresponding to the best and worst case scenarios. Second, the same scenario may not occur for all the input parameters at the same time (Raychaudhuri 2008). This is where MC simulations are useful; they allow users to methodically investigate a variety of different scenarios involving risky input parameters. This is done by modeling the input parameters as statistical distributions.

A typical MC simulation process involves four main stages as discussed by Raychaudhuri (2008). First, a deterministic model resembling the real case scenario is developed. This model uses the most likely values of the input parameters and applies mathematical relationships to produce the desired outputs. Once the user is satisfied with the deterministic model, the risk component is added. This risk is associated with the stochastic nature of the input variables; thus, the underlying statistical distributions governing the inputs are identified. A common approach to determine the parameters of these distributions is using historical data combined with expert opinion. Using these statistical distributions, a set of random numbers, each corresponding to the value of one input parameter, is generated. The input values are fed into the mathematical model to provide a set of output variables. The process is repeated with a newly generated set of random numbers until the specified runs of the simulation are completed. This is the core of the MC simulation. Finally, the collected sample of output variables are statistically analyzed and used for decision-making purposes. The MC simulation process provides more confidence in the results.

This chapter relies on the fundamental concepts of MC simulation to design an empirical experiment for evaluating cost- and schedule-forecasting models. The output of the experiment is a set of randomly generated activity-on-node (AON) networks representing fictitious construction projects. The parameters of these projects, including the network topology (activities and relationships), activity budgets, durations, actual time, and cost expended, are determined using statistical distributions representing the input variables. Accordingly, a strongly random set of networks is generated representing the whole solution space of possible projects from the given inputs. The output of the experiment is used to answer of number of questions regarding the performance of different forecasting methods:

1. Which forecasting technique provides more accurate, stable, and timely forecasts?
2. Does the complexity of the project network affect the accuracy, timeliness, and stability of a forecasting method?
3. Does the length of the project affect the accuracy, timeliness, and stability of a forecasting method?
4. When do forecasts begin to stabilize and converge towards the final actual performance for different forecasting methods?

The experiment is set up in a manner that answers these questions by creating project networks with varying complexities and lengths, and testing several forecasting techniques including EVA, ES, KFFM, and non-linear regression Gompertz growth model (NLR-GGM). The following sections will describe the approach used to create project networks, the independent and dependent variables of the experiment that affect complexity and length of the networks, and automating the process to facilitate the testing and comparison of the various methods (Barton 2004).

3.1.1 Creating Random Project Networks

Typical construction projects begin with a plan that outlines the sequence in which work will be performed and is often presented in the form of an AON network. The AON network illustrates the relationships between activities or work packages involved in the project. It also includes the expected duration and allocated budget for every activity that has been derived from estimated quantities, productivity rates, and resource (labor and equipment) and material requirements. The CPM is used to calculate different properties for each activity including the early start, early finish, late start, late finish, total float, and free float. As early as 1993, researchers began to develop random activity network generators for comparing and rating the performance of different solution procedures. Demeulemeester et al. (1993) developed the first random network generator for creating test problem networks of varying sizes and structures. The number of activities and relationships were specified as input parameters, and the generator, referred to as strongly random, used these values to generate networks drawn from the full solution space of all possible networks. However, the generator only provided the network topology without any parameters. Agrawal et al. (1996) extended this work to overcome this limitation and created a random network generator, DAGEN, which added different parameters such as duration, as well as renewable and non-renewable resources, to the network. Kolisch et al. (1995) also developed a network generator, ProGen, that addressed the issue of resource-constrained scheduling problems. The methodology described the network construction procedure and defined two new resource measure parameters – the resource factor and resource strength. However, these procedures all fell short of developing strongly random networks by failing to assign equal probabilities to all feasible solutions (Browning and Yassine 2010). Accordingly, RanGen was developed by Demeulemeester et al. (1993) for generating AON networks that satisfy predetermined values of topological and resource-based parameters that control the hardness of the problem. These networks were intended to be used with several classes of scheduling problems. Vanhoucke et al. (2004) enhanced RanGen to become RanGen2 by modifying several topological network indicators previously developed by other researchers. Despite the success of these network generators, they were not aimed at specifically solving the resource-constrained multi-project scheduling problem (RCMPSP) and, consequently, do not include the important characteristics of these problems. Browning and Yassine (2010) attempted to overcome this by developing a random generator of RCMPSPs.

Most network generators discussed above focus on creating strongly random networks that form a largely variable sample set to objectively compare different solution procedures. However, specifying predetermined values for the inputs, such as the number of links and activities, limits the number of possible outcomes, thus affecting the degree of randomness of the network. In addition, including limitations to the topological structure of the network, for example, an activity in one tier (a subset of activities with no links between them) can only have successor activities in the immediately succeeding tier (Browning and Yassine 2010) to avoid creating redundant relationships, further affects the degree of randomness of the network. Accordingly, the method proposed in this chapter overcomes these limitations and creates strongly random networks that are more representative of actual construction project networks. It allows an activity to have successor activities in any succeeding tier but puts a higher weight on immediately succeeding tiers. Redundant links are then eliminated to ensure that the network complexity value accurately represents the topology. Finally, to ensure the validity of the project network topology and cumulative progress over time, face validation is performed. In this technique, experts knowledgeable about construction projects are asked whether the produced networks are reasonable (Sargent 2007) and resemble actual construction project networks. The following sections describe the proposed method in detail.

3.1.1.1 Activities

Project networks are created using a code developed on Microsoft.Net Framework (Microsoft Visual Studio, version 15.9.7). The number of activities for each project is determined by sampling from a distribution to create varying project networks. Next, the activities are organized into columns as shown in Figure 2.

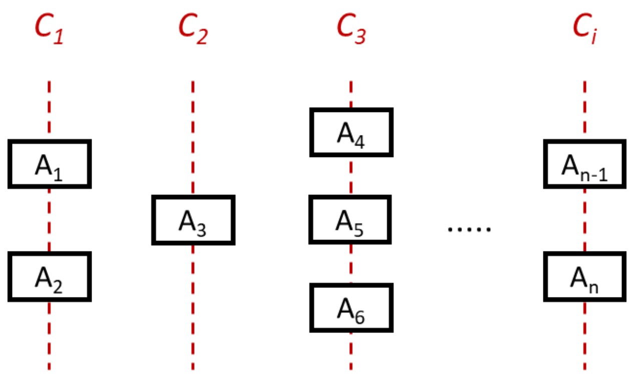


Figure 2 Activities and columns

3.1.1.2 Relationships

Relationships are then generated probabilistically for each activity using a weighted random selection technique (Kelly, 2016). The process involves assigning weights to each activity based on its position in the CPM network. Activities in the immediately succeeding column are given a higher weight than activities in later columns. For example, in Figure 3, relationships are generated for A_1, which is located in the first column. Accordingly R_1 has a higher weight since it connects to an activity in the second column, whereas R_2 has a lower weight since it connects to an activity in the third column. This can be illustrated as follows:

$$w_2 > w_3 > \cdots > w_i \tag{8}$$

The weighted random selection algorithm works as follows (Kelly, 2016):

1. Assign a weight for each activity based on the column in which it is located.
2. Add up all the weights for the activities in the list.
3. Sample a random number between 1 and the sum of weights.
4. For the first activity in the list, subtract its weight from the random number.
5. Compare the result obtained to zero.
6. If it is less than or equal to zero, select the activity and create a link. The maximum number of relationships for each activity is limited to three (i.e. activities can have 1, 2 or 3 relationships). If it is greater than zero, repeat steps 4–6 for the next activities on the list.

Based on the above algorithm, the random number decreases in each iteration ensuring that an activity will be always be selected. However, activities with higher weights will be *more likely* to be selected.

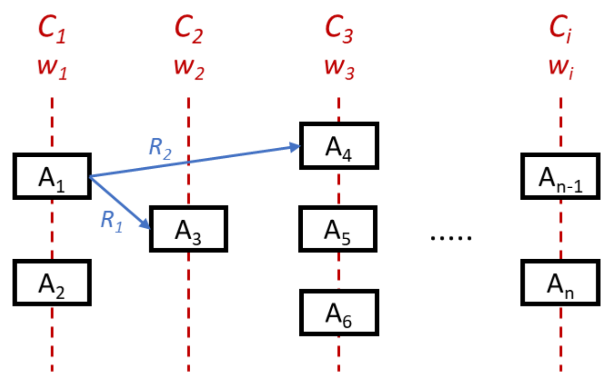


Figure 3 Generation of relationships between activities

Moreover, using the random weighted selection technique and organizing activities into columns ensures that there are no relationships between activities in the same column (parallel activities) or with activities in preceding columns. The list of possible relationships only includes activities in succeeding columns

3.1.1.3 Removing redundant relationships

The number of tasks and their relationships in a network often indicate schedule complexity (Bashir 2010). Although redundant relationships do not affect schedule logic, they can indicate a higher level of complexity than actually exits. As such they need to be removed before schedule complexity can be evaluated. It is expected that some of relationships created above may be redundant. Thus, the method developed by Bashir (2010) has been used to convert the AON project networks created into a minimum-edge diagraph networks without redundancies.

The method is based on the graph theory and one of its applications, interpretive structural modeling (ISM). AON networks can be described as acyclic diagraphs, where activities are represented by vertices, and the precedence relationships between two activities are described by arcs or links. Based on the number of activities, the maximum number of redundant relationships (MR) can be calculated using equation (9).

$$MR = \frac{n(n-1)}{2} - (n-1) \tag{9}$$

where n = number of activities in the project network.

Figure 4 shows a hypothetical project network to illustrate the method to remove redundancies. According to equation (9), the maximum number of redundant relationships in this network is 10. The first step in the process is to construct the matrix of immediate predecessors (MIP). This is a $n \times n$ matrix of binary entries (0 or 1), where 1 is used to represent the existence of a precedence relationship between activity i and activity j, while 0 is used otherwise, as shown in Table 1.

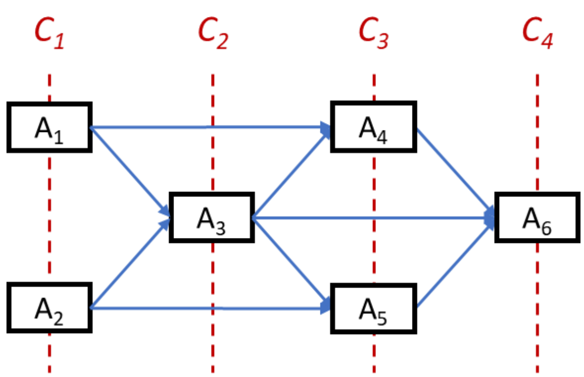


Figure 4 Hypothetical project network with redundant relationships

Table 1 MIP for the hypothetical project network

Activity *i*	Activity *j*					
	1	**2**	**3**	**4**	**5**	**6**
1	1	0	1	1	0	0
2	0	1	1	0	1	0
3	0	0	1	1	1	1
4	0	0	0	1	0	1
5	0	0	0	0	1	1
6	0	0	0	0	0	1

The next step involves transforming the MIP to a lower triangular format (LTF). This is accomplished by searching in the matrix sequentially for any row containing a single 1. These rows are entered sequentially into the lower-triangular matrix. In this next iteration, the process is repeated without including these rows or their corresponding columns. The process continues until the entire MIP is transformed to a LTF as shown in Table 2.

Table 2 Lower-triangular matrix for the hypothetical project network

Activity *i*	**Activity *j***					
	6	**5**	**4**	**3**	**1**	**2**
6	1	0	0	0	0	0
5	1	1	0	0	0	0
4	1	0	1	0	0	0
3	1	1	1	1	0	0
1	0	0	1	1	1	0
2	0	1	0	1	0	1

Finally, the minimum adjacency matrix is constructed. The first step involves replacing all the diagonal entry values of 1 with 0. Next, the rows are iterated over sequentially from Row 1 to Row n. For each e_{ij} of 1, column i is searched for a value of 1 in the rows greater than i. These values are replaced by 0 in the corresponding j column. The process is repeated for other entries until the minimum adjacency matrix is reached as shown in Table 3. Figure 5 shows the minimum edge diagraph of the hypothetical project with all the redundant relationships removed. The above process is automated for the created project networks to give a more realistic value of schedule complexity.

Table 3 Minimum adjacency matrix for the hypothetical project network

Activity *i*	Activity *j*					
	6	**5**	**4**	**3**	**1**	**2**
6	0	0	0	0	0	0
5	1	0	0	0	0	0
4	1	0	0	0	0	0
3	0	1	1	0	0	0
1	0	0	0	1	0	0
2	0	0	0	1	0	0

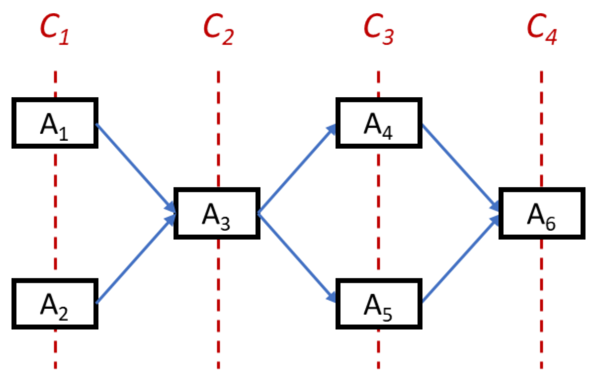


Figure 5 Minimum-edge diagraph for the hypothetical project

3.1.2 Actual Progress

During the execution of the project, actual performance on the construction site varies from planned performance due to several factors, including weather conditions, financial difficulties, and change orders. Accordingly, forecasting models are crucial for project managers to predict future performance based on actual progress, detect deviations, and make informed decisions to recover performance.

The variations in actual performance usually occur in expenditures and progress. Actual progress and expenditures are either greater or less than planned progress and expenditures. There can be several outcomes to each project depending on the conditions of the project, and these outcomes are somewhat random. To accurately reflect this randomness in the experimental design, a number of variations of actual progress will be created for each project as shown in Figure 6. In these variations, the project may end up finishing in one of the following states:

1. A cost and schedule overrun.
2. A cost and schedule underrun.
3. A cost overrun and schedule underrun.
4. A cost underrun and schedule overrun.

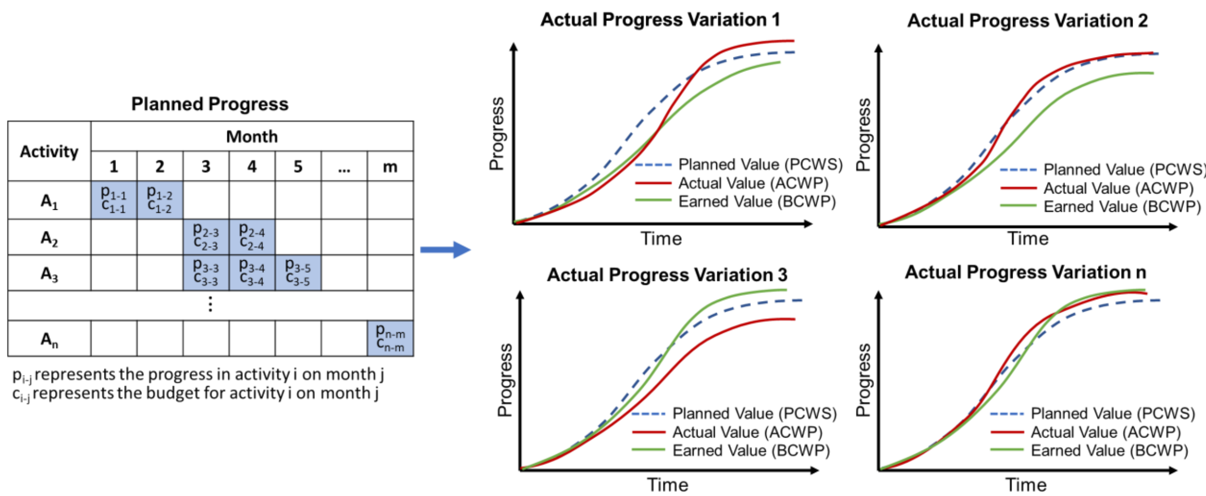


Figure 6 Actual progress variations from planned performance

To create these variations in performance, a Markov chain method was implemented. A Markov process is a special type of stochastic process in which the future state of a system only depends on the current state and not on preceding states. Accordingly, they are often referred to as *memoryless.* Directed graphs are used to represent Markov chains where the vertices of the graph correspond to the states and the edges correspond to the probability of transitioning from the current state i to the future state j (i.e. the transition probabilities). In this research, it was assumed that a project can occupy one of four states: outstanding performance (A), within target performance (B), below target performance (C), and poor performance (D). Each month, the project transitions from one state to another based on the transition probabilities. The illustration in Figure 7 shows the Markov chain used to transition from state i to state j (P_{ij}).

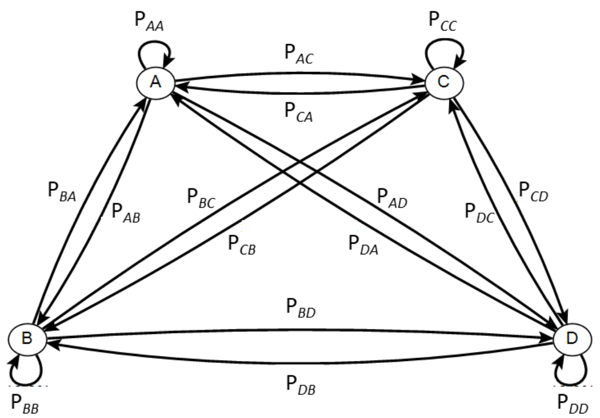


Figure 7 Markov chain directed graph for state transition

Assuming that there are four states the project performance can occupy, the 4×4 transition matrix is as follows:

$$[P] = \begin{bmatrix} P_{AA} & P_{AB} & P_{AC} & P_{AD} \\ P_{BA} & P_{BB} & P_{BC} & P_{BD} \\ P_{CA} & P_{CB} & P_{CC} & P_{CD} \\ P_{DA} & P_{DB} & P_{DC} & P_{DD} \end{bmatrix} \tag{10}$$

where P_{ij} represents the probability of transitioning from the current state *i* at time *t* to the future state *j* at time *t+1* for *i, j = A, B, C, D*.

Accordingly, there are several features of a transition matrix:

1. The matrix is square since all states can be possible current or future states.
2. The probabilities must be values between 0 and 1.
3. The sum of all the probabilities for one row must be equal to 1 since this represents all possible future states the system can occupy given a current state. Based on this, it can be concluded that:

$$\sum_{j=A}^{D} P_{ij} = 1 \text{ for } i = A, B, C, D \tag{11}$$

The initial state ($\pi^{(0)}$) of a Markov process shows the probability (P_n) of being in each state at the start of the process. It is represented by a vector with one row and N number of columns corresponding to the number of states (in this case 4 columns).

$$\pi^{(0)} = (P_A, P_B, P_C, P_D) \tag{12}$$

In order to calculate subsequent probabilities, the initial state vector is multiplied by the transition matrix.

$$\begin{gathered} \pi^{(1)} = \pi^{(0)} \times [P] \\ \pi^{(2)} = \pi^{(1)} \times [P] \\ \cdot \\ \cdot \\ \cdot \\ \pi^{(n)} = \pi^{(n-1)} \times [P] \end{gathered} \tag{13}$$

where $\pi^{(n)}$ = the probability vector at the last period and $[P]$ = the transition matrix.

Based on the above, a system can be modeled as a Markov process if it has the following features:

1. The outcome of the system is one of a set of finite states.
2. The outcome of each experiment with the system depends solely on the current state of the system and not on any past states.
3. The initial state probabilities can be determined.
4. The probability of transitioning from one state to another is constant, i.e. the transition probability matrix does not change.

The fourth feature, however, is not applicable for *Dynamic Markov chains* in which the transition probabilities vary from one period to another. This is the case with project performance in construction, where the probability of transitioning from one state to another does not remain constant over the project duration and is often a function of the progress (N. K. Nassar 2005). In other words, a project has a higher probability of changing from within target performance (state

B) to poor performance (state D) at the beginning of the project rather than towards the end of it. Nassar proposed using the cumulative planned progress S-curve so calculate the transition matrix probabilities at each period. However, this method assumes that the project duration is constant irrespective of the performance, which is rarely the case. To overcome this limitation, actual progress (AP) based on the EV will be used to calculate the transition probabilities from one period to the next. Given that the initial and final state vectors are known, if performance is poor and the project duration is extended, the cumulative actual progress in each period will be less than the cumulative planned progress; 100% progress (i.e. completion of the project) will be attained at a later period; and vice versa if performance is satisfactory.

$$AP = \sum_{n=1}^{N} EV_n \Big/ BAC \tag{14}$$

where *n* = the number of activities/work packages in the project, and BAC = the project budget at completion.

Accordingly, the transition probabilities at period k are calculated as follows.

$$P_{(ij)_k} = P_{(ij)_0} + \left(P_{(ij)_{final}} - P_{(ij)_0}\right) \times AP_k \tag{15}$$

Where $P_{(ij)_0}$ = the initial probability of going from state i to state j and $P_{(ij)_{final}}$ = the final probability of going from state i to state j.

By determining the transition probability for the current period, built-in rules are used to determine how the project performs given a particular state. Each month the Markov chain transitions to a new state; based on this state, the planned progress and cost for activities ongoing during that month are increased or decreased by a randomly sampled value. The remaining progress or the sampled progress, whichever less, is chosen as the actual progress for the activity on that month. The adjustment is determined as shown in Table 4.

Table 4 Adjustment to progress and cost based on Markov state

State	Interpretation	Effect on Progress and Cost
A	Outstanding Performance	Increase the progress by a random number sampled between a% and b% Decrease the cost by a random number sampled between a% and b%
B	Within Target Performance	Increase the progress by a random number sampled between c% and d% Decrease the cost by a random number sampled between c% and d%
C	Below Target Performance	Decrease the progress by a random number sampled between a_1% and b_1% Increase the cost by a random number sampled between a_1% and b_1%
D	Poor Performance	Decrease the progress by a random number sampled between c_1% and d_1% Increase the cost by a random number sampled between c_1% and d_1%

3.1.3 Variables

To answer the questions stated above, the values of several variables need to be altered. These include the duration and number of activities, the number of relationships, and number of MC simulation runs, i.e. the number of projects and actual progress variations of each project. Altering these variables will affect different aspects such as the length and complexity of the created projects. The duration and number of activities can be randomly set when each project is created and will ultimately affect the length of the project. However, the number of relationships and MC simulation runs chosen must be justified as they can directly affect the outcomes of the experiment. The following sections discuss these two variables in detail.

3.1.3.1 Number of MC Simulation Runs

The number of MC simulation runs is directly linked to the number of project networks created, as well as the number of variations of actual progress for each project. Although MC simulations have been used in several applications, one of their major disadvantages is the computational time they take (Lerche and Mudford 2005). The customary procedure used to determine the optimum number of MC runs is to run a fixed number of realizations and compute the mean and the standard deviation of the output of interest. Then the number of runs is increased, and the mean and standard deviation are calculated again. The process is repeated until a point is reached where increasing the number of runs affect mean value less than the level of accuracy required. Although this method can be time-consuming, it provides reliable results (Lerche and Mudford 2005). For the designed experiment, the standard error is used as a for determining the suitable number of runs. The standard error measures how precise the sample mean is versus the true mean of the population, thus provides a measure of accuracy. It is considered a part of descriptive statistics as it represents the standard deviation of the mean. Standard deviation, on the other hand, is a measure for volatility and is most commonly used in risk assessment. As the sample size increases, both the standard error and standard deviation decrease. To determine the optimum number of runs, the experiment is repeated until the increase in the number of runs has no significant effect on the standard error of the forecasts.

3.1.3.2 Number of Activities and Relationships

The number of activities in each project and the precedence relationships between these activities are directly linked to the complexity of the project. A project with many activities will generally have more dependencies/relationships and will be more complex than one with fewer activities. A dependency represents a relationship between two activities in which the initiation of one activity hinges on the completion or initiation of the other. Dependencies can be classified into logical, resource-based, or preferential. Logical dependencies are essential to the nature of the project and the activities involved. Resource-based dependencies are constrained to the limited number of resources that are available in a project. If an activity requires two skilled workers but they are currently unavailable, the activity cannot start until the both workers become available. Activities that have resource-based dependencies generally do not have logical dependencies; if more resources become available, the activities can be performed simultaneously rather than sequentially. Preferential dependencies are enforced based on convenience or best practice and often focus on improving the quality of deliverables. Roof foundations are often soaked for five to seven days before the tiles are laid. Although the tiles can be laid immediately, the preferential dependency is imposed to improve the structure’s integrity (Moitra, 2016).

Dependencies can also be characterized based on conditions such as the start and finish of tasks. In this case, there are four types of dependencies:

- Finish-to-Start (FS): where the start of the successor activity depends on the completion of the predecessor activity as shown in Figure 8. FS relationships are the most common in construction projects. Pouring concrete cannot begin until rebar has been placed and tied is a typical example of a FS relationships.

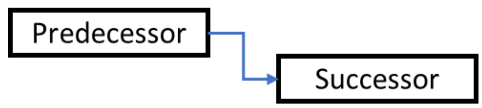


Figure 8 Finish-to-Start (FS) relationship

- Finish-to-Finish (FF): where the completion of the successor activity depends on the completion of the predecessor activity as shown in Figure 9. The start of the predecessor and successor activity are independent. FF relationships are also quite common in construction. An example can be placing forms (successor) and tying rebar (predecessor) where both activities do not necessarily start at the same time but generally placing forms should be completed after the rebar has all been tied.

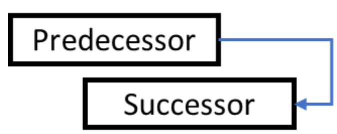


Figure 9 Finish-to-Finish (FF) relationship

- Start-to-Start (SS): where the successor activity cannot start until the predecessor activity has started as shown in Figure 10. SS dependencies also exist in several construction applications such as laying a pipe in a trench. Excavation of the trench begins until enough space has been cleared to begin laying the first section of the pipe.

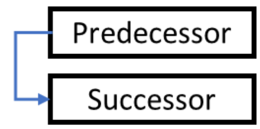


Figure 10 Start-to-Start (SS) relationship

- Start-to-Finish (SF): where the completion of the successor depends on the start of the predecessor as shown in Figure 11 . This type of dependency is uncommon in construction and is rarely relied upon in scheduling.

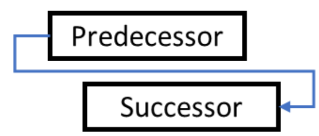


Figure 11 Start-to-Finish (SF) relationship

As mentioned above, FS, FF, and SS dependencies are more likely to exist in construction projects as compared to SF dependencies. However, project networks with several types of relationships are not considered more complex than those with only FS relationships, and accordingly do not affect the accuracy of forecasting techniques. The reason being that non-FS relationships with positive lags can be easily transformed into equivalent FS using generic transform schemes as shown in Figure 12 (Lu and Lam 2009).

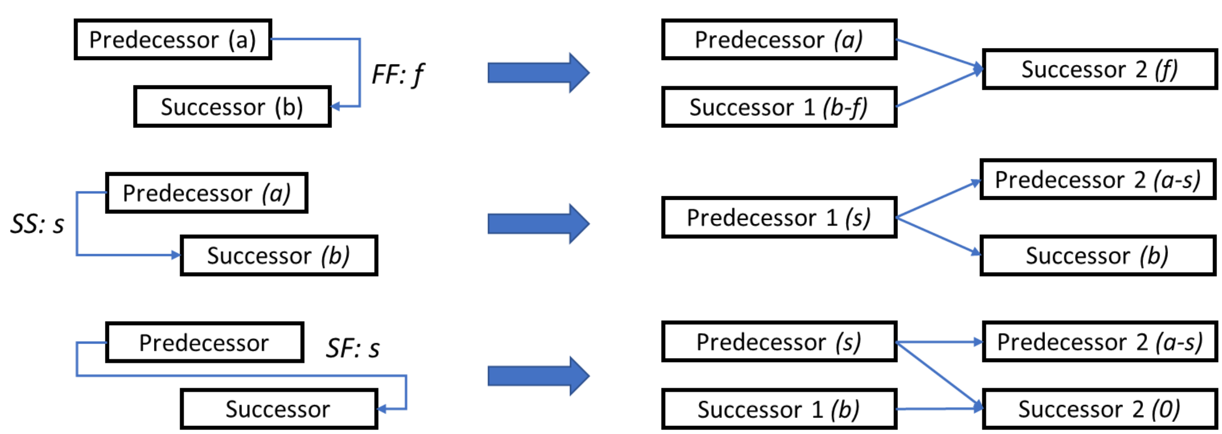


Figure 12 Transforming non-FS relationships to FS relationships (Lu and Lam 2009)

Consequently, only the number of tasks and FS relationships will be altered to create a dataset of varying complexity and test the effectiveness of forecasting models on this dataset.

3.1.4 Project Schedule Complexity

Complexity is a term that is difficult to define, and researchers have yet to reach a consensus on what project complexity actually is (Vidal, Marle, and Bocquet 2011a). Previously, the "characteristics of complexity have been used to define it, however not a single definition has been able to encompass all the situations where complexity exists" (Sinha, Kumar, and Thomson 2006). Complexity can have different contexts within the same field, as well as in different fields. For the purpose of this research, the definition proposed by Vidal et. al (2011b) will be used to describe the concept of project complexity: "Project complexity is the property of a project which makes it difficult to understand, foresee and keep under control its overall behaviour, even when given reasonably complete information about the project system."

Project performance can be directly associated with its complexity. Thus, measuring project complexity is crucial for efficient project management. Project complexity measures can be categorized into three groups (Vidal, Marle, and Bocquet 2011a):

1. Measures based on the *computational complexity* of different project management aspects.

    - Akileswaran et. al (1983) developed a computational complexity measure for a project sequencing problem to minimize cost by optimizing the expansion sequence of multi-purpose projects.

    - Reyck and Herroelen (1996) proposed a complexity index to differentiate between "easy and difficult occurrences of multiple resource-constrained problems and discrete time/cost trade-off problems" in terms of the computing effort required.

2. Measures based on *schedule complexity*.

- The simplest network complexity measure was developed by Pascoe (1966). It calculates complexity by evaluating the interconnectivity between activities.

$$CNC_{PERT} = \frac{number\ of\ activities}{number\ of\ nodes} \tag{16}$$

- The coefficient of network complexity (CNC) calculates the degree of interrelationship in critical path networks such as PERT and precedence networks. It can be useful in determining the network computer processing time and the amount of effort that should be allocated for planning the project (Kaimann 1974).

$$CNC_{PERT} = \frac{(number\ of\ activities)^2}{number\ of\ events} \tag{17}$$

$$CNC_{Precedence\ Networks} = \frac{(number\ of\ preceding\ work\ items)^2}{number\ of\ work\ items} \tag{18}$$

- Davies (1973) defined a network complexity measure coefficient for a multi-activity project resource allocation problem.

$$CC = \frac{2(number\ of\ activities - number\ of\ nodes + 1)}{(number\ of\ nodes - 1)(number\ of\ nodes - 2)} \tag{19}$$

- Arguing that other network complexity measures are less efficient as they fail to account for redundancies and do not provide intuitive results for project management to base their decisions upon, Nassar and Hegab (2006) came up with a network complexity measure (C_n) expressed as a percentage.

$$C_n = \begin{cases} 100 \times \left\{\dfrac{Log[a/(n-1)]}{Log\,[(n^2-1)/4(n-1)]}\right\}\% & if\ n\ is\ odd \\ 100 \times \left\{\dfrac{Log[a/(n-1)]}{Log\,[n^2/4(n-1)]}\right\}\% & if\ n\ is\ even \end{cases} \tag{20}$$

where a = number of arcs (arrows) in AON networks and n = number of nodes AON networks

3. Measures based on a universal view of the project.

    - Sinha et. al (2006) developed complexity index (CI) for measuring complexity at several stages of the project The index can be used by project managers as a planning tool to minimize the impact of any complexities that may materialize during the project lifecycle.

    - Research studies have identified multiple aspects of project complexity using the Delphi process and ranked them using the analytical hierarchy process (AHP) to develop a complexity measure based on a multi-criteria approach. (Vidal, Marle, & Bocquet, 2010; Vidal, Marle , & Boucquet, 2011; Xia & Chan, 2012).

Based on the discussion above, project complexity in the context of this research refers to the project schedule complexity. Other complexity measures that account for computational efficiency or are developed based on a holistic view can be used if the projects are further complicated, for example, by adding resource constraints. As equation (20) accounts for redundancies and provides complexity measures as a percentage, it was used in this research for comparison purposes between the efficiency of forecasting techniques with varying project complexity.

3.1.5 Forecasting Measures

Research works that propose new forecasting methods often attempt to enhance three aspects of the forecasts: the accuracy, stability, and timeliness. This section discusses the meaning of each of these measures in the context of this research and how they will be evaluated.

3.1.5.1 Accuracy

Accuracy is one of the most important factors to be considered in selecting a forecasting method, especially in construction projects where inaccurate forecasts have a large impact on the performance of a project. However, one of the difficulties with accuracy measures is the absence of a single universally accepted measure (Mahmoud, 1984). Despite this, there are number of commonly used accuracy measures, including the mean absolute error (MAE), the mean absolute percentage error (MAPE), the mean percentage error (MPE), the mean squared error (MSE), and the root mean square error (RMSE). Table 5 shows the equations for each measure.

Table 5 Accuracy measure equations

Error	**Equation[1]**
Mean Absolute Error (MAE)	$MAE = \frac{100\%}{n}\sum_{i=1}^{n}\lvert a_i - f_i\rvert$
Mean Absolute Percentage Error (MAPE)	$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left\lvert\frac{(a_i - f_i)}{a_i}\right\rvert$
Mean Percentage Error (MPE)	$MPE = \frac{100\%}{n}\sum_{i=1}^{n}\frac{(a_i - f_i)}{a_i}$
Mean Square Error (MSE)	$MSE = \frac{1}{n}\sum_{i=1}^{n}(a_i - f_i)^2$
Root Mean Square Error (MSE)	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(a_i - f_i)^2}$

[1] where *n* = number of projects, *a* = actual cost/duration of project *i*, and *f* = forecasted cost/duration of project *i* for the each forecasting period.

MAE measures the average of the absolute differences between the actual value and the forecasted values. Due to the nature of the calculation, the individual differences all have equal weights and taking the absolute value ensures that the positive and negative errors do not cancel out. MAPE is considered the weighted version of MAE as the differences are scaled against the actual value (Drakos, 2018) . Since MAPE is expressed as a percentage, it is often easier to interpret than MAE. However, there are several weaknesses to MAPE which stem from the division operation and make it less likely to be used. MAPE is undefined for actual values of zero and is biased towards negatively skewed predictions (prediction is less than actual) as compared to positively skewed predictions (prediction is higher than the actual) by the same amount (Pascual, 2018). MPE is similar to MAPE, but it incorporates both positive and negative values by not taking the absolute value. Accordingly, MPE is better suited for measuring model bias rather than accuracy since the positive and negative errors cancel out (MAE and RMSE — Which Metric is Better?, 2016). Alternatively, MSE measures the average of the squared differences between the actual value and the forecasts. By squaring the differences, larger errors have higher weights than smaller errors, and similar to MAE and MAPE, the negative and positive values do not cancel out (Drakos, 2018). RMSE is simply the square root of MSE. By introducing the square root in the RMSE, the error is scaled to be the same as the actual value. Both these measures have the benefit of penalizing larger errors. However, the effect of the square term becomes apparent when outliers are present in the dataset, as the error grows quadratically in MSE. This means that outliers will have a higher weight and ultimately their effect on the total error will be greater (Pascual, 2018).

Subsequently, in the case of forecasting total project duration and cost, the ***mean absolute error*** (MAE) and ***mean absolute percentage error*** (MAPE) are the most appropriate measures to use in comparing the accuracy of forecasting models. The former being unbiased and the latter being easy to interpret. MSE and RMSE are more useful when larger residual values are particularly undesirable, which is not the case here (MAE and RMSE — Which Metric is Better?, 2016). The reason behind this is, for example, having a difference of 10 days between the actual and the forecast is exactly twice as bad as having a difference of 5 days which is reflected by MAE. Whereas, RMSE and MSE measures assume that having a difference of 10 days is four times as bad as a difference of 5 days.

The accuracy of the cost and duration forecasts was calculated at forecasting period for each forecasting technique for comparison purposes. This was repeated for the number of projects created.

3.1.5.2 Timeliness

As the project progresses, forecasting accuracy is expected to converge towards the actual value at completion. Forecasting models that converge more quickly than others are more reliable because managers can detect deviations earlier, when they can be controlled and their impacts reduced. Figure 13 shows the impact of different variables based on the project time. As the project time increases, the ability to make changes that will influence the final outcomes of the project decreases, and the cost of such changes increases.

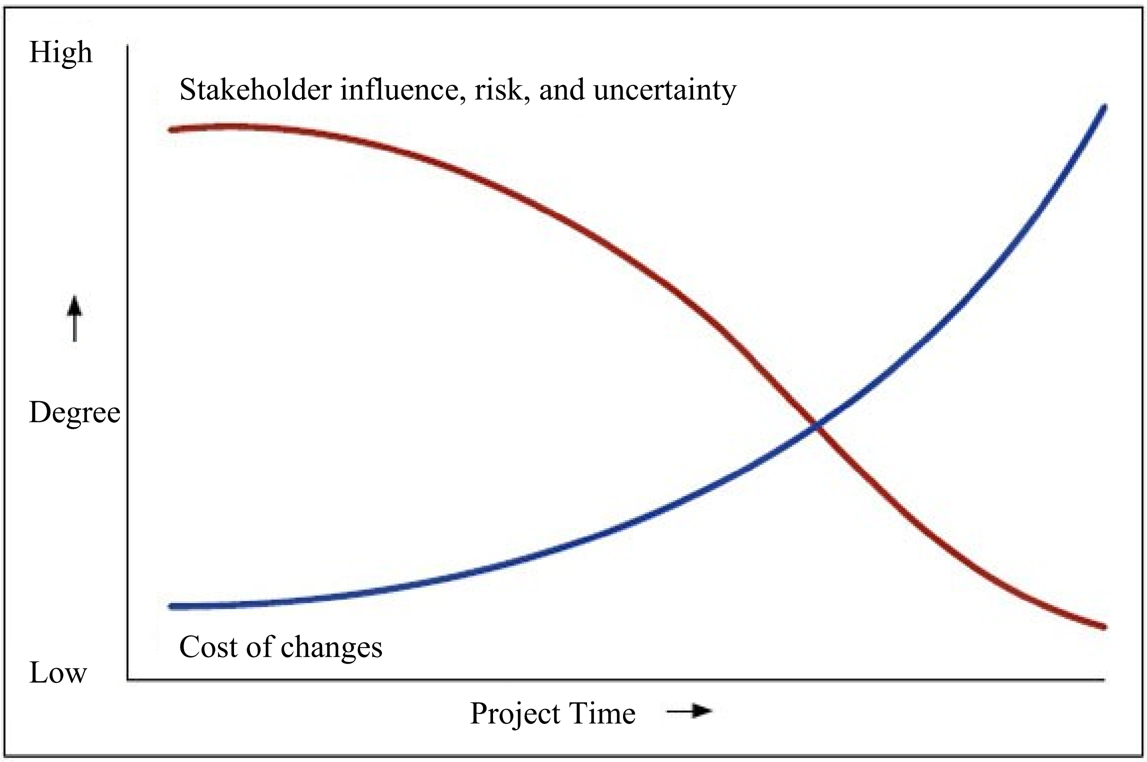


Figure 13 Impact of variables based on project time (Schoonwinkel, Fourie, & Conradie, 2019)

Accordingly, when comparing the accuracy of the models, one important feature to identify is when this accuracy was achieved. In other words, a model which has a MAE of 15% when the project is 30% complete is more favorable than a model that achieves the same MAE when the project is 70% complete. The timeliness (*T*) of a forecasting method is calculated as:

$$T_x = \frac{time\ of\ forecast}{actual\ project\ duration} \tag{21}$$

where x = desired level of accuracy (MAE)

3.1.5.3 Stability

The stability of a forecasting technique is dictated by how much predictions vary between one forecasting period and the next. If forecasting is performed on a monthly basis, a more stable technique would generally provide similar forecasts from month to month unless extreme changes happen to the project. If forecasts fluctuate greatly from month to month, projects managers would find it difficult to draw conclusions or make decisions based on these forecasts. Accordingly, the more stable a forecasting method is, the more reliable it is considered to be. The stability of a forecasting technique can be measured using the *coefficient of variation* (*CV*), also known as the relative standard deviation. It is a standardized measure of the dispersion of a dataset around the its mean. The lower the CV, the greater the forecasting stability.

$$CV = \frac{\sqrt{\sum_{i=1}^{n} \frac{\left(f_i - \bar{f}\right)^2}{n}}}{\bar{f}} \tag{22}$$

where f_i = forecast at each forecasting period, $\bar{f}$ = average of the forecasts, and n = the number of forecasting periods for each project.

It is important to differentiate between forecasting stability and forecasting accuracy. Forecasting stability is about comparing forecasts to forecasts, whereas forecasting accuracy is about comparing forecasts to the actual value. A forecasting method may have a low coefficient of variability indicating that the forecasting stability is high, but still have a low forecasting accuracy if $\bar{f}$ is far from the actual value. On the other hand, it is expected that a method which is less stable will likely have low accuracy since the mean of the forecasts will be further away from the actual value.

3.1.6 Process Automation

The entire experiment detailed above is automated using a code developed on Microsoft .Net Framework (Microsoft Visual Studio, version 15.9.7). Figure 14 shows a summary of the process. The process begins by creating a project network, including developing the activities for the network and organizing them in columns, linking them randomly with relationships, removing redundancies, assigning cost and duration distributions to each activity, and carrying out CPM calculations to get the total project duration (TPD) and total project cost (TPC) and calculate the complexity using equation (20).

The activity durations and cost are then translated into a monthly Gantt chart to determine the monthly cumulative progress and budget. As construction projects are executed, actual performance on the site differs from planned performance. Accordingly, several variations of actual progress are developed in this experiment from the planned project. These variations represent what actually happened on site in terms of monthly progress and expenditure for each activity, as well as actual project completion time and cost.

Actual progress from each month is periodically input into each forecasting method, and the methods are used to predict the final project duration and cost. Since the actual TPD and TPC are already known, they can be compared with the forecasts to calculate forecasting accuracy, timeliness, and stability. This is repeated for each actual progress variation and then the entire process is repeated until the specified number of projects are created. The results from all the forecasting techniques are compared based on varying project complexities and lengths to be able to draw reliable conclusions.

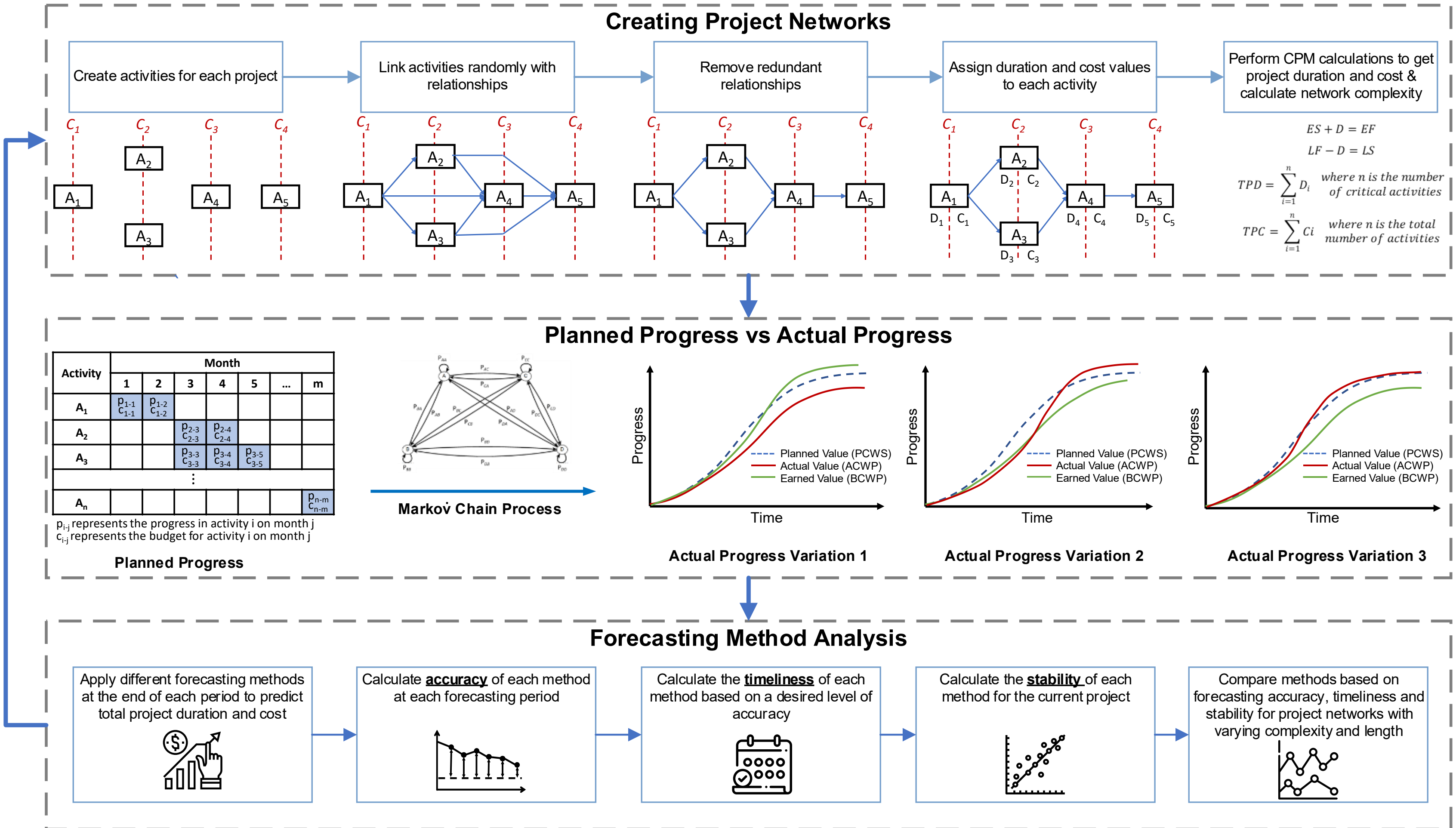


Figure 14 Process automation

CHAPTER 4: EXPERIMENT TESTING

4.1 Introduction

As discussed in the previous chapter, an MC simulation experiment was designed to test the performance of different forecasting theories. Because most projects in construction are unique, the use of actual completed project data for such comparisons is insufficient because these data are limited, incomplete, and biased towards the forecasting method used during the project execution. As such, an empirical experiment was developed to overcome these limitations. The experiment relies on MC simulation to create random projects, thus it methodically investigates a variety of different scenarios. The experiment entails creating these random projects based on statistical distributions for a number of activities, the relationships between these activities, the durations, and the costs. Predefined rules ensure that the projects are random yet mimic real life construction projects. Next, a Markov Chain method is applied to create random actual progress variation (i.e. schedule/cost overrun/underrun scenarios). The actual data is fed periodically into the forecasting technique, resembling what occurs in typical construction projects. The periodic forecasts from different methods are compared using specific measures of accuracy, timeliness, and stability (outlined in section 3.1.5) to provide a more reliable idea of their performance.

This chapter focuses on the implementation of the experiment using four forecasting techniques, EVA, NLR-GGM, ES, and KFFM. EVA is well-known method for estimating the cost at completion for a project given the planned and actual progress. The GGM has a shape resembling the S-curve of a project and uses non-linear regression to fit actual expenditure and project future expenditure. ES uses concepts similar to EVA to estimate the duration at completion of a project in time units. Similarly, the KFFM also predicts the estimated duration at completion by applying the Kalman filter algorithmic approach. These methods will be described in further detail below. The remainder of the chapter shows the results of running the MC experiment with these four methods. First, a sample of the random projects is shown to provide an illustration of the project's cumulative progress, expenditure, and AON network. Next, a number of hypotheses are tested regarding the performance of the tested methods under different conditions by varying the complexity and the length of the projects. The accuracy, timeliness, and stability results are displayed as graphs.

4.2 Forecasting Methods

Several forecasting methods have been developed over the years to predict duration, cost, quality, etc. (discussed in Chapter 2). This section details the four methods – EVA, NLR-GGM, ES, and KFFM – chosen to test the MC experiment.

4.2.1 Earned Value Analysis (EVA)

EVA is a method used in project management for estimating, budgeting, and control. It approximates the amount of work required to complete project activities and measures project performance against the scope, schedule, and baseline to ultimately forecast TPC and TPD. The method relies on three key measurements, expressed in cost units: the planned value (PV), EV, and AC. Figure 15 shows the different EVA measures that are explained below.

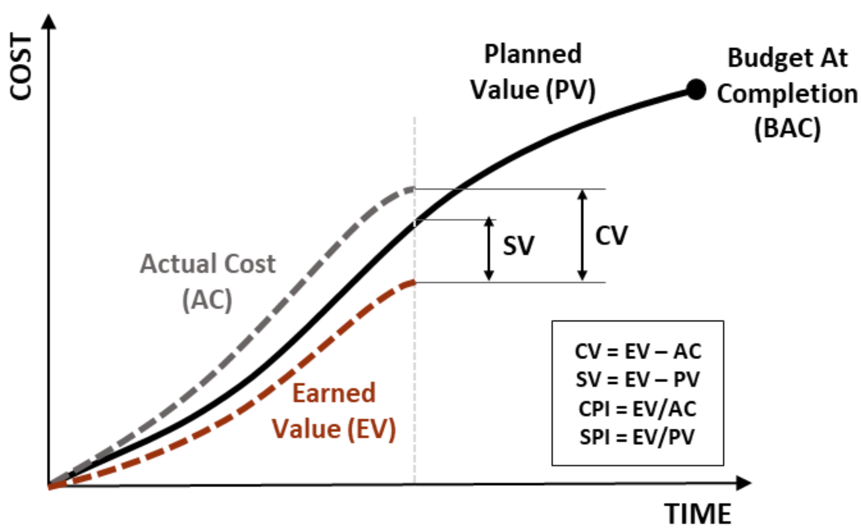


Figure 15 EVA calculations

PV, also referred to as the budgeted cost of work scheduled (BCWS), is the initial forecast of how much of the budget would be spent to finish the work as originally planned. It comprises the baseline schedule and approved budget to complete the work scheduled. During the execution of the project, the actual productivity is compared against the PV. As illustrated in Figure 15, the BAC represents the planned value at the conclusion of the project.

PV at any time (t) during the execution of the project can be calculated as:

$$PV_t = \sum_{i=1}^{n} (Planned\ \%\ Complete)_i \times Budget_i \tag{23}$$

where n = the number of activities in the project.

EV, also referred to as the budgeted cost of work performed (BCWP), calculates the budget that was originally planned for the amount of work actually finished. It expresses, in cost units, the amount of work completed up to a certain point in time (F. Anbari 2012). To calculate EV at a specific time (t) in the project, the following equation can be used.

$$EV_t = \sum_{i=1}^{n} (Actual\ \%\ Complete)_i \times Budget_i \tag{24}$$

where n = the number of activities in the project.

AC, on the other hand, expresses the cumulative amount of money actually spent for the amount of work completed up to a certain point in the project. It is not a measure of the amount of work done as per the plan. In other words, in case of lower productivity or higher unit costs, there may be unfinished work on certain activities even though the budget for those activities has already been spent. AC, in cost units, is calculated as follows.

$$AC_t = \sum_{i=1}^{n} Actual\ Cost_i \tag{25}$$

where n = the number of activities in the project.

Using these measures, the CPI and SPI can be calculated. The former measures how well the project is doing in terms of cost, i.e. whether there has been over-spending or under-spending. A value greater than 1 indicates that the project is under budget. The amount by which the project is over or under budget is expressed by the cost variance (CV). SPI measures the performance of the project in terms of duration, i.e. whether the project is ahead or behind schedule. Similar to CPI, a value greater than 1 indicates that the project is performing well and is currently ahead of schedule. The schedule variance (SV) calculates amount by which the project is behind or ahead of schedule, in terms of cost units.

$$CPI = EV/AC \ > 1 \ indicates \ that \ the \ project \ is \ under \ budget \tag{26}$$

$$CV = EV - AC \ > 0 \ indicates \ there \ are \ cost \ savings \tag{27}$$

$$SPI = EV/PV \ > 1 \ indicates \ that \ the \ project \ is \ ahead \ of \ schedule \tag{28}$$

$$SV = EV - PV \ > 0 \ indicates \ there \ are \ time \ savings \tag{29}$$

Decisions affecting the future are a primary concern to project management making forecasting an extremely important aspect of it. EVA is a useful technique that has been adopted in construction practices for cost and time forecasting. Using the measures of the current project performance, the cost estimate at completion can be predicted based on two major assumptions (Anbari, The Earned Schedule, 2012):

1. The remaining work will be performed based on the original plan, and deviations will not materialize into the future.

$$EAC = BAC - CV \tag{30}$$

2. The remaining work will be performed while continuing to over-spend or under-spend at the same rate.

$$EAC = BAC/CPI \tag{31}$$

4.2.2 Earned Schedule (ES)

ES is another method that has emerged from EVA as a means to forecast the time to completion or the project end date. EVA measures schedule performance in units of cost rather than units of time, which makes it difficult to interpret. Another disadvantage of EVA in terms of schedule forecasting is that a project that is completed behind schedule has a schedule variance of 0 and a schedule performance index of 1 at project completion (Lipke 2012). This further complicates the interpretation and understanding of the forecasts, limiting the use of EVA to cost performance analysis.

The concept of ES is analogous to that of EVA; the only difference is that schedule performance is measured in units of time. The fundamental theory behind ES is to determine the point on the planned value curve that corresponds to the time at which PV equals EV (Lipke 2012). This concept is illustrated in Figure 16. By locating this point, time-based indicators are formulated to provide the time variance and schedule performance efficiency (Lipke and Henderson 2006).

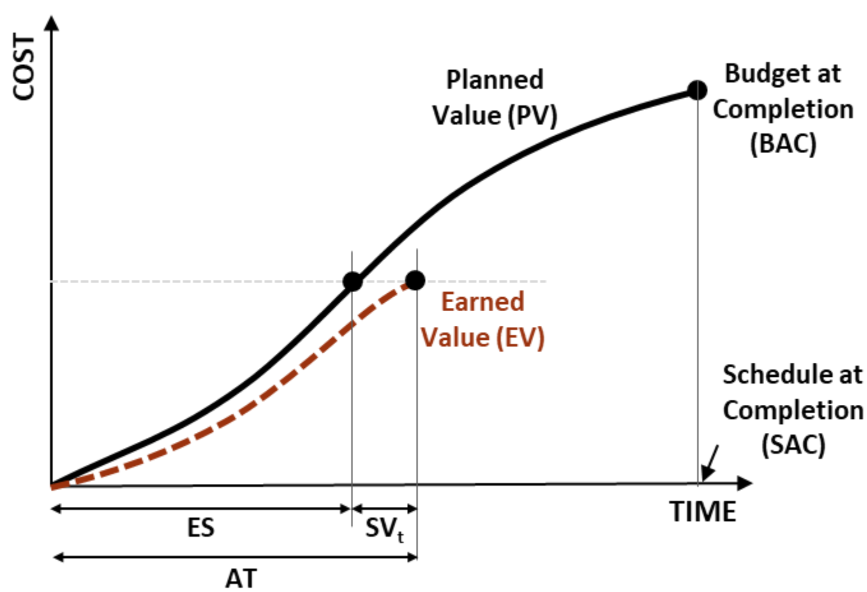


Figure 16 Earned schedule

Referring to Figure 16, the schedule at completions (SAC) is the time at which the project is planned to be finished. Actual time (AT) denotes the time at which EV accrued is recorded or, in other words, the duration between the start of the project and the current date. On the other hand, ES is the time at which EV accrued is earned (Lipke and Henderson 2006). Figure 17 shows how to evaluate ES at a given time during the project.

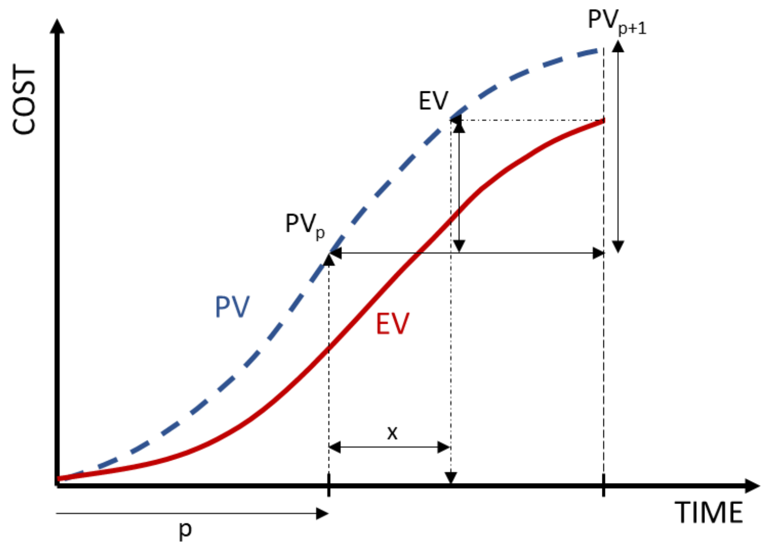


Figure 17 Earned schedule calculation

Given the planned value for the current period (PV_{p+1}), the EV and the PV of the previous period (PV_p), x, can be estimated using equation (32). The elapsed time, p, is added to the value of x to calculate the earned schedule at the current time as shown in equation (33).

$$x = \frac{EV - PV_p}{PV_{p+1} - PV_p} \tag{32}$$

$$ES = p + x \tag{33}$$

Using the two measures of AT and ES, the time-based indicators, time variance (TV) and schedule performance index time (SPI_t), can be evaluated as follows:

$$SPI_t = ES/AT\ > 1\ indicates\ that\ the\ project\ is\ ahead\ of\ schedule \tag{34}$$

$$SV_t = ES - AT\ > 0\ indicates\ there\ are\ time\ savings \tag{35}$$

Furthermore, the time estimate at completion (TEAC) can be forecast based on two assumptions similar to EVA (F. Anbari 2012).

1. The remaining work will be performed based on the original plan, and deviations will not materialize into the future.

$$TEAC = SAC - SV_t \tag{36}$$

2. The remaining work will be performed while continuing to carry out work at the same rate.

$$TEAC = SAC/SPI_t \tag{37}$$

4.2.3 Non-linear Regression Gompertz Growth Model (NLR-GGM)

A regression-based approach using S-curve fitting was proposed by Narbaev and De Marco (2014) to overcome limitations of traditional EVA index-based techniques, such as their unreliability at early stages or that past data is the best representation of future information. Construction progress is often presented as an S-shape sigmoidal curve representing slower work progress (lower rate) in the beginning and end of the project, and faster work progress (higher rate) in the middle. The Gompertz Growth Model (GGM) has been used in several applications in many fields as it describes "phenomena inherent to data with a growth pattern" (Narbaev and De Marco 2014). The equation for GGM is as follows.

$$GGM(x) = \alpha e^{-e^{(\beta - \gamma x)}} \tag{38}$$

The growth rate pattern of the GGM is similar to the work progress rate in construction projects, where the rate monotonously increases until it reaches a point where it begins to steadily decrease to zero.

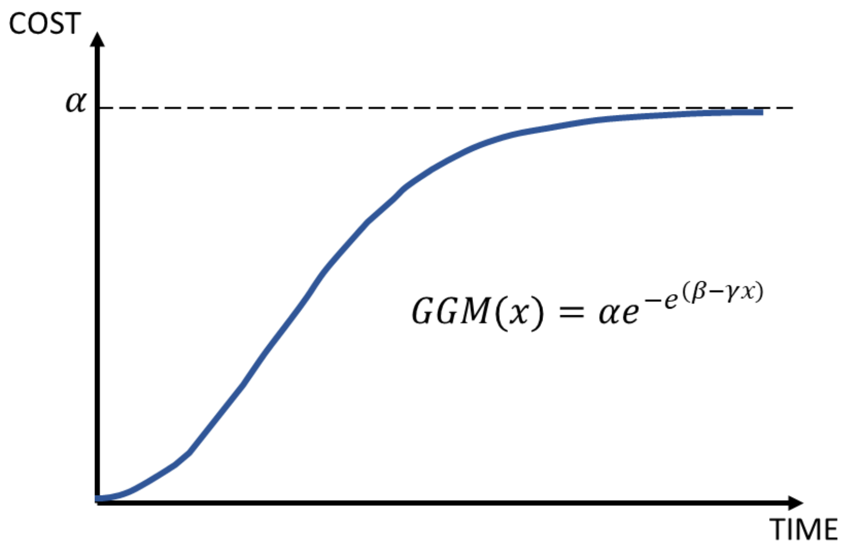


Figure 18 Gompertz Growth Model

Accordingly, Narbaev and De Marco (2014) proposed a modified index-based methodology for estimating the cost estimate at completion (CEAC). This methodology forecasts the cost for remaining work with a GGM via non-linear regression curve fitting. The advantage of the model is that it integrates schedule progress by accounting for the expected duration at completion using an Earned Schedule-based factor. A more detailed explanation on the parameters of the GGM can be found in (Narbaev and De Marco 2014). The methodology follows the following three steps:

1. Apply non-linear regression curve fitting to find the three parameters (α, β, γ) of the GGM.
    - The time points (predictor variable) are normalized by assuming that the project is 100% complete at the planned duration (i.e. PD = 1.00).
    - The cumulative cost points (response variable) are normalized; actual cost (AC) from time zero to actual time (AT) and PV from AT to project completion are normalized to unity by assuming that the BAC = 1.00.
2. Calculate the CEAC using a modified equation that includes the GGM.

$$GEAC(x) = AC(x) + BAC[GGM(1.00) - GGM(x)]$$

where x is the actual time.

3. The CEAC estimate is further modified by adding a factor that accounts for the schedule progress.

$$GEAC(x) = AC(x) + BAC[GGM(CF(x)) - GGM(x)]$$

where x is the actual time and CF is the Completion Factor calculated as the inverse of the schedule variance time (SPI_t^{-1}).

4.2.4 Kalman Filter Forecasting Method (KFFM)

Kalman filter is a recursive algorithm that attempts to predict the state of a dynamic system in the presence of noisy observations. Within this framework, the state of a dynamic system can be described using two variables, the state variable and error covariance variable, where the latter represents the error in the estimate of the state variable. In each iteration, the state variable and error covariance variable are predicted based on the system model and the previous state. The theoretical prediction is then updated using observed information to determine the current state. This process is outlined in Figure 19.

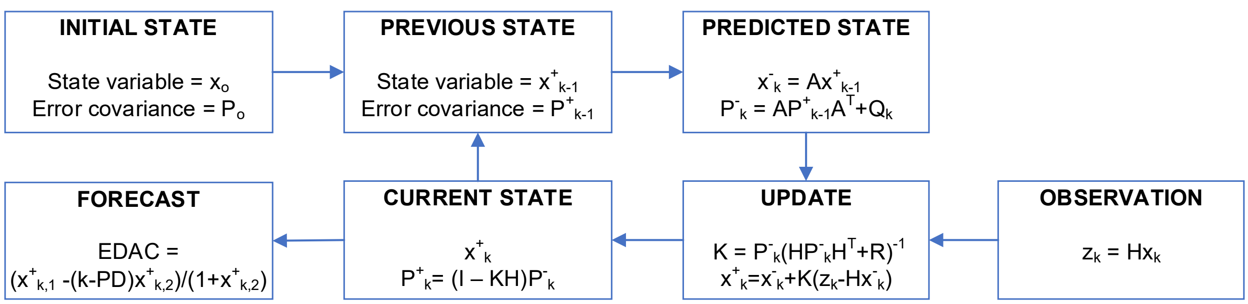


Figure 19 Kalman filter process

The initial state and error covariance become the previous state in the first iteration. Using the previous states, the prior estimates for the state (x_k^-) and error covariance (P_k^-) are predicted. The prediction relies on the transition matrix (A_k) and the process noise covariance matrix (Q_k), which represents the uncertainty in the model. Next, the Kalman gain is evaluated to determine how much the posterior estimates for the state (x_k^+) and error covariance (P_k^+) matrices are impacted by the actual observations compared to the theoretical prediction.

Kim (2007) relied on the concept of the Kalman filter to come up with a Kalman Filter Forecasting Model (KFFM) to predict project duration at completion. The method uses previous performance data available in the form of planned progress curves, actual performance data from the construction site, and the uncertainties and errors in measurements and execution to obtain an optimal estimate for the state of the project at completion that minimizes the error covariance. In the KFFM, the progress on a project is modeled as a dynamic system with two variables, the time variance (TV) and its rate of change over time. The former is defined as the deviation between the actual time and the planned time at which the work actually completed should have been reported. This is consistent with the earned schedule technique explained above. Accordingly the state matrix becomes

$$x_k = \begin{bmatrix} TV \\ \Delta TV \end{bmatrix} \quad for\ k = 0, 1, 2, 3 \ldots. \tag{39}$$

Where ΔTV is the difference between the time variation at *k* and *k–1*.

The KFFM algorithm description is too extensive to be covered in detail here so only a summary is provided in the following section. More detailed information can be found in (B. Kim 2007; B. C. Kim and Reinschmidt 2010). The KFFM is applied in the following way:

a) *Initialization of the Kalman filter parameters:*

1. Set the initial state (x_0) and error covariance matrices (P_o) to 0 at the beginning of the project.

$$x_o = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad P_o = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \tag{40}$$

2. Calculate the mean (μ_T) and the variance (σ_T^2) of the planned duration (PD) using PERT range estimates (assuming optimistic, pessimistic, and most likely rates as $O = 0.8 \times PD$, $P = 1.4 \times PD$, and $ML = PD$, respectively).

$$\mu_T = 1.03 \times PD \tag{41}$$

$$\sigma_T^2 = 0.1 \times PD \tag{42}$$

3. Based on the assumption that uncertainty in the system model is consistent with uncertainty in the estimate of project duration, the process noise covariance matrix (Q_k) is calculated. Given the mean and variance of a project duration, the error covariance matrix at $k = \mu_T$ is

$$P_{k=\mu_T}^{+} = \begin{bmatrix} \sigma_T^2 & 0 \\ 0 & \sigma_T^2 \end{bmatrix} \tag{43}$$

Using the assumption stated above, the Kalman filter prediction and updating processes are determined as

$$Prediction \rightarrow P_k^- = AP_{k-1}^+ A^T + Q_{k-1} \tag{44}$$

$$Updating \rightarrow P_k^+ = P_k^- \tag{45}$$

By substituting the initial covariance matrix and the covariance matrix at $k = \mu_T$, the only unknown variable becomes Q_k

$$\begin{bmatrix} \sigma_T^2 & 0 \\ 0 & \sigma_T^2 \end{bmatrix} = \begin{bmatrix} 1 & \mu_T \\ 0 & 1 \end{bmatrix} . \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} . \begin{bmatrix} 1 & 0 \\ \mu_T & 1 \end{bmatrix} + \begin{bmatrix} q & 0 \\ 0 & q \end{bmatrix} \tag{46}$$

4. The measurement error matrix (R_k) is determined based on expert opinion. It represents the sensitivity of the updating process to actual observations of progress collected from the construction site. As this measurement approaches zero, the Kalman gain increases causing the state and error covariance to be more influenced by the actual observations rather than the theoretical predictions.

b) Recursive Kalman filter algorithm at each reporting period (k).

1. Based on the actual data observed on the site, calculate the value of z_k.

$$z_k = Hx_k = [1 \quad 0] . \begin{bmatrix} TV_k \\ \Delta TV \end{bmatrix} \tag{47}$$

2. Prediction: calculate the predicted state x_k^- and predicted covariance matrix P_k^-.

$$x_k^- = A_k x_{k-1}^+ = \begin{bmatrix} 1 & \Delta T \\ 0 & 1 \end{bmatrix} . x_{k-1}^+ \tag{48}$$

$$P_k^- = A_k P_{k-1}^+ A_k^T + Q_{k-1} \tag{49}$$

3. Update: calculate the Kalman gain for the current reporting period.

$$K = P_{k-1}^+ H^T (H P_{k-1}^+ H^T + R)^{-1} = P_{k-1}^+ H^T \left(\begin{bmatrix} 1 & 0 \end{bmatrix} P_{k-1}^+ \begin{bmatrix} 1 \\ 0 \end{bmatrix} + [r] \right)^{-1} \tag{50}$$

$$x_k^+ = x_k^- + K(z_t - H x_k^-) = x_k^- + K(z_k - \begin{bmatrix} 1 & 0 \end{bmatrix} x_k^-) \tag{51}$$

$$P_k^+ = (I - KH) P_k^- \tag{52}$$

4. To predict the state estimates at time $t > k$, the Kalman gain is set to zero and the state and covariance matrices become

$$K = 0$$

$$x_k^+ = x_k^- = A_k x_{k-1}^+ \tag{53}$$

$$P_k^+ = P_k^- = A_k P_{k-1}^+ A_k^T + Q_{k-1} \tag{54}$$

5. Calculate the EV_k

$$EV_k = PV(ES_k) \tag{55}$$

$$ES_k = t + TV_k \tag{56}$$

   If the $EV_k > BAC$ then $TEAC = t$, else repeat steps b.4 and b.5.

6. Repeat steps b.1-b.5 for the next reporting period.

4.3 Randomly Generated Projects

As discussed in Chapter 3, the first step in the MC simulation-based experiment is generating planned project baseline schedules. The total number work packages involved in each project are sampled. These work packages are organized into columns/tiers, and using a weighted average method, the links between these activities are generated. The weighted average technique ensures that a realistic schedule is developed by preventing work packages at the beginning of the project from being linked to work packages at the very end of the project. This ensures that the project AON network is both realistic (by closely resembling actual construction projects) and random. After adding the relationships to define the predecessors and successors of every work package, redundant relationships are identified and eliminated to avoid a false sense of complexity to the project network. This is crucial since the effect of project complexity on the forecasting method performance is evaluated. The final step is adding durations and budgets to the individual work packages. The duration of each work package is first sampled from a predefined distribution, then budget/period is sampled. This ensures that work packages with a relatively short duration have lower budgets compared to work packages with relatively long durations. CPM calculations are then applied to the finalized work packages to come up with total final cost and durations of each project, representing the planned completion budget and schedule. Table 6 shows a summary of the different projects' network properties. A sample of the AON networks with the cumulative planned value S-curve can be found in Appendix A.

Table 6 Projects' network properties

Number of Work Packages	Number of Links	Project Complexity (%)	Project Duration (periods)	Project Cost ($)
6–30	6–48	2.86–83.65	16–60	~ 40,000–1,200,000

After creating the baseline project schedules, actual progress variations were generated for each project to represent a number of possible outcome scenarios. This was done using a Markov process with four defined states that a project can occupy in each period (A – outstanding performance, B – within target performance, C – below target performance, and D – poor performance). Based on the current state, progress and expenditure adjustments are made to any activities ongoing during that period. Accordingly, the projects ends up finishing in one of the following states:

1. actual total project expenditure and duration are greater than the planned total budget and duration, resulting in a cost and schedule overrun.

2. actual total project expenditure and duration are less than the planned total budget and duration, resulting in a cost and schedule underrun.

3. actual total project expenditure is greater than the planned total budget and actual total project duration is less than the planned total duration, resulting in a cost overrun and schedule underrun.

4. actual total project expenditure is less than the planned total budget and actual total project duration is greater than the planned total duration, resulting in a cost underrun and schedule overrun.

Table 7 shows a sample of the summary output from different Markov runs for each project. Two runs are shown for each project with the different combinations of cost and schedule overruns/underruns. The values by which the actual duration, cost overrun/underrun, planned duration, and cost differ in each run. This indicates that the Markov process is successful in simulating the randomness and uncertainty that takes place during the execution of project and produces a largely random sample dataset that can be used for validating different methodologies in a reliable and effective way.

Table 7 Markov runs actual project summary data

Project No	Markov Run No	Tasks	Links	Complexity Index (%)	Planned Duration (periods)	Planned Cost ($)	Actual Duration (periods)	Actual Cost ($)	Project Duration Overrun/ Underrun	Project Cost Overrun/ Underrun
1	1	18	23	19.362	54	314969	65	378660.8	Overrun	Overrun
1	2	18	23	19.362	54	314969	53	279532.3	Underrun	Underrun
2	1	17	20	14.836	80	354610	88	388961.5	Overrun	Overrun
2	2	17	20	14.836	80	354610	84	353193.6	Overrun	Underrun
3	1	13	14	12.305	58	284719	54	220402.2	Underrun	Underrun
3	2	13	14	12.305	58	284719	58	246838.1	Underrun	Underrun
4	1	14	18	24.526	47	181863	51	205568.6	Overrun	Overrun
4	2	14	18	24.526	47	181863	52	191766.4	Overrun	Overrun
5	1	10	13	35.993	27	103039	26	92566.33	Underrun	Underrun
5	2	10	13	35.993	27	103039	26	109763.1	Underrun	Overrun
6	1	12	15	26.16	33	131442	41	171410.8	Overrun	Overrun
6	2	12	15	26.16	33	131442	34	128571.1	Overrun	Underrun
7	1	12	15	26.16	33	131442	33	127762.4	Underrun	Underrun
7	2	20	25	16.525	65	294664	69	288040.9	Overrun	Underrun
8	1	21	30	23.784	63	286497	65	270138.8	Overrun	Underrun
8	2	21	30	23.784	63	286497	67	281822.2	Overrun	Underrun
9	1	16	20	19.829	51	218123	56	256659.6	Overrun	Overrun
9	2	16	20	19.829	51	218123	51	198425.8	Underrun	Underrun
10	1	10	12	28.159	40	187244	39	161445.1	Underrun	Underrun
10	2	10	12	28.159	40	187244	38	196191.8	Underrun	Overrun
11	1	22	28	16.427	89	411036	98	468198.7	Underrun	Overrun
11	2	22	28	16.427	89	411036	94	423739.7	Underrun	Overrun

4.4 Number of MC Runs

One of the most important aspects of an MC simulation is the number of runs that are performed. There is often a trade-off between the number of runs, the standard error, and the computational time. As the number of runs increases, the standard error decreases whereas the computational time increases. However, the relationship is not linear and there is a point which the decrease in the standard error does not justify the increase in computation time. Accordingly, the number of MC runs is increased and the standard errors of the MAE and MAPE are calculated for each scenario. **Error! Reference source not found.** to Figure 22 shows the relationship between the s tandard error and the number of runs.

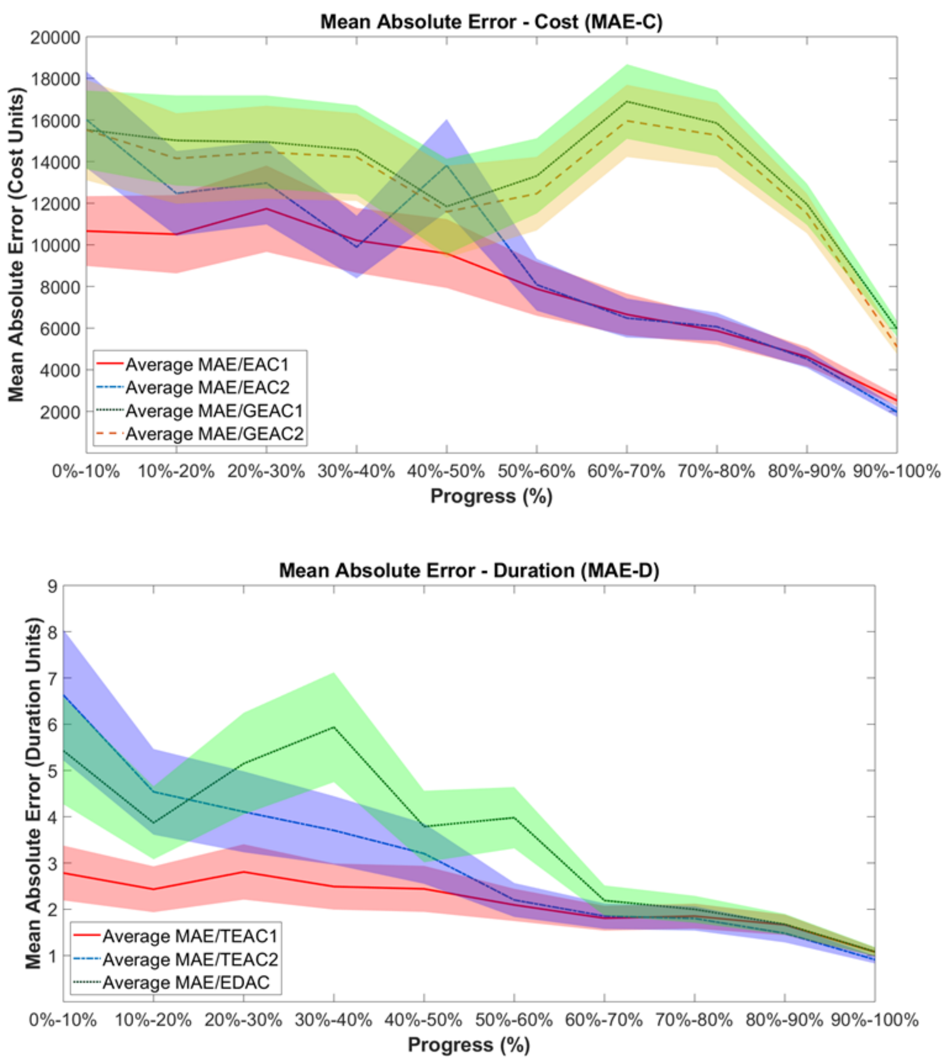


Figure 20 MAE in cost and duration forecasting methods for 3 projects/4 Markov runs

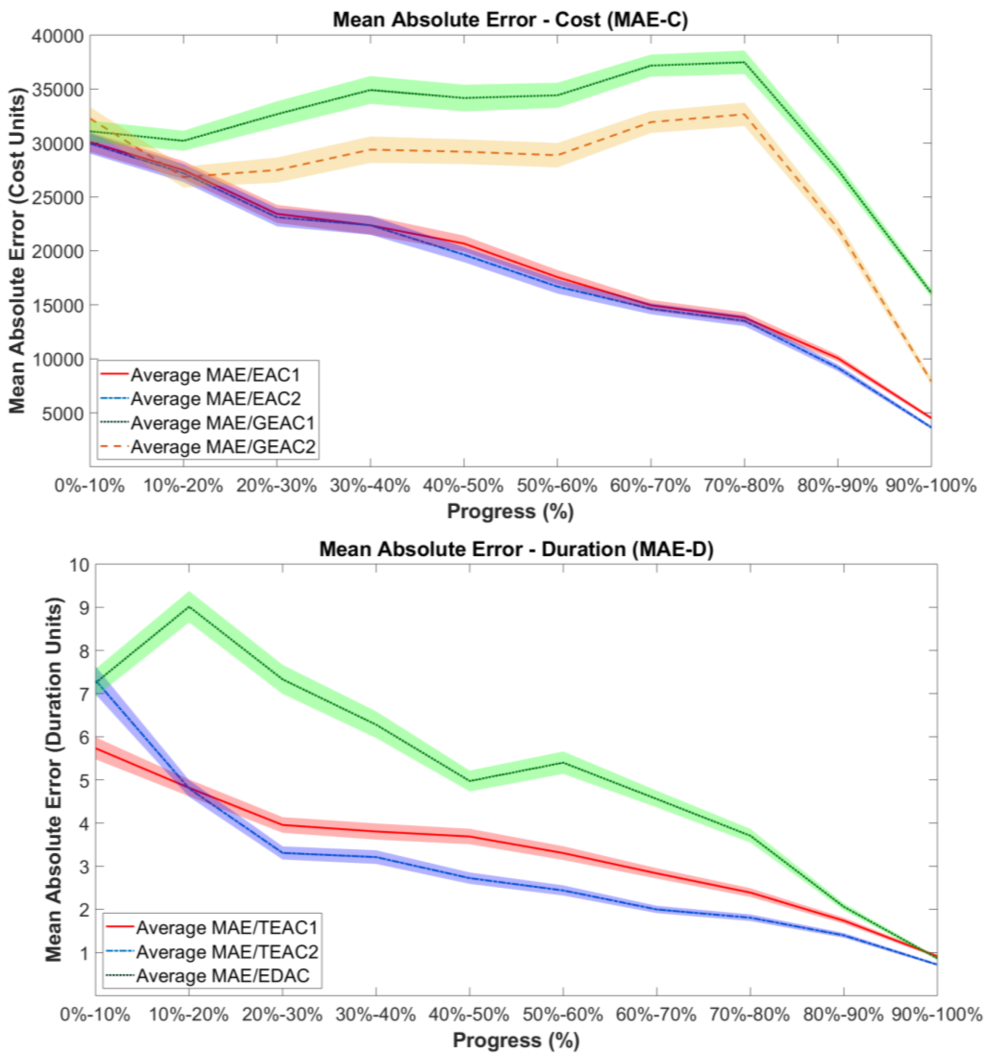


Figure 21 MAE in cost and duration forecasting methods for 10 projects/10 Markov runs

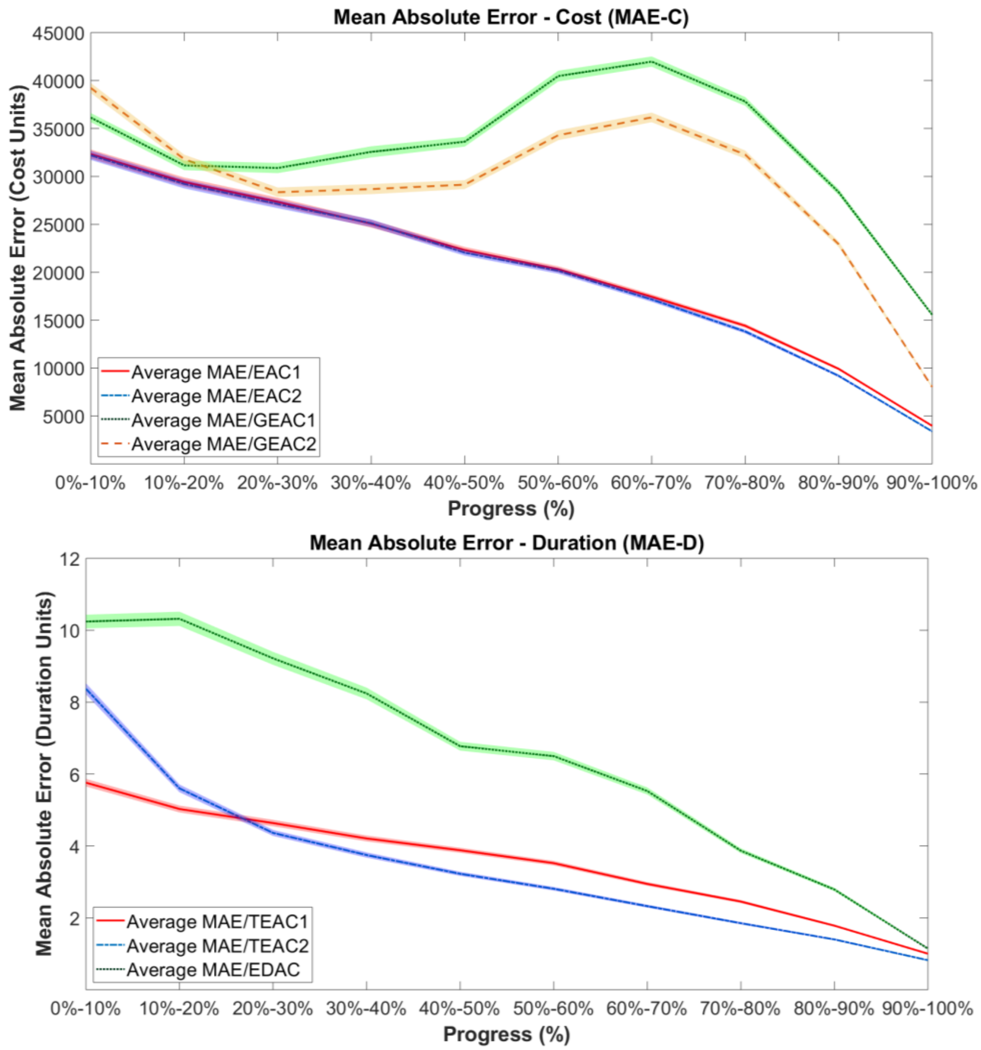


Figure 22 MAE in cost and duration forecasting methods for 10 projects/50 Markov runs

The MAE graphs are shown for the cost and duration forecasting methods. The shaded region around each line represents the standard error; the bigger the error, the thicker this region. The topmost graphs in Figure 20 portray the results of creating 3 projects and running each project 4 times, giving a total of 12 runs. The shaded region around the bar is quite large which indicates that the mean of the sample set (shown by the line in the center) is far from the true mean of the population thus illustrating low accuracy and reliability of the experiment. The experiment is then repeated with 10 projects and 10 Markov runs for each project, a total of 100 runs. The shaded region around each line narrows significantly as shown by the middle graphs in Figure 22, indicating a reduction in the standard error. Finally, the experiment is repeated with 50 projects and 10 Markov runs, a total of 500 runs. Although the shaded bands narrow slightly as compared to the scenario with 100 runs, the decrease in the standard error is not as significant and does not justify further increasing the runs. Thus the optimum number of runs is determined as being 500 runs, and the remaining results are presented for this scenario.

4.5 Performance Measures

After applying the different forecasting methods, their performance is compared in terms of accuracy, timeliness, and stability (discussed in Chapter 3). Accuracy is measured using the MAE and MAPE. Timeliness is measured in terms of accuracy achieved at certain percentage complete or actual progress points. Stability is compared using the coefficient of variation which is a standardized measure of dispersion of the forecasts around their mean. Methods which forecast cost at completion, EVA and NLR-GGM, are compared against each other. On the other hand, methods that predict the total project duration at completion, ES and KFFM, are compared against each other. The methods are compared using all created projects irrespective of their complexity or length to get a holistic view of their performance.

4.5.1 Accuracy

The accuracy of a method shows how close the forecasted value of cost or duration is to the actual values at completion. The lower the MAE and MAPE, the more accurate a method is. MAE for cost methods is expressed in units of cost making it difficult to interpret, for e.g. if a method forecasts the cost at completion to be $80,000 whereas the actual completion cost is $82,000, the MAE is $2,000. Similarly, if the cost at completion is forecasted as $8,000 whereas the actual completion cost is $10,000, the MAE is also $2,000. However, intuitively the first prediction is considered more accurate due to the scale of the values. Accordingly, the MAPE, the weighted version of MAE, is advantageous as it is expressed as a percentage. For the first case, the MAPE is 2.43% and for the second case is 20% indicating the importance of using the MAPE.

Figure 23 and Figure 24 show the MAE and the MAPE for the cost methods, respectively. EAC1 and EAC2 represent EVA with two assumptions as mentioned above. GEAC1 and GEAC2 represent the NLR-GGM, where GEAC2 is a modification of GEAC1 that accounts for schedule performance. The figures indicate that throughout the entire duration of the project, EVA outperforms the NLR-GGM. Both assumptions in the EVA method have fairly similar accuracies with the lines overlapping over the duration of the project. On the contrary, accounting for schedule progress in the Gompertz model reduces the error in forecasting after achieving 20% completion.

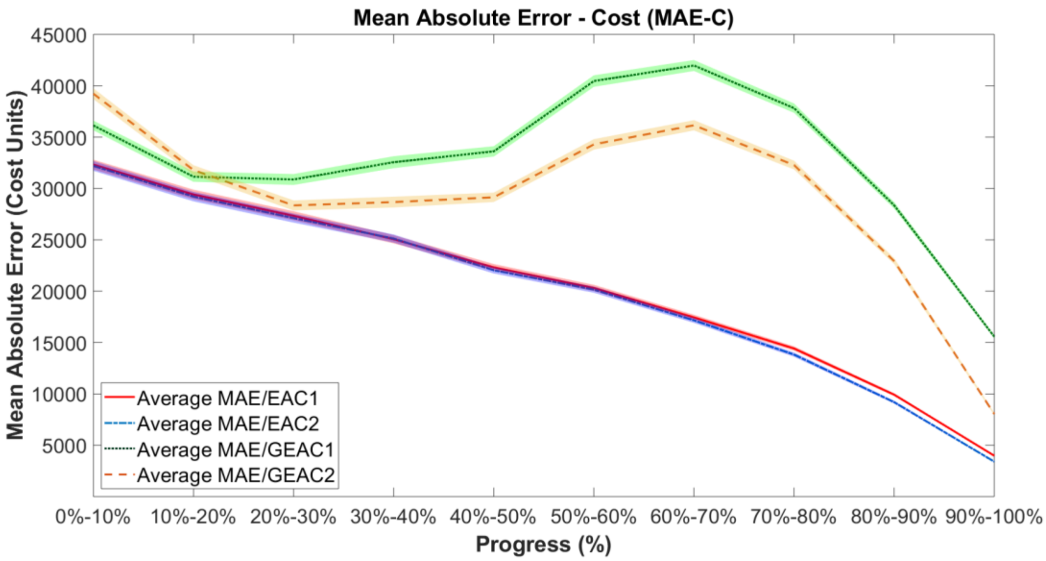


Figure 23 MAE for cost forecasting methods (all projects)

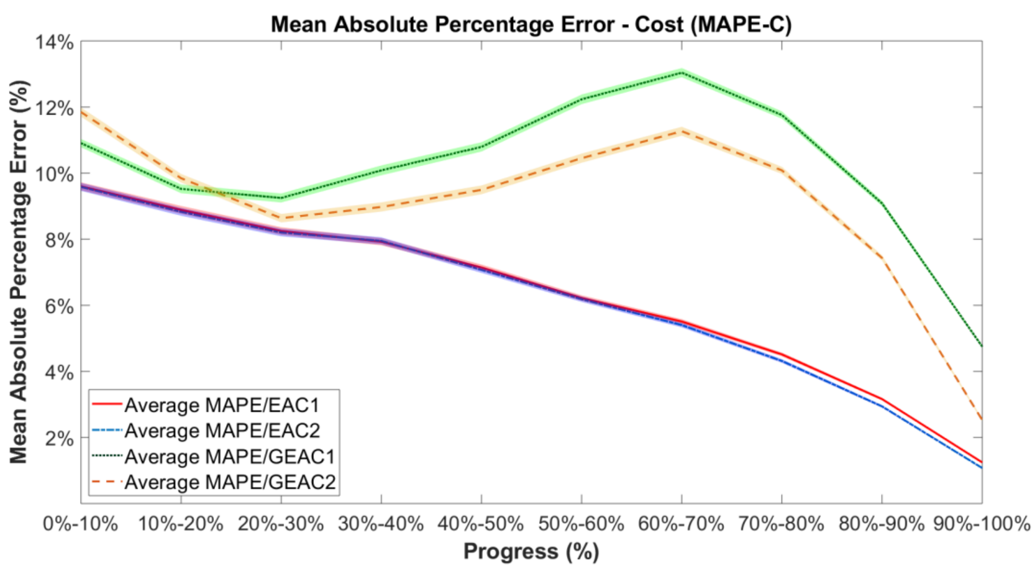


Figure 24 MAPE for cost forecasting methods (all projects)

Figure 25 and Figure 26 show the MAE and MAPE for the duration forecasting methods respectively. The red (TEAC1) and blue (TEAC2) lines represent the ES method with the two assumptions described above similar to EVA, whereas the green line (EDAC) represents the KFFM application. ES outperforms the KFFM throughout the duration of the project. The assumption that remaining work will be performed based on the original plan and deviations will not materialize into the future is shown by TEAC1 and results in a lower error at earlier stages of the project. After 30% progress, assuming that remaining work will be performed while continuing to carry out work at the same rate (TEAC2) results in more accurate forecasts.

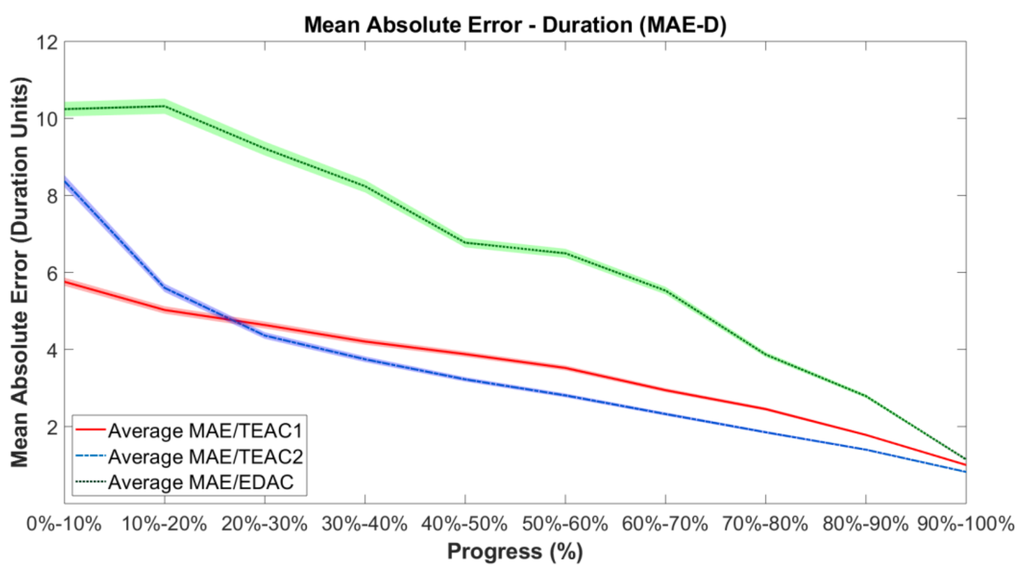


Figure 25 MAE for duration forecasting methods (all projects)

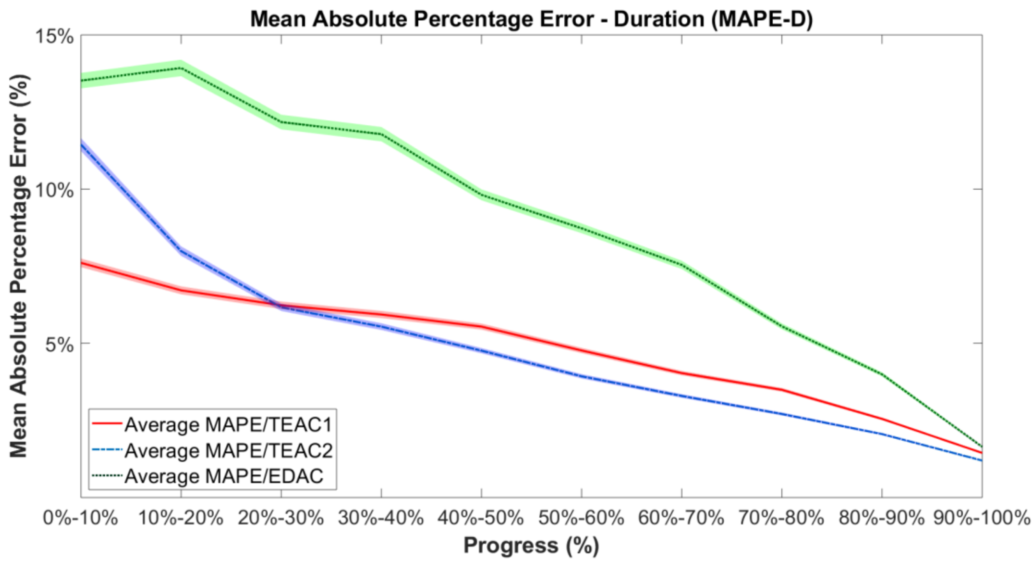


Figure 26 MAPE for duration forecasting methods (all projects)

4.5.2 Timeliness

The timeliness performance of the different cost and duration forecasting models can also be seen in Figure 23–Figure 26. The *x*-axis for the above figures represents the actual progress in 10% increments. As projects progress, changes become more costly and the number of changes decrease; thus, forecasts are expected to converge towards the actual values at completion, and forecasting accuracy is expected to increase. This relationship is depicted by all duration and cost forecasting methods except the NLR-GGM, which begins to diverge from the final cost at completion. This can be attributed to the exponential function of the Gompertz model. As the project progresses beyond 40% complete, the exponential function begins to tighten giving less flexibility to fit the performance data to it. Consequently, the error increases until a point where the actual performance follows the same trend as the exponential function (flattens out towards the end), resulting in the error decreasing again.

4.5.3 Stability

The smaller the CV, the less dispersed data are around the mean. For all the cost forecasting methods, the CV is quite small indicating that all the methods are stable as shown in Figure 27 and

Figure 28. Unlike the accuracy and timeliness performance, GEAC1 has a lower CV than EAC2, meaning that it is more stable. However, the difference is less than 0.01 and can be considered insignificant. For the duration forecasting methods, the CV is also quite small, indicating that all methods are stable. The KFFM has the highest CV out of all methods, which can be attributed to its sensitivity to changes in the ES from one period to the next.

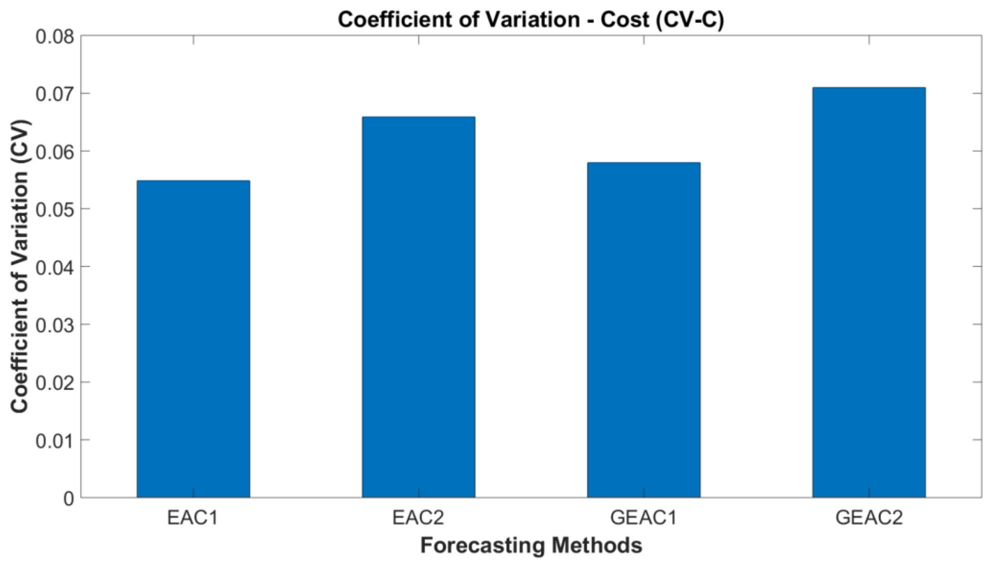


Figure 27 CV for cost forecasting methods (all projects)

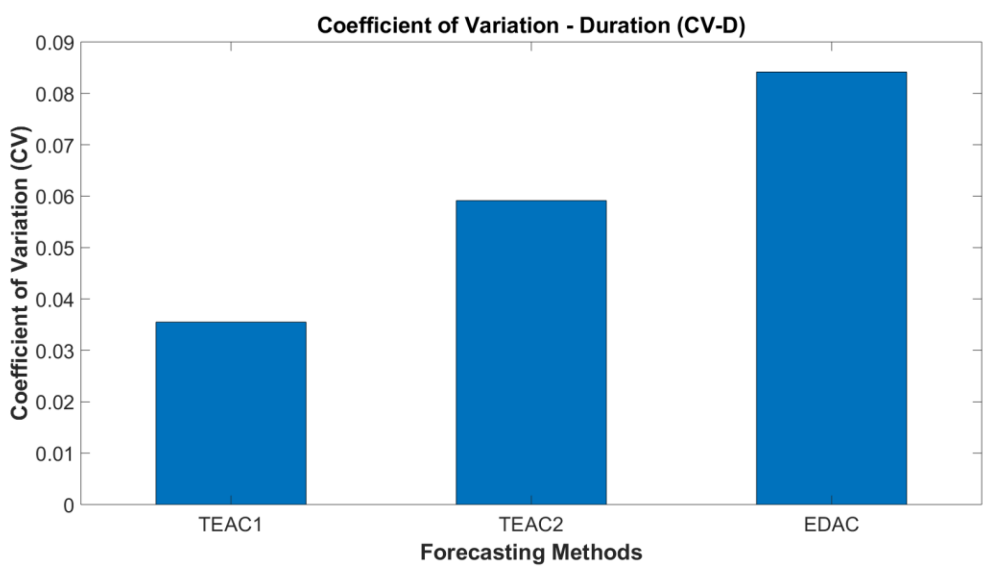


Figure 28 CV for cost forecasting methods (all projects)

4.6 Performance Measures (Complexity)

The previous section discussed the performance of the different forecasting models for all types of projects. In this section, the effect of project schedule complexity on forecasting method performance is discussed. Different methods will be compared at the same complexity, and the same methods will be compared at varying complexities. Schedule complexity, expressed as a percentage, can be interpreted as shown in Table 8.

Table 8 Interpretation of complexity index (K. M. Nassar and Hegab 2006)

Complexity Index	Interpretation
0% - 30%	Acceptable
30% - 50%	Satisfactory, but may be improved
50% - 70%	Consider review schedule
70% - 100%	Seriously consider review schedule

4.6.1 Accuracy

Table 9 shows the MAPE for the EVA and NLR-GGM models at varying project complexities. The results at each complexity level are consistent with the previous results; EVA outperforms the NLR-GGM for the entire duration of the project. For each method, as the project complexity increases, the MAPE also increases particularly exceeding 50% complexity.

Table 9 MAPE for cost forecasting methods at varying complexities

Complexity	Progress	MAPE/EAC1	MAPE/EAC2	MAPE/GEAC1	MAPE/GEAC2
0% - 30%	0% - 20%	12%	12%	13%	14%
	20% - 40%	10%	10%	12%	11%
	40% - 60%	8%	9%	14%	12%
	60% - 80%	6%	5%	15%	13%
	80% - 100%	3%	2%	8%	6%
30% - 50%	0% - 20%	12%	12%	14%	15%
	20% - 40%	11%	11%	14%	13%
	40% - 60%	9%	9%	15%	14%
	60% - 80%	6%	6%	15%	14%
	80% - 100%	3%	3%	9%	7%
50% - 70%	0% - 20%	15%	15%	20%	19%
	20% - 40%	10%	9%	12%	12%
	40% - 60%	9%	8%	17%	16%
	60% - 80%	7%	6%	18%	17%
	80% - 100%	3%	3%	10%	9%
70% - 100%	0% - 20%	19%	19%	19%	20%
	20% - 40%	14%	15%	26%	23%
	40% - 60%	11%	12%	20%	17%
	60% - 80%	7%	7%	17%	15%
	80% - 100%	3%	3%	10%	8%
Scale		**Larger**			**Smaller**

Table 10 shows the MAPE for the ES and KFFM models at varying project complexities. Similar to the cost methods, the duration methods also exhibit the same performance as above (all projects) at different complexities; ES outperforms the KFFM. However, unlike the cost methods, the forecasting error does not differ significantly with the increase in complexity for each of the schedule methods.

Table 10 MAPE for duration forecasting methods at varying complexities

Complexity	Progress	MAPE/TEAC1	MAPE/TEAC2	MAPE/EDAC
0% - 30%	0% - 20%	9%	12%	14%
	20% - 40%	8%	8%	13%
	40% - 60%	7%	6%	10%
	60% - 80%	5%	4%	7%
	80% - 100%	3%	2%	3%
30% - 50%	0% - 20%	9%	13%	13%
	20% - 40%	8%	9%	13%
	40% - 60%	7%	6%	10%
	60% - 80%	5%	4%	7%
	80% - 100%	3%	2%	3%
50% - 70%	0% - 20%	11%	14%	15%
	20% - 40%	8%	8%	8%
	40% - 60%	7%	7%	10%
	60% - 80%	7%	5%	6%
	80% - 100%	4%	3%	3%
70% - 100%	0% - 20%	12%	13%	11%
	20% - 40%	8%	6%	16%
	40% - 60%	7%	6%	9%
	60% - 80%	7%	5%	7%
	80% - 100%	5%	4%	3%
Scale		**Larger**		**Smaller**

4.6.2 Timeliness

Table 9 and Table 10 also show the timeliness performance for the different cost and schedule forecasting models. For all the models at increasing complexities, the forecasting accuracy increases as the project progresses. Different models of the same method (i.e. EAC1 and EAC2 in EVA, GEAC1 and GEAC2 in NLR-GGM, and TEAC1 and TEAC2 in ES) exhibit similar behavior with the accuracies being within 1–2% at each reporting period in the different forecasting methods. However, the performance exhibited by the NLR GGM at around 40% progress for the different complexities is the same as that for all projects; i.e. the forecasting errors increase and then begin to decrease again due to the exponential function curve fitting, as explained above.

4.6.3 Stability

As established above, the CV for all cost and schedule forecasting methods indicates that the methods are stable. As shown in Figure 29 and Figure 30, the methods also demonstrate high forecasting stability as complexities are varied. However, Figure 29 shows that, for all cost forecasting methods, as the complexity increases, the CV increases and the forecasting stability decreases. This is not the case for the schedule forecasting methods. Figure 30 shows that the forecasting stability remains relatively constant for different project complexities.

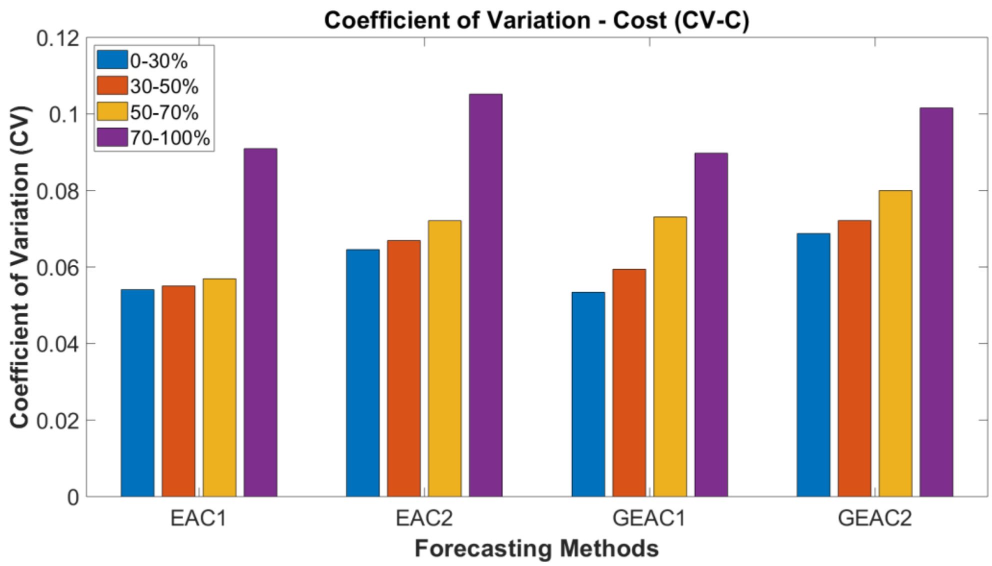


Figure 29 CV for cost forecasting methods at varying complexities

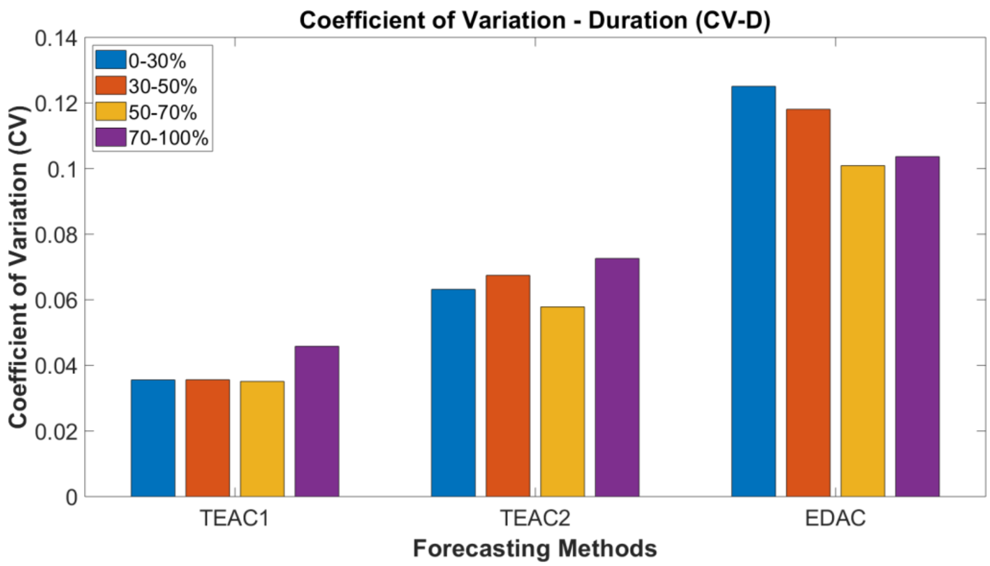


Figure 30 CV for duration forecasting methods at varying complexities

4.7 Verification and Validation

Developers and users of simulation models as well individuals who make decisions based on the output of these models are often concerned with whether the output of the model is correct. Verification and validation techniques are useful in answering this question. Model verification is defined as ensuring that the computer programming and implementation of the conceptual model are correct. Validation ensures that the simulation model is an accurate reflection of the real world system it is representing. In the case of this experiment, model verification and validation are essential for reliable decision-making based on the results.

For verification of the model, it was important to ensure that the code which automated the process of creating random projects, generating actual progress variations, and applying the different forecasting methods was correct. This process was simulated using a higher level programming language, C#. Sargent (2007) explains that when using a higher level programming language the chance of errors is higher than when a special-purpose simulation language is used. Accordingly, the development needs to be implemented using software engineering techniques, and verification is primarily concerned with "determining that the simulation functions and the computer model have been programmed and implemented correctly" (Sargent 2007). Testing simulation software can be done using two approaches, static and dynamic testing. For the given experiment, verification was mainly done using dynamic testing. In this approach, the variables of the model are changed, and the values obtained are used to determine whether the development itself and its implementation are correct. Specific techniques applied include manual tracing of the output to ensure that all calculations are correct by comparing them to a basic Excel prototype model and investigating the relationships between inputs and outputs (e.g. when changing the adjustments to progress and expenditure for the different states of the Marko chain process, how does this affect the output (actual performance variations)?).

Validation of the proposed experiment is mainly concerned with ensuring that the generated random project networks resemble actual construction projects' networks. This is crucial for the results of the experiment to be valid and reliable. The validation technique used to guarantee this is face validation. Face validation is described by Sargent (2007) as asking individuals with knowledge about the system whether the model and its output are acceptable. Thus, this method was applied and sample of generated projects was sent to experts with knowledge of construction projects' network topology to establish that the output sample of projects are reasonable.

4.8 Discussion

This chapter showed the application of the MC simulation-based experiment on validating project control models in a more efficient and reliable way. Four methods were discussed, EVA, NLR-GGM, ES and KFFM. The first two are concerned with forecasting cost estimate at completion, while the second two predict duration estimate at completion. Different models are used within each method based on a number of assumptions. After creating a large number of random projects and generating 10 Markov-based actual variations for each project, the methods are applied and actual performance data is fed into the methods periodically to resemble what occurs on a typical construction site.

First, the methods are tested with the entire set of generated projects. The results showed that in terms of accuracy and timeliness, EVA and ES outperform NLR-GGM and KFFM methods, respectively. The results contradict the results reported by the researchers who developed these models. Narbaev and De Marco (2014) compared the accuracy of NLR-GGM with EVA on eight completed projects in the early, middle, and late stages of the project. The results showed that the proposed model outperformed EVA in all eight projects and in the different stages. These results contradict with the behavior exhibited when applying the MC simulation-based experiment and show the importance of reliable validation. Using eight actual projects with possible incorrect/missing data does not justify the results of the validation experiment. On the other hand, the MC simulation-based experiment creates random projects with varying number of activities, relationships, complexities, durations, and costs that are then validated to ensure these projects resemble actual construction projects. This creates the basis for a reliable validation experiment that can be used in a number of applications. Kim and Reinschmidt (2010) compared the output of the KFFM with ES using a single project and reported improvements over ES. Similarly, the validation results are unreliable, and after applying the MC simulation-based experiment, the results differed completely.

In addition to validation, the experiment provides further insight into the developed methods by allowing the user to test different scenarios and understand their effects on performance. This was portrayed by studying the effect of project complexity on model performance in terms of accuracy, timeliness, and stability. With the limitations in actual project data and its availability, obtaining projects with varying complexities would be next to impossible. Accordingly, the experiment opens up the door for different validation scenarios that would not be feasible using actual project data.

CHAPTER 5: CONCLUSIONS

5.1 Summary of the Work

An MC simulation-based experiment was designed to create a basis for testing and validating different methodologies in construction management. The focus of this research was to apply the simulation in testing different project control methods and comparing the results of using actual project data for validation to using MC simulation. The design of the experiment involved creating random project networks that resemble actual baseline plans of construction projects, generating actual progress variations that mimic what happens on the site when these projects are executed, and developing different performance measures to compare the tested forecasting models.

Generating project networks is comprised of five main steps. First, the number of work packages included in the project are determined. These are randomly assigned to tiers/columns, which determine their position in the project network. Second, the relationships among the work packages are generated. This step relies on a random weighted selection technique. Work packages in immediately succeeding tiers are given a higher weight increasing the likelihood of being linked to the current work package than work packages in subsequent tiers. This method of creating links among work packages to determine the different successors and predecessors results in the creation of redundancies. Thus, the next step is eliminating these redundancies, as they provide a false sense of complexity to a project. This is done systematically using a method developed by Bashir (2010). Next, the duration and cost of each work package are sampled from predefined distributions. Finally, CPM calculations are performed on the generated project networks to determine the total project duration and cost.

A Markov process is used to simulate the randomness and uncertainty that occurs when planned projects are executed. A number of outcomes are possible, and projects usually end up with a cost and schedule overrun, a cost and schedule underrun, a cost overrun and schedule underrun, or a cost underrun and schedule overrun. In the proposed method, the projects can occupy one of four states in each period: A – outstanding performance, B – within target performance, C – below target performance, or D – poor performance. The probability of transitioning from the current state to a future state is determined by the transition probability matrix. In static Markov chains, the transition matrix is constant and probabilities are the same regardless of whether the project is beginning or is near completion. However, this is unrealistic as probabilities of transitioning from one state to another are a function of progress. In other words, the probability of transitioning from state A to state C is higher at the beginning of the project than at the end. Accordingly, a dynamic Markov chain is used in which the transition probabilities vary from one period to the next based on the actual progress. After determining the state in each period, adjustments are made to the planned progress and cost of activities ongoing during that period. In every run of the Markov process, a different outcome is achieved, creating a large sample dataset with varying outcomes.

Finally, the different performance measures are identified to compare forecasting models. The first is accuracy, which shows how close the forecasted value is in each period to the actual value at completion. Accuracy measures adopted in this research include the MAE and MAPE. MAE is unbiased but difficult to interpret, while MAPE is biased towards negatively skewed projections but is easy to interpret as it is a percentage. Thus, both measures are used to overcome the disadvantages of the other. Second, the timeliness of a method is evaluated, i.e. the speed at which the forecast converges to the actual value. This is measured by the progress at which a certain level of accuracy is achieved. Finally, forecasting stability indicates how much the forecasts vary from one period to the next. It is measured using the coefficient of variation, which calculates the dispersion of the data round the mean.

After designing the experiment, it is tested with four forecasting methods, EVA, NLR-GGM, ES, and KFFM. The methods are first applied on the entire dataset of Markov runs for each project, and the performance is assessed using the aforementioned measures of accuracy, timeliness, and stability. Next, the projects are divided according to complexity (measured as a percentage), and the performance is assessed to determine the effect of project complexity on forecasting accuracy, timeliness, and stability. Table 11 shows a summary of the experiment testing with the cost methods ranked from 1-4 in terms of performance, 1 being the best and 4 being the worst. The schedule forecast methods are also ranked on a scale from 1-3.

Table 11 Summary of experiment testing

Scenario	Cost Forecasting Models Rank in Accuracy and Timeliness (1-4)				Duration Forecasting Models Rank in Accuracy and Timeliness (1-3)		
	EVA		NLR-GGM		ES		KFFM
	EAC1	*EAC2*	*GEAC1*	*GEAC2*	*TEAC1*	*TEAC2*	*EDAC*
All Projects	2	1	4	3	2	1	3
0% - 30% Complexity	1	2	4	3	2	1	3
30% - 50% Complexity	1	1	3	2	2	1	3
50% - 70% Complexity	2	1	4	3	2	1	3
70% - 100% Complexity	1	2	4	3	2	1	3

5.2 Overall Conclusion

MC simulation is a useful technique for modeling a system using random sampling. In a typical deterministic system, inputs are represented as single values; and a number of scenarios, the best worst, and most likely scenarios, are tested. This approach has a number of drawbacks: 1) it ignores the effect of uncertainty on input values; 2) it may be difficult to determine the exact and correct value for each input parameter; and 3) the same scenario for each input parameter may not occur at the same time. MC simulation is useful, as it allows users to methodically investigate a number of scenarios involving stochastic input parameters.

Accordingly, the concept of MC simulation was applied in designing the proposed experiment. The inputs represented by statistical distributions include the network topology (i.e. activities and relationships). Using these inputs and a number of predefined rules, an AON network is created. Finally, the output of the MC simulation experiment is a set of randomly generated AON networks. By further applying a Markov chain simulation process to model executional variations in performance, the entire solution set of possible projects is created. This provides a more reliable method of validation as compared to using actual project data, which are limited, can have incorrect/missing entries, and are affected by managerial decisions taken during the execution of the projects.

The importance of relying on MC simulation for validation rather than using actual project data is also magnified by testing the experiment on different forecasting methods. EVA and its extensions, e.g. ES, are the most well-known and applied methods for project control in construction. However, their limitations have encouraged researchers to come up with other methods using techniques in the literature such as Bayesian inference, Markov chains, artificial intelligence, and others. Over the years, numerous methods have been developed, and when compared to EVA or ES using actual project data, the results often indicate that these methods outperform EVA. NLR-GGM and KFFM were compared using the same techniques, and the outcomes were similar. However, when applying the proposed experiment, the results did not confirm the outcome of previous research, suggesting that actual project validation is flawed, inadequate, and yields erroneous results. With further testing it can be shown that EVA and its extensions, despite their limitations, remain the most reliable and easily applied project control methods.

5.3 Contributions

This thesis focused on developing a methodology for creating project baseline schedules and actual progress variations using Monte Carlo simulation and Markov chains, respectively. By doing so, a larger more representative sample of construction projects with a range of possible outcomes. These projects were used as a basis for enhancing validation experiments in construction, which generally rely on actual data that has several shortcomings as discussed above. Using this technique, the results of the validations experiments are more robust and sound and can be more confidently generalized as opposed to case-based studies.

The experiment was particularly designed for evaluating the performance of forecasting methods for project controls in an unbiased and simulated environment that allows several scenarios to be tested, as demonstrated previously. Not only is this technique useful for validation, but it also provides a better insight into forecasting method performance. By advancing the understanding and application of forecasting methods, project managers benefit greatly, and project execution is significantly enhanced.

5.4 Recommendations for Future Research

The following is recommended for further research:

- Currently, projects are generated with dummy work packages that do not resemble actual work packages. Creating different types of projects by understanding the work packages involved would allow further testing of developed methodologies.

- Generated projects can be further complicated by adding resources, risks, and interruptions to more closely mimic actual projects and understand the impact for validation purposes.

- The scope of this research focused on testing the developed experiment with different forecasting strategies. The experiment can be expanded for use in other applications, such as risk management or comparing project delivery methods.

REFERENCES

Abdel Azeem, S.A., Hossam E. Hosny, and Ahmed H. Ibrahim. 2014. "Forecasting Project Schedule Performance Using Probabilistic and Deterministic Models." *HBRC Journal* 10 (1): 35–42. https://doi.org/10.1016/j.hbrcj.2013.09.002.

Acebes, Fernando, María Pereda, David Poza, Javier Pajares, and José Manuel Galán. 2015. "Stochastic Earned Value Analysis Using Monte Carlo Simulation and Statistical Learning Techniques." *International Journal of Project Management* 33 (7): 1597–1609. https://doi.org/10.1016/j.ijproman.2015.06.012.

Agrawal, M. K., S. E. Elmaghraby, and W. S. Herroelen. 1996. "DAGEN: A Generator of Testsets for Project Activity Nets." *European Journal of Operational Research* 90 (2): 376–82. https://doi.org/10.1016/0377-2217(95)00361-4.

Akileswaran, V., G. B. Hazen, and T. L. Morin. 1983. "Complexity of the Project Sequencing Problem." *Operations Research* 31 (4): 772–78. https://doi.org/10.2307/170789.

Amelin, Mikael. 2013. "Monte Carlo Methods in Engineering." https://doi.org/10.1063/1.53528.

Anbari, Frank. 2012. "The Earned Schedule." In *PMI® Research and Education Conference*. Limerick, Munster, Ireland: Project Management Institute. https://www.pmi.org/learning/library/evm-cpm-evaluate-project-performance-6355.

Anbari, Frank T. 2003. "Earned Value Project Management Method and Extensions." *Project Management Journal* 34 (4): 12–23. https://doi.org/10.1109/EMR.2004.25113.

Baik, Hyeon-Shik, Hyung Seok Jeong, and Dulcy M. Abraham. 2006. "Estimating Transition Probabilities in Markov Chain-Based Deterioration Models for Management of Wastewater Systems." *Journal of Water Resources Planning and Management* 132 (1): 15–24. https://doi.org/10.1061/(ASCE)0733-9496(2006)132:1(15).

Barraza, Gabriel A., W. Edward Back, and Fernando Mata. 2004. "Probabilistic Forecasting of Project Performance Using Stochastic S Curves." *Journal of Construction Engineering and Management* 130 (2): 235–44. https://doi.org/10.1061/(ASCE)0733-9364(2004)130.

Barton, Russell R. 2004. "Designing Simulation Experiments." *Proceedings of the 2004 Winter Simulation Conference*, no. January 2004: 73–79.

Bashir, H. A. 2010. “Removal of Redundant Relationships in an AON Project Network for Evaluating Schedule Complexity.” *Journal of Construction Engineering and Management* 136 (7): 787–93. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000181.

Boussabaine, A. Halim. 1996. “The Use of Artificial Neural Networks in Construction Management: A Review.” *Construction Management and Economics* 14 (5): 427–36. https://doi.org/10.1080/014461996373296.

Browning, Tyson R., and Ali A. Yassine. 2010. “A Random Generator of Resource-Constrained Multi-Project Network Problems.” *Journal of Scheduling* 13 (2): 143–61. https://doi.org/10.1007/s10951-009-0131-y.

Caron, Franco, Fabrizio Ruggeri, and Alessandro Merli. 2013. “A Bayesian Approach to Improve Estimate at Completion in Earned Value Management.” *Project Management Journal* 44 (1): 3–16. https://doi.org/10.1002/pmj.21303.

Davies, E. M. 1973. “An Experimental Investigation of Resource Allocation in Multiactivity Projects.” *Operational Research Quarterly* 24 (4): 587–91. https://doi.org/10.1057/jors.1973.107.

Demeulemeester, E., B. Dodin, and W. Herroelen. 1993. “Random Activity Network Generator.” *Operations Research* 41 (5): 972–80. https://doi.org/10.1287/opre.41.5.972.

Du, Jing, Byung Cheol Kim, and Dong Zhao. 2016. “Cost Performance as a Stochastic Process: EAC Projection by Markov Chain Simulation.” *Journal of Construction Engineering and Management* 142 (6): 1–13. https://doi.org/10.1061/(ASCE)CO.1943-7862.0001115.

Kaimann, Richard A. 1974. “Coefficient of Network Complexity.” *Management Science* 21 (2): 172–77. https://doi.org/10.1287/mnsc.21.2.172.

Khafri, Amin Amini. “A Framework for Forecasting Project Estimate at Completion Using Historical and Current Performance Data.” PhD dissertation., University of Alberta, 2018.

Kim, Byung-cheol. “Forecasting Project Progress and Early Warning of Project Overruns with Probabilistic Methods.” PhD dissertation., Texas A&M University, 2007.

Kim, Byung Cheol, and Kenneth F. Reinschmidt. 2009. “Probabilistic Forecasting of Project Duration Using Bayesian Inference and the Beta Distribution.” *Journal of Construction*

Engineering and Management 135 (3): 178–86. https://doi.org/10.1061/(ASCE)0733-9364(2009)135:3(178).

———. 2010. "Probabilistic Forecasting of Project Duration Using Kalman Filter and the Earned Value Method." *Journal of Construction Engineering and Management* 136 (8): 834–43. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000192.

———. 2011. "Combination of Project Cost Forecasts in Earned Value Management." *Journal of Construction Engineering and Management* 137 (11): 958–66. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000352.

Knight, Karla, and Aminah Robinson Fayek. 2002. "Use of Fuzzy Logic for Predicting Design Cost Overruns on Building Projects." *Journal of Construction Engineering and Management* 128 (6): 503–12. https://doi.org/10.1061/(ASCE)0733-9364(2002)128:6(503).

Ko, Chien Ho, and Min Yuan Cheng. 2007. "Dynamic Prediction of Project Success Using Artificial Intelligence." *Journal of Construction Engineering and Management* 133 (4): 316–24. https://doi.org/10.1061/(ASCE)0733-9364(2007)133:4(316).

Kolisch, Rainer, Arno Sprecher, and Andreas Drexl. 1995. "Characterization and Generation of a General Class of Resource-Constrained Project Scheduling Problems." *Management Science* 41 (10): 1693–1703. https://doi.org/10.1287/mnsc.41.10.1693.

Lee, Dong Eun. 2005. "Probability of Project Completion Using Stochastic Project Scheduling Simulation." *Journal of Construction Engineering and Management* 131 (3): 310–18. https://doi.org/10.1061/(ASCE)0733-9364(2005)131:3(310).

Lerche, Ian, and Brett S. Mudford. 2005. "How Many Monte Carlo Simulations Does One Need to Do?" *Energy Exploration and Exploitation* 23 (6): 405–27. https://doi.org/10.1260/014459805776986876.

Li, Ningyuan, Wei-Chau Xie, and Ralph Haas. 1996. "Reliability-Based Processing of Markov Chains for Modeling Pavement Network Deterioration." *Transportation Research Record: Journal of the Transportation Research Board* 1524 (1): 203–13. https://doi.org/10.1177/0361198196152400124.

Ling, Florence Yean Yng, and Min Liu. 2004. "Using Neural Network to Predict Performance of Design-Build Projects in Singapore." *Building and Environment* 39 (10): 1263–74.

https://doi.org/10.1016/j.buildenv.2004.02.008.

Lipke, Walt. 2012. "Earned Schedule Contribution to Project Management." *PM World Journal* 1 (2). www.pmworldlibrary.net.

Lipke, Walt, and Kym Henderson. 2006. " Earned Schedule: An Emerging Enhancement to Earned Value Management." *Cross Talk* 19 (6). https://www.researchgate.net/publication/277764079_Earned_Schedule_An_Emerging_Enhancement_to_Earned_Value_Management.

Lu, Ming, and Hoi Ching Lam. 2009. "Transform Schemes Applied on Non-Finish-to-Start Logical Relationships in Project Network Diagrams." *Journal of Construction Engineering and Management* 135 (9): 863–73. https://doi.org/10.1061/(ASCE)CO.1943-7862.0000062.

Markov. 2003. *Markov Chains*. Edited by Pearson Education. Addison-Wesley. https://people.cas.uab.edu/~mosya/teaching/MarkovChains.pdf.

Microsoft Visual Studio Community 2017 (version 15.9.7). Windows. Microsoft Corporation. 2017

Moselhi, Osama, Tarek Hegazy, and Paul Fazio. 1991. "Neural Networks as Tools in Construction." *Journal of Construction Engineering and Management* 117 (4): 606–25. https://doi.org/10.1061/(ASCE)0733-9364(1991)117:4(606).

Narbaev, Timur, and Alberto De Marco. 2014. "An Earned Schedule-Based Regression Model to Improve Cost Estimate at Completion." *International Journal of Project Management* 32 (6): 1007–18. https://doi.org/10.1016/j.ijproman.2013.12.005.

Nassar, Khaled M., and Mohamed Y. Hegab. 2006. "Developing a Complexity Measure for Project Schedules." *Journal of Construction Engineering and Management* 132 (6): 554–61. https://doi.org/10.1061/(ASCE)0733-9364(2006)132:6(554).

Nassar, Nadim Kamil. "An Integrated Framework for Evaluation, Forecasting and Optimization of Performance of Construction Projects." PhD dissertation., University of Alberta, 2005.

Ortiz-García, José J., Seósamh B. Costello, and Martin S. Snaith. 2006. "Derivation of Transition Probability Matrices for Pavement Deterioration Modeling." *Journal of Transportation Engineering* 132 (2): 141–61. https://doi.org/10.1061/(ASCE)0733-947X(2006)132:2(141).

Ranjith, Shrigandhi, Sujeeva Setunge, Rebecca Gravina, and Srikanth Venkatesan. 2013.

"Deterioration Prediction of Timber Bridge Elements Using the Markov Chain." *Journal of Performance of Constructed Facilities* 27 (3): 319–25. https://doi.org/10.1061/(ASCE)CF.1943-5509.0000311.

Raychaudhuri, Samik. 2008. "Introduction to Monte Carlo Simulation." *Proceedings - Winter Simulation Conference*, 91–100. https://doi.org/10.1109/WSC.2008.4736059.

Reyck, Bert De, and Willy Herroelen. 1996. "On the Use of the Complexity Index as a Measure of Complexity in Activity Networks." *European Journal of Operational Research* 91 (2): 347–66. https://doi.org/10.1016/0377-2217(94)00344-0.

Robelin, Charles Antoine, and Samer M. Madanat. 2007. "History-Dependent Bridge Deck Maintenance and Replacement Optimization with Markov Decision Processes." *Journal of Infrastructure Systems* 13 (3): 195–201. https://doi.org/10.1061/(ASCE)1076-0342(2007)13:3(195).

Robinson Fayek, A., and Z. Sun. 2001. "A Fuzzy Expert System for Design Performance Prediction and Evaluation." *Canadian Journal of Civil Engineering* 28 (SUPPL. 1): 1–25. https://doi.org/10.1139/l00-075.

Rubinstein, Reuven Y., and Dirk P. Kroese. 2016. *Simulation and the Monte Carlo Method: Third Edition*. *Simulation and the Monte Carlo Method*. 3rd ed. Wiley Series in Probability and Statistics. Hoboken, NJ, USA: John Wiley & Sons, Inc. https://doi.org/10.1002/9781118631980.

Sargent, Robert G. 2007. "Proceedings of the 2007 Winter Simulation Conference S. G. Henderson, B. Biller, M.-H. Hsieh, J. Shortle, J. D. Tew, and R. R. Barton, Eds.," 124–37.

Shaikh, Aejaz. 2017. "Forecasting." ProjectManagement.Com. February 21, 2017. https://www.projectmanagement.com/wikis/293296/Forecasting.

Sinha, Sanjeev, Bimal Kumar, and Avril Thomson. 2006. "Measuring Project Complexity: A Project Manager's Tool." *Architectural Engineering and Design Management* 2 (3): 187–202. https://doi.org/10.1080/17452007.2006.9684615.

Sweis, Rateb J., Sharaf M. Bisharat, Lorina Bisharat, and Ghaleb J. Sweis. 2014. "Factors Affecting Contractor Performance on Public Construction Projects." *Life Science Journal* 11 (SPEC. ISSUE 4): 28–39.

Teicholz, Paul. 1994. “Forecasting Final Cost and Budget of Construction Projects.” *Journal of Computing in Civil Engineering*. https://doi.org/10.1016/0169-2070(94)90082-5.

Vanhoucke, M., and S. Vandevoorde. 2007. “A Simulation and Evaluation of Earned Value Metrics to Forecast the Project Duration.” *Journal of the Operational Research Society* 58 (10): 1361–74. https://doi.org/10.1057/palgrave.jors.2602296.

Vanhoucke, Mario, José Coelho, Dieter Debels, and Luis Valadares Tavares. 2004. “On the Morphological Structure of a Network.” *Vlerick Leuven Gent Working Paper Series 2004/9*, no. December 2015. http://ideas.repec.org/p/vlg/vlgwps/2005-9.html%5Cnhttp://www.vlerick.be/en/2611-VLK/version/default/part/AttachmentData/data/vlgms-wp-2005-9.pdf.

Vidal, Ludovic Alexandre, Franck Marle, and Jean Claude Bocquet. 2011a. “Measuring Project Complexity Using the Analytic Hierarchy Process.” *International Journal of Project Management* 29 (6): 718–27. https://doi.org/10.1016/j.ijproman.2010.07.005.

———. 2011b. “Using a Delphi Process and the Analytic Hierarchy Process (AHP) to Evaluate the Complexity of Projects.” *Expert Systems with Applications*. https://doi.org/10.1016/j.eswa.2010.10.016.

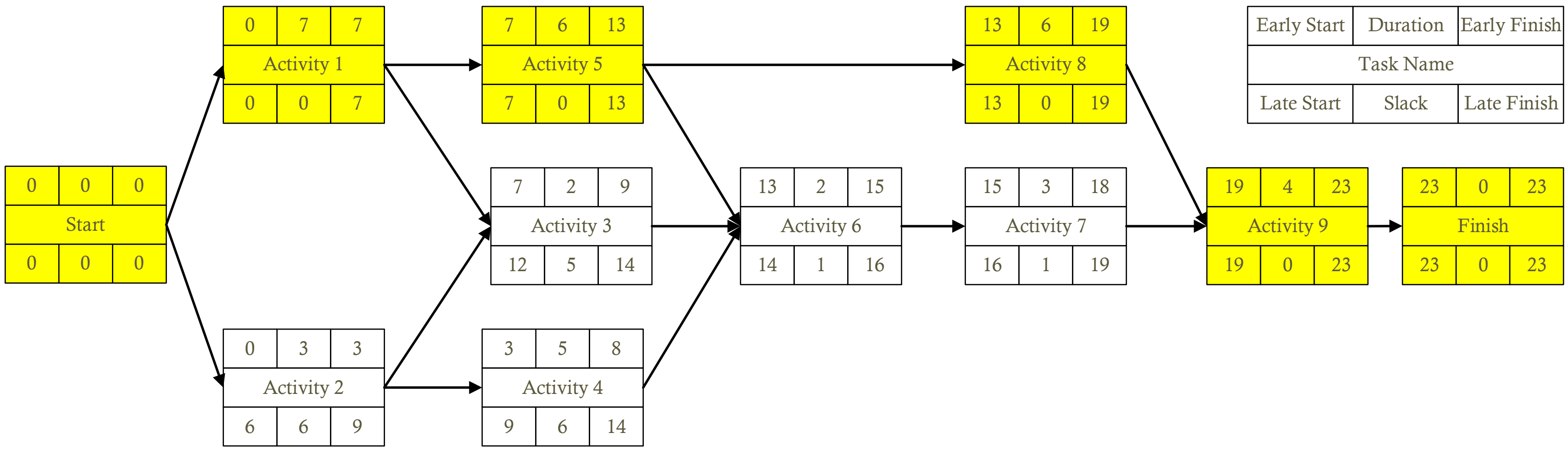
Vignesh, N, and S Sowmya. 2013. “Analysis of ‘Estimate at Completion’ of a Project’s Duration to Improve ‘Earned Value Management System.’” *International Journal of Research and Technology* 2 (10): 1056–61.

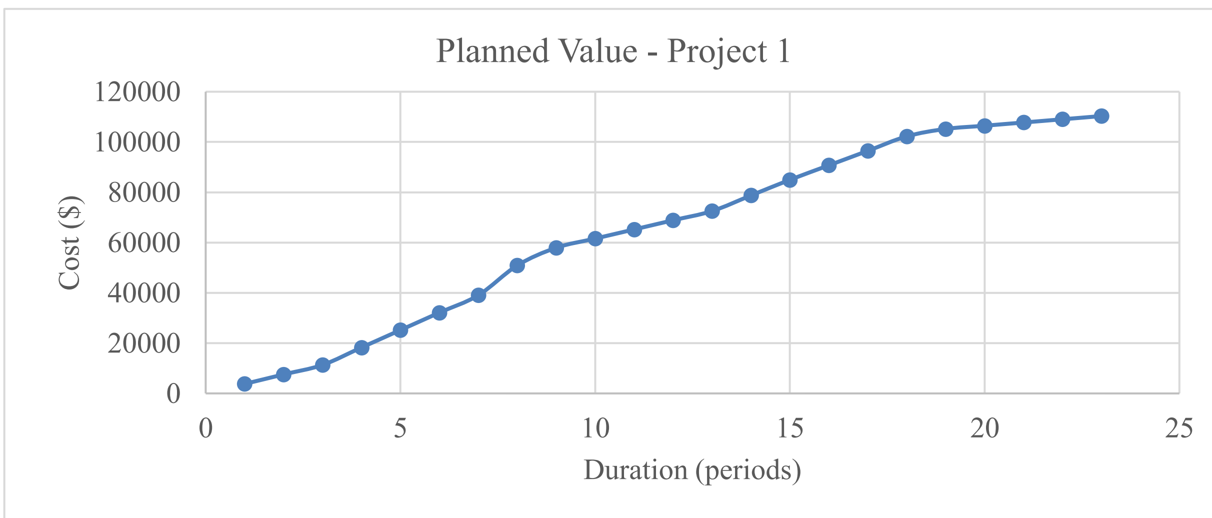
Willems, Laura L., and Mario Vanhoucke. 2015. “Classification of Articles and Journals on Project Control and Earned Value Management.” *International Journal of Project Management* 33 (7): 1610–34. https://doi.org/10.1016/j.ijproman.2015.06.003.

Wirahadikusumah, Reini, Dulcy M. Abraham, and Judy Castello. 1999. “Markov Decision Process for Sewer Rehabilitation.” *Engineering, Construction and Architectural Management* 6 (4): 358–70. https://doi.org/10.1108/eb021124.

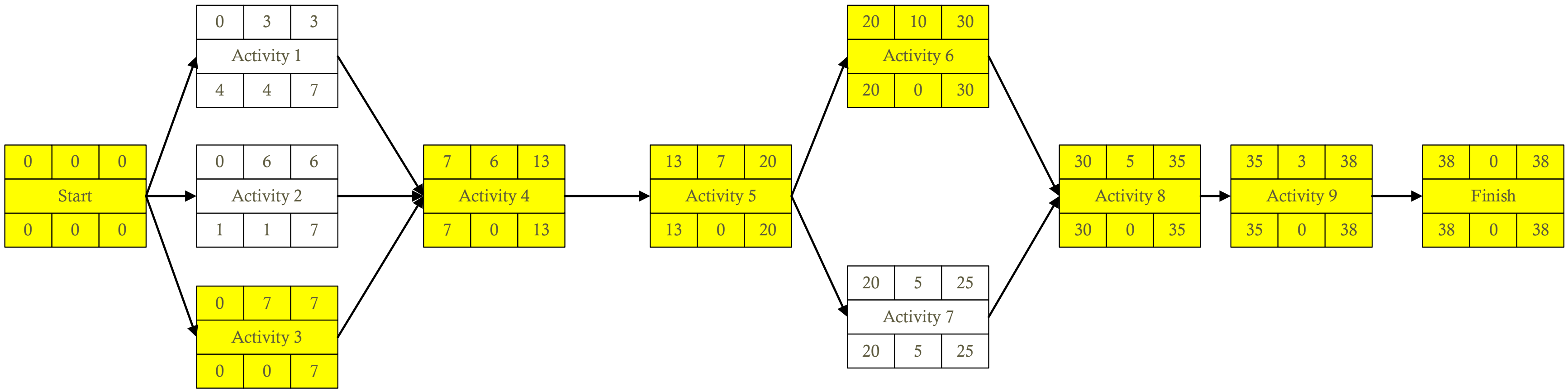
APPENDIX A: SAMPLE PROJECTS

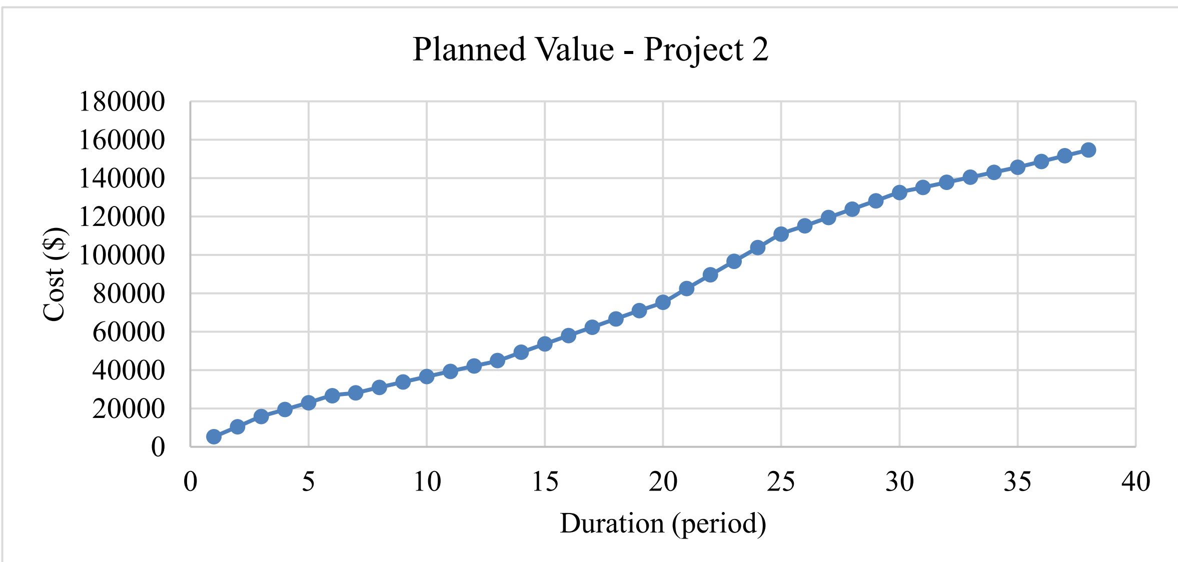
Sample Project 1



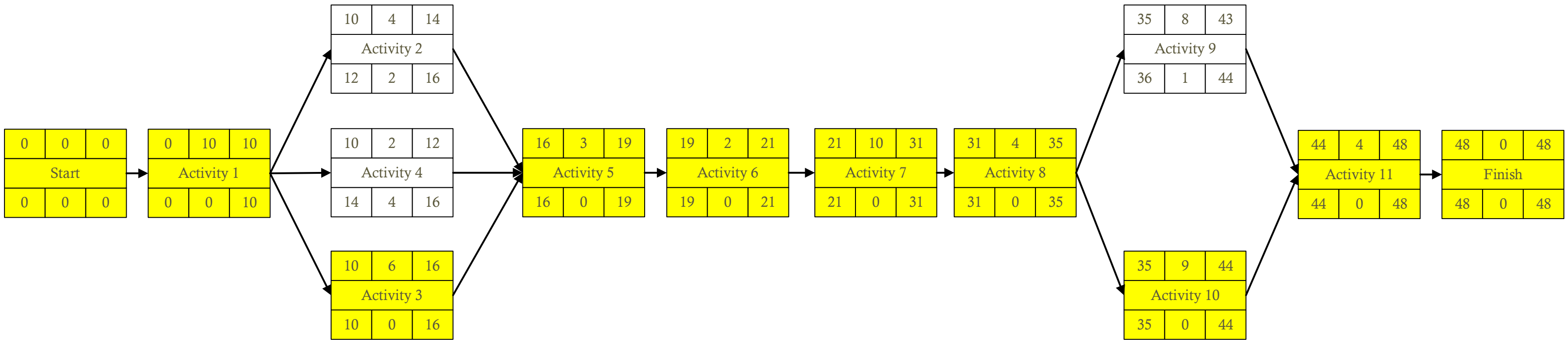


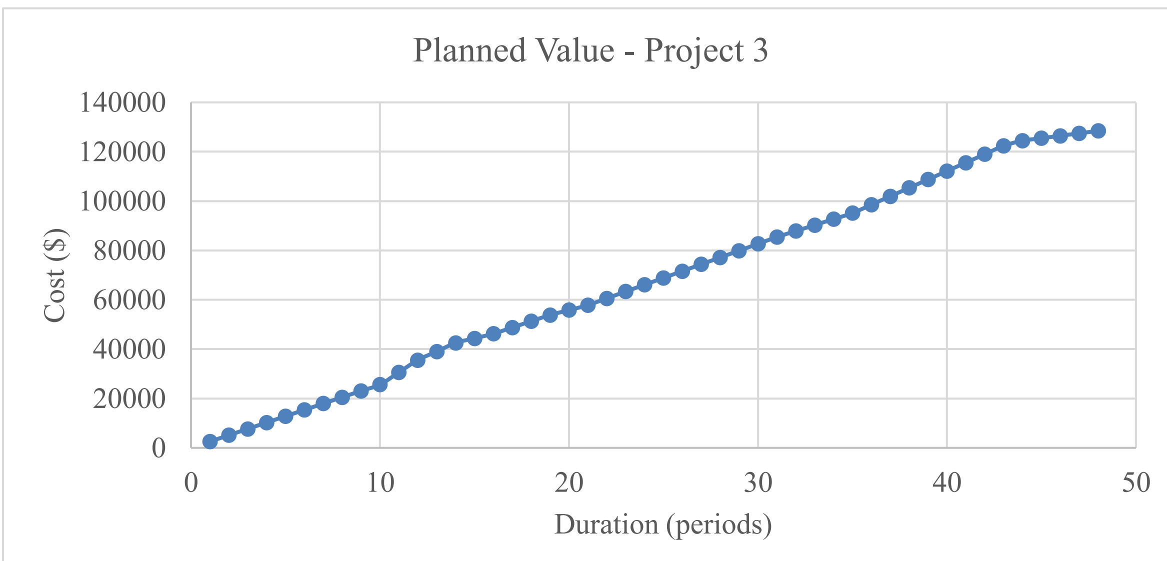
Sample Project 2



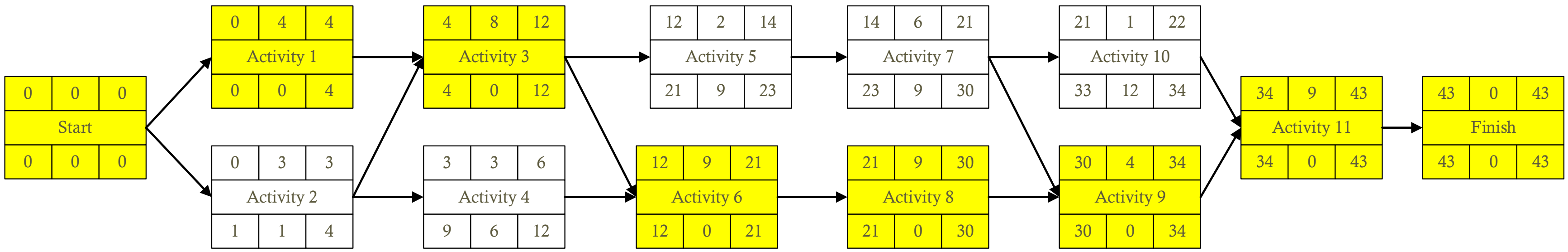


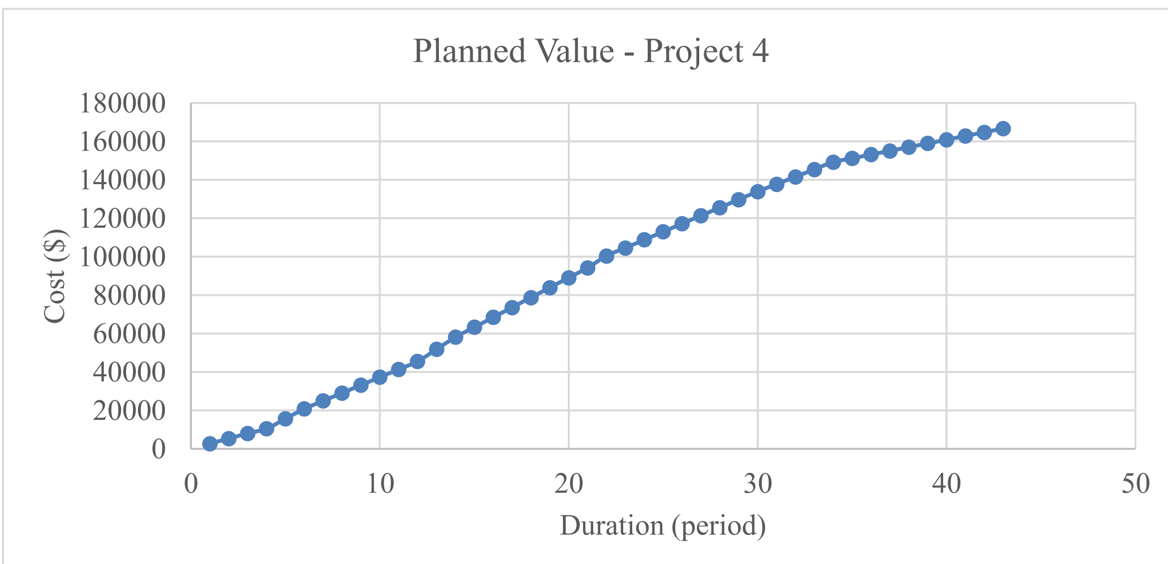
Sample Project 3



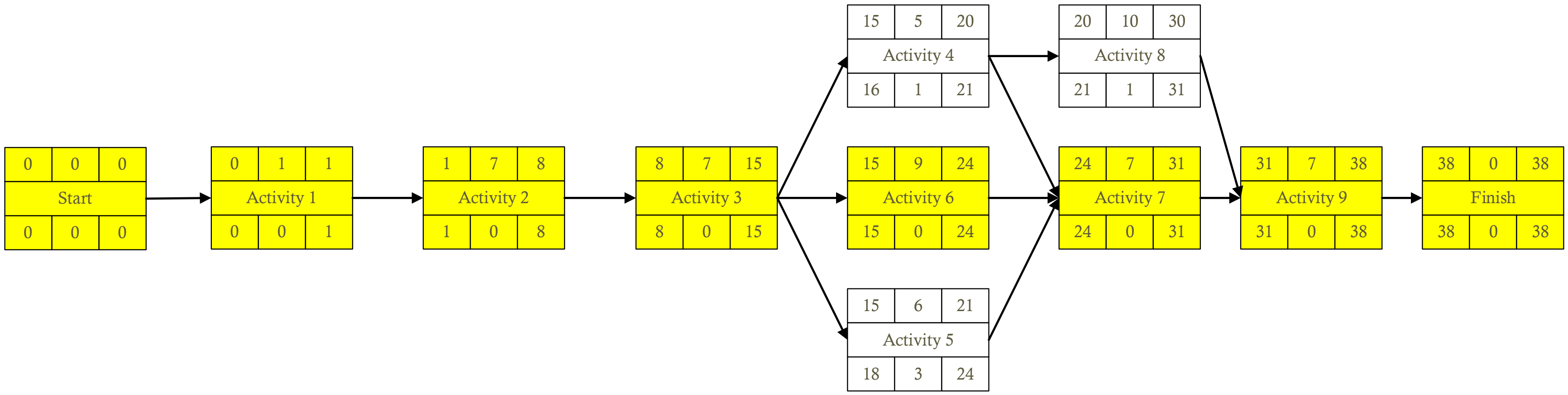


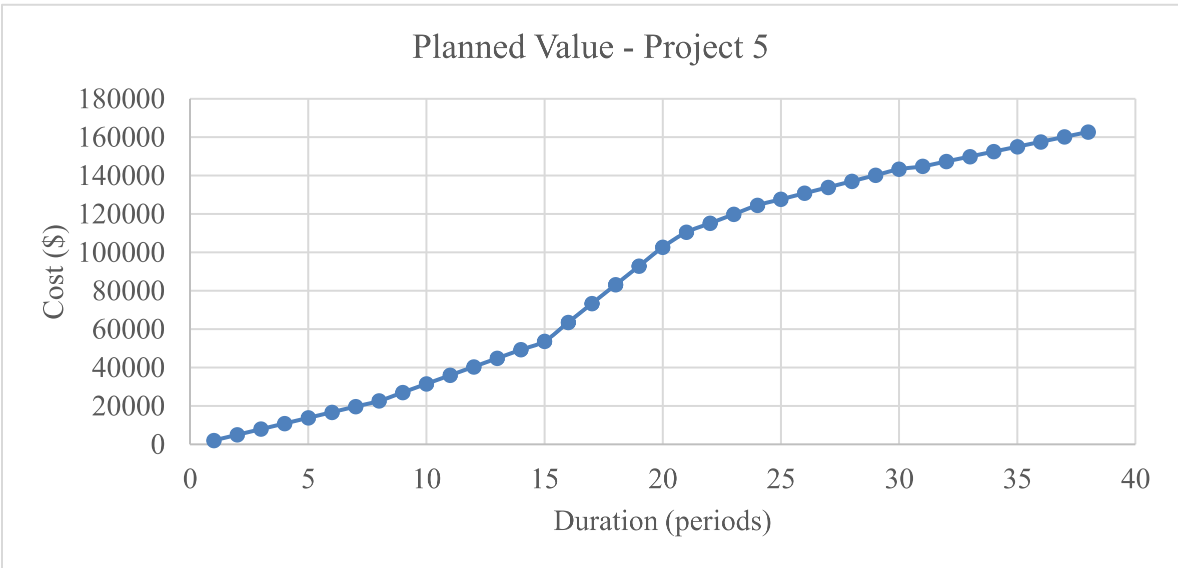
Sample Project 4





Sample Project 5





Sample Project 6