



**Politecnico  
di Torino**

**ScuDo**

Scuola di Dottorato ~ Doctoral School  
WHAT YOU ARE, TAKES YOU FAR

Doctoral Dissertation

Doctoral Program in Management, Production, and Design (36<sup>th</sup> cycle)

# **Data-driven Project Management**

## **Leveraging Analytics to Improve Monitoring and Control**

By

**Filippo Maria Ottaviani**

\*\*\*\*\*

**Supervisor(s):**

Prof. Alberto De Marco

**Doctoral Examination Committee:**

Prof. Hodayoun Khamooshi, Referee, GW School of Business

Prof. Mauro Mancini, Referee, Politecnico di Milano

Prof. Timur Narbaev, Politecnico di Torino

Prof. Pablo Ballesteros-Pérez, Universitat Politècnica de València

Prof. Richard Maltzman, Boston University Metropolitan College

Politecnico di Torino

2024

# Chapter 1

## Introduction

A fundamental challenge in project monitoring and control is the significant influence of human factors in decision-making processes. Control actions are developed based on the perceived risk and risk appetite. Estimates at completion depend on the forecasting method adopted and assumptions about future events. Assessing current performance depends on the assessment method and the significant factors for explaining and forecasting the project cost and schedule behaviors.

Both academic and professional worlds have recognized the effectiveness of the Earned Value Management methodology for project monitoring and control (Santos et al., 2023).

The EVM methodology is based on three key metrics: Earned Value ( $EV$ ), Planned Value ( $PV$ ), and Actual Cost ( $AC$ ). While  $PV$  denotes the budgeted cost of work scheduled, determined per the PMB,  $EV$  and  $AC$  denote the budgeted and actual cost of work performed, respectively, recorded throughout project execution.

Performance indexes include the Cost Performance Index ( $CPI$ ) and the EVM Schedule Performance Index ( $SPI^{EVM}$ ). The former is determined by the ratio of  $EV$  to  $AC$ , as per Eq. 1.1.

$$CPI(t) = \frac{EV(t)}{AC(t)} \quad (1.1)$$

The latter is determined by the ratio of  $EV$  to  $PV$ , as per Eq. 1.2.

$$SPI^{EVM}(t) = \frac{EV(t)}{PV(t)} \quad (1.2)$$

With regard to forecasting methods, EVM employs two different approaches to forecast the project Cost Estimate at Completion ( $cEAC$ ) and Time Estimate at Completion ( $tEAC^{EVM}$ ). The former is calculated as the sum of  $AC$  and the Cost Estimate to Complete ( $cETC$ ), which is defined as the ratio of the budgeted cost of remaining work ( $BAC - EV$ ) to the Cost Performance

Factor ( $cPF$ ), as per Eq. 1.3.

$$\begin{aligned} cEAC(t) &= AC(t) + cETC(t) \\ &= AC(t) + \frac{BAC - EV(t)}{cPF(t)} \end{aligned} \quad (1.3)$$

The latter is determined by the ratio of PD to the Schedule Performance Factor ( $sPF$ ), as per Eq. 1.4.

$$tEAC^{EVM}(t) = \frac{PD}{sPF(t)} \quad (1.4)$$

Literature has provided consistent results regarding the best-performing  $cPF$  and  $sPF$ . Concerning cost forecasting, studies have confirmed that  $CPI$  provides adequate results as  $cPF$  (Ballesteros-Pérez et al., 2019; Batselier and Vanhoucke, 2015b; Kim, 2019). Regarding duration forecasting,  $SPI$  has faced criticism due to relying on cost metrics (i.e.,  $EV$  and  $PV$ ) for quantifying schedule delay (Borges and Mário, 2017). This assumption can lead to unreliable forecasts when the relationship between cost and schedule is not linear (Khamooshi and Golafshani, 2014; Warburton et al., 2017), and as the project approaches completion, since the index converges to one, hinting at no delay (Chang et al., 2020; De Marco et al., 2017).

To address the issues of  $SPI$ , Lipke (2003) developed the Earned Schedule (ES) methodology. ES relies on the homonym metric (Earned Schedule,  $ES$ ), representing the status date  $EV$  should have been reached according to PMB, assuming linear progress between consecutive tracking periods, as per Eq. 1.5.

$$\begin{aligned} ES(t) &= z + \frac{EV(t) - PV(z)}{PV(z+1) - PV(z)} \\ PV(z) &\leq EV(t) \leq PV(z+1) \end{aligned} \quad (1.5)$$

As EVM, ES comes with the Schedule Variance ( $SV^{ES}$ ) and ES Schedule Performance Index ( $SPI^{ES}$ ). The latter is determined by the ratio of  $ES$  to  $t$ , as per Eq. 1.6.

$$SPI^{ES}(t) = \frac{ES(t)}{t} \quad (1.6)$$

The same considerations for EVM variance indicators and performance indexes apply.

Despite their widespread use among practitioners, EVM and ES have several limitations.

- Rely on the correct breakdown of the scope of work defining the PMB (National Defense Industrial Association, 2021).
- Overlook performance trends (Du et al., 2016) and the possible impact of control actions on them (Narbaev and De Marco, 2011; Willems and Vanhoucke, 2015).
- Do not yield reliable forecasts during early project stages because of no adequate data for reliable performance assessment (Barrientos-Orellana et al., 2022).
- Do not consider any correlation between cost and schedule performances (Khamooshi and Golafshani, 2014), whereas both factors should be analyzed together (Karimi et al., 2018; Khamooshi et al., 2021).

- Do not account for the current phase (Ottaviani and De Marco, 2021), whereas project must have reached stability to obtain reliable forecasts (Ballesteros-Pérez et al., 2019; Barrientos-Orellana et al., 2022, 2023; De Koning and Vanhoucke, 2016; Kim et al., 2016).
- Do not account for project network features, such as subcritical paths (Narbaev and De Marco, 2011; Vanhoucke, 2010), level of parallelism between paths (Gálvez et al., 2015, 2017), and other relationships between activities (Elshaer, 2013; Vanhoucke, 2012).

Ignoring the EVM and ES issues compromises project performance assessment, leading to incorrect forecasts and, in turn, to sub-optimal response actions.

This thesis aims to improve project monitoring and control processes by identifying, analyzing, modeling, and exploiting qualitative factors influencing them. This will be done through three related yet independent studies addressing the project at the descriptive, predictive, and prescriptive analytics levels. All three studies share the assumption that only EVM and ES metrics will be used.

# Chapter 2

## Work Rate Indicators for Project Performance Analysis and Forecasting

### 2.1 Introduction

This study aims to verify that work rate factors are significant to project performance analysis and forecasting. To accomplish this, the study quantifies work rate factors through three indicators describing the project relative phase, work rate, and work rate acceleration. The study verifies the significance of the indicators using eight artificial neural network (ANN) models, each incorporating a different combination of the proposed indicators. This approach allows for determining whether the indicators influence the ANNs' forecasting performance and, if so, in what sense. In addition, the study benchmarks the performance of the ANNs against the EVM and ES methods under the accuracy, precision, timeliness, and goodness of fit criteria.

### 2.2 Research Methodology

This section describes the framework followed to develop and verify the significance of the work rate indicators to project performance forecasting.

#### 2.2.1 Data Collection

Data collection involves selecting completed projects and retrieving their PMB — i.e.,  $t$ , and  $PV$  — and monitoring data — i.e.,  $EV(t)$ ,  $AC(t)$ , and  $ES(t)$ .

#### 2.2.2 Feature Scaling

Feature scaling involves dividing time features (i.e.,  $t$ , and  $ES$ ) by PD and cost features (i.e.,  $PV$ ,  $EV$ , and  $AC$ ) by BAC.

### 2.2.3 Feature Evaluation

Feature evaluation involves calculating the work rate indicators to provide as input, along with the scaled features, for regression analysis.

The first indicator ( $F_1$ ) quantifies the project “relative phase” by comparing the current  $WP$  with the flex in  $WS$ . The indicator is calculated as per Eq. 2.1.

$$F_1(t) = WS^* - WP(t) = WS[\operatorname{argmax} WS'(t)] - WP(t) \quad (2.1)$$

The  $F_1$  indicator is predicated on the assumption that a project’s most work-intensive phase occurs when the work rate reaches its peak (Liao et al., 2011; Putnam, 1978; Warburton, 1983). Therefore,  $F_1$  is most significant in projects with a clear and distinct work-intensive phase, indicated by an S-shaped cumulative work profile and amplitude in work rate oscillations (S-shape curvature). The sign of  $F_1$  indicates whether the project has already passed through the peak phase ( $F_1 < 0$ ) or not ( $F_1 > 0$ ), while its absolute value ( $|F_1|$ ) measures the “distance” between the peak phase and the current phase.

The second indicator ( $F_2$ ) quantifies the project’s “relative work rate” by comparing  $WP'$  with the peak in  $WS'$ . The indicator is calculated as per Eq. 2.2.

$$F_2(t) = \max \{WS'(t) : t = 1 \dots PD\} - WP'(t) \quad (2.2)$$

The indicator estimates the effort required by project resources (i.e., pressure put on the resources) compared to the planned maximum. The sign of  $F_2$  indicates whether the current work rate is higher ( $F_2 < 0$ ) or lower ( $F_2 > 0$ ) than the planned maximum work rate. The absolute value ( $|F_2|$ ) represents the difference between the actual and planned maximum work rates.

No specific range of values for  $F_2$  is considered optimal, depending on the project’s unique characteristics and requirements. However, a positive value of  $F_2$  indicates that the project is progressing according to plan and is not placing undue pressure on project resources. Conversely, a negative value of  $F_2$  suggests that the project is experiencing a higher work rate than planned, which could increase the risk of delays, cost overruns, and resource burnout.

The third indicator ( $F_3$ ) is similar to  $F_2$  but concerns work rate acceleration, as it compares the current  $WP''$  to the peak in  $WS''(t)$ . The indicator is calculated as per Eq. 2.3.

$$F_3(t) = \max \{WS''(t) : t = 1 \dots PD\} - WP''(t) \quad (2.3)$$

The indicator is a proxy for how frequently and significantly the work rate changes. The sign of  $F_3$  indicates whether the current work rate acceleration exceeds the planned maximum ( $F_3 < 0$ ) or not ( $F_3 > 0$ ). The absolute value ( $|F_3|$ ) represents the difference in work rate acceleration between the actual and planned maximum values.

Both  $F_2$  and  $F_3$  indicators are determined through the difference operation rather than the ratio, preserving the indicators’ sign, which conveys additional information for regression model development. Furthermore, the difference between the two values does not constrain the range

of values the indicators could assume, whereas the ratio would do. The indicator equals zero when the work rate or work rate acceleration equals its respective maximum planned value. If either indicator is negative, it suggests that the project entropy is high, increasing the likelihood of cost or schedule overruns (Vanhoucke, 2019).

Table 2.1 summarizes the features provided as input for regression analysis, including their ID, formula, and name.

Table 2.1 Regression analysis input features

ID	Feature	Formula	Name
1	$t_s$	$t/PD$	Time
2	$WS$	$PV/BAC$	Work Scheduled
3	$WP$	$EV/BAC$	Work Performed
4	$AC_s$	$AC/BAC$	Actual Cost
5	$ES_s$	$ES/PD$	ES Metric
6	$F_1$	$WS^* - WP$	Relative Phase
7	$F_2$	$\max WS' - WP'$	Relative Work Rate
8	$F_3$	$\max WS'' - WP''$	Relative Work Rate Acceleration

#### 2.2.4 Regression Analysis

Regression analysis is a statistical technique for estimating the true relationship ( $f$ ) between a target variable ( $y$ ) and a set of independent features ( $\mathbf{X}$ ) through a fitted model ( $\hat{f}$ ). The fitted model serves both inferential and predictive purposes: inference refers to the degree of understanding of the association between the variables being analyzed, while prediction refers to forecasting the target variable.

This study utilizes ANNs to evaluate  $\hat{f}$ . An ANN comprises interconnected neurons organized into input ( $I$ ), hidden ( $H$ ), and output ( $O$ ) layers. The input layer corresponds to  $\mathbf{X}$ . The hidden layers apply a nonlinear function to the values they receive as input. Finally, the output layer receives the values from the last hidden layer and generates the forecasts ( $\hat{y}$ ).

Verifying the significance of the work rate indicators requires comparing the performance of the ANNs with and without them. Table 2.2 provides the eight ANNs tested in the study. ANN 0 contains only EVM and ES features (with ID from 1 to 5 as per Table 2.1). ANNs 1, 2, and 3 implement  $F_1$ ,  $F_2$ , and  $F_3$  into ANN 0, respectively. ANNs 12, 13, and 23 implement  $F_1$  and  $F_2$ ,  $F_1$  and  $F_3$ , and  $F_2$  and  $F_3$ , respectively. Lastly, ANN 123 implements all three proposed indicators.

Table 2.2 ANN models architecture summary

ANN	Additional Variables	Input Neurons	Hidden Neurons
0		5	3
1	$F_1$	6	4
2	$F_2$	6	4
3	$F_3$	6	4
12	$F_1, F_2$	7	5
13	$F_1, F_3$	7	5
23	$F_2, F_3$	7	5
123	$F_1, F_2, F_3$	8	5

### 2.2.5 EVM, ES, and ANN Forecasts

Forecasts involve using the EVM, ES, and ANN  $cEAC$  and  $tEAC$  models to generate the forecasts of the target variables.

Concerning cost, EVM forecasts are determined as per Eq. 2.4.

$$\hat{y}_i = \frac{cEAC_i}{BAC_i} = AC_{si} + \frac{1 - WP_i}{CPI_i} \quad (2.4)$$

Instead, ANN forecasts are determined through the fitted  $cEAC$  ANN models.

Concerning duration, EVM forecasts are determined as per Eq. 2.5.

$$\hat{y}_i = \frac{tEAC_i^{EVM}}{PD_i} = \frac{1}{SPI_i^{EVM}} \quad (2.5)$$

ES forecasts are determined through Eq. 2.6.

$$\hat{y}_i = \frac{tEAC_i^{ES}}{PD_i} = t_{si} + \frac{1 - ES_{si}}{SPI_i^{ES}} \quad (2.6)$$

ANN forecasts are determined through the fitted  $tEAC$  ANN models.

## 2.3 Results

### 2.3.1 Data Collection

The project monitoring dataset used consists of 80 construction projects data, selected from the OR&S database (Batselier and Vanhoucke, 2015a; Vanhoucke et al., 2016), publicly available at [projectmanagement.ugent.be/research/data/realdata](http://projectmanagement.ugent.be/research/data/realdata).



Table 2.3 *cEAC* models performance results

Model	<i>AE</i>				<i>SE</i>				$R^2_{adj}$
	<i>MAE</i>	<i>SAE</i>	<i>wMAE</i>	<i>wSAE</i>	<i>RMSE</i>	<i>SSE</i>	<i>wRMSE</i>	<i>wSSE</i>	
EVM	0.1070	0.1796	0.0310	0.0788	0.2090	0.1450	0.1168	0.0601	-1.70
ANN 0	0.0511	0.0634	0.0212	0.0326	0.0814	0.0176	0.0494	0.0074	0.59
ANN 1	0.0414	0.0533	0.0155	0.0223	0.0675	0.0140	0.0377	0.0050	0.72
ANN 2	0.0466	0.0553	0.0181	0.0262	0.0723	0.0152	0.0417	0.0065	0.68
ANN 3	0.0484	0.0556	0.0191	0.0274	0.0737	0.0148	0.0431	0.0066	0.66
ANN 12	0.0398	0.0512	0.0156	0.0213	0.0649	0.0142	0.0367	0.0044	0.74
ANN 13	0.0404	0.0533	0.0156	0.0223	0.0669	0.0146	0.0375	0.0049	0.72
ANN 23	0.0498	0.0589	0.0185	0.0285	0.0771	0.0163	0.0440	0.0071	0.63
ANN 123	0.0392	0.0514	0.0155	0.0218	0.0647	0.0138	0.0368	0.0043	0.74

Table 2.4 *tEAC* models performance results

Model	<i>AE</i>				<i>SE</i>				$R^2_{adj}$
	<i>MAE</i>	<i>SAE</i>	<i>wMAE</i>	<i>wSAE</i>	<i>RMSE</i>	<i>SSE</i>	<i>wRMSE</i>	<i>wSSE</i>	
EVM	0.3259	1.3243	0.0845	0.1081	1.3638	44.5161	0.2480	0.3637	-61.28
ES	0.2970	1.5866	0.0735	0.1340	1.6142	69.0170	0.2708	0.5937	-86.25
ANN 0	0.0949	0.0883	0.0410	0.0495	0.1296	0.0308	0.0820	0.0152	0.44
ANN 1	0.0881	0.0817	0.0369	0.0460	0.1202	0.0263	0.0752	0.0126	0.52
ANN 2	0.0908	0.0846	0.0390	0.0472	0.1241	0.0292	0.0784	0.0146	0.49
ANN 3	0.0901	0.0864	0.0387	0.0485	0.1248	0.0294	0.0790	0.0156	0.48
ANN 12	0.0820	0.0830	0.0353	0.0449	0.1167	0.0263	0.0730	0.0123	0.54
ANN 13	0.0866	0.0839	0.0377	0.0474	0.1206	0.0277	0.0765	0.0134	0.51
ANN 23	0.0908	0.0856	0.0389	0.0481	0.1248	0.0289	0.0788	0.0143	0.48
ANN 123	0.0842	0.0835	0.0364	0.0446	0.1186	0.0283	0.0739	0.0129	0.53

## 2.4 Conclusions

The results confirmed that the indicators were significant and able to improve the ANN models. The maximum improvement was achieved when all three indicators were used as input. Furthermore, the study proved that even a simple ANN with MLP architecture could benefit from implementing the indicators, leading to more accurate, precise, and timely forecasts than the EVM and ES methodologies. This confirms ANNs' capability to analyze the relationship between project performance and work rate indicators, as well as the ability of work rate indicators to anticipate or assist in evaluating performance variations.

# Chapter 3

## Robust Machine Learning Pipeline for Project Performance Forecasting

### 3.1 Introduction

This chapter aims to improve project monitoring by providing solutions, structured in an ML pipeline, that address the problems of underfitting and overfitting in ML project performance forecasting models. The study tests the pipeline on a subset of the project monitoring dataset from Chapter 2 by evaluating 30 ML models and benchmarking their performance against that of EVM and ES.

### 3.2 Research Methodology

This section outlines the ML pipeline and the criteria for assessing the performance of ML models evaluated through it. The ML pipeline consists of seven phases: data collection, data preprocessing, feature engineering, model preselection, feature selection, model training and evaluation, and model selection. Each phase includes several steps that implement procedures to reduce underfitting and overfitting.

For clarity, the terms *procedure* and *technique* are used interchangeably; the same applies to *variable* and *feature*.

#### 3.2.1 Data Collection

Data collection involves assembling raw monitoring data from completed projects into a structured dataset.

Raw monitoring data include  $t$ , BAC, PD,  $PV(t)$ ,  $EV(t)$ ,  $AC(t)$ , and  $AD$ .

### 3.2.2 Data Preprocessing

Data preprocessing involves transforming raw monitoring data into a format suitable for subsequent analysis. This phase includes four steps: handling missing values, data cleaning, data transformation, and data balancing and augmentation.

#### 3.2.2.1 Handling Missing Values

Handling missing values involves identifying incomplete records and either removing or filling them in.

#### 3.2.2.2 Data Cleaning

Data cleaning involves identifying and removing inappropriate records from the dataset.

#### 3.2.2.3 Data Transformation

Data transformation involves normalizing, standardizing, or scaling monitoring data.

#### 3.2.2.4 Data Balancing and Augmentation

Data balancing involves equalizing the number of observations per project. Instead, data augmentation involves undersampling (i.e., reducing the number of records) or oversampling (i.e., increasing the number of records).

Data augmentation procedures include linear and nonlinear interpolation. The former assumes a constant rate between successive records, resulting in multiple straight lines connecting adjacent data points. The latter assumes the rate varies according to a theoretical model with a non-uniform growth rate, preserving the monotonicity and cumulative nature of  $t$ ,  $PV$ ,  $EV$ , and  $AC$ .

This pipeline implements balancing using synthetic records generated by linear interpolation at 5% increments, as per Eq. 3.1.

$$x [WP(t_s) = z] = \{x(t_s) : WP(t_s) = z\} \quad \forall z \in [0, .05, \dots, 1.00] \quad (3.1)$$

where  $x$  denotes the metric to interpolate, and  $z$  denotes the specific value of  $WP$  at  $t_s$ . The choice of 5% is arbitrary and allows the generation of 20 synthetic records per project.

### 3.2.3 Feature Engineering

Feature engineering involves combining project metrics to evaluate new features. These features will be used as input features, which serve as independent regression variables, and target features, representing the dependent variables to forecast. This step improves the ML inferential process by guiding the analysis of significant features for assessing project performance while reducing underfitting and overfitting.

### 3.2.3.1 Input Features

Input features comprise both project metrics and indicators. Project metrics include the synthetic values of  $t_s$ ,  $WS$ ,  $WP$ ,  $AC_s$ , and  $BAC_s$  (for cost forecasting) or  $PD_s$  (for duration forecasting). Instead, monitoring indicators consist of different combinations of project metrics.

The first set of indicators is based on the EVM methodology and includes both cost and schedule indicators.

The second set of indicators is based on the ES methodology.

### 3.2.3.2 Target Variable

Selecting the target variable depends on the forecasting target and the regression method used. The forecasting target can be cost or duration, and the regression method can be direct or indirect.

Direct regression (DR) involves setting the target variable to the regression target. In this approach, records within the same project have identical values for the target variable. As a result, analysis of the relationships between input features is limited, reducing model complexity. Therefore, DR prioritizes the reduction of underfitting over the reduction of overfitting.

Indirect regression (IR) involves setting the target variable to an intermediate variable and using it within a specific formula to calculate the regression target. In this approach, data sets from the same project have different values for the target variable, allowing for a more flexible analysis of the relationships between input features. As a result, IR prioritizes overfitting reduction over overfitting.

This pipeline implements both regression methods for both forecasting targets.

### 3.2.4 Feature Selection

Feature selection involves identifying a subset of the input features to build the model.

This pipeline employs the forward Sequential Feature Selection (fSFS) procedure with a tolerance-based stopping criterion.

### 3.2.5 Model Preselection

Model preselection involves identifying candidate models to be tested in forecasting project performance. This approach makes it possible to select, from the various models tested, the one that best fits the needs of the case.

This pipeline tests 30 models selected based on their extensive presence in the project performance forecasting literature and their open-source availability.

### 3.2.6 Model Training and Evaluation

Model training involves evaluating the coefficients of the parameters of the model built using the features selected through the fSFS procedure. Instead, model evaluation involves using the trained model to compute the forecasts and evaluate its performance.

This pipeline implements the Group  $k$ -fold CV technique in training the models to evaluate their performance, to reduce overfitting.

Within each iteration of the Group  $k$ -Fold CV procedure, the pipeline uses the Grid Search CV procedure to tune the model hyperparameters and balance underfitting and overfitting. The procedure tests all possible combinations of predefined values of the hyperparameters, identifying the combination that minimizes a regression scorer evaluated over the  $k$  folds.

### 3.2.7 Model Selection

Model selection involves choosing the best-performing ML model for cost and duration forecasting.

### 3.2.8 Results Comparison

The study compares the performance of the ML models with the EVM and ES ones under three criteria in two steps.

In cost forecasting, the EVM model consists of  $EVM(CPI)$ , based on Eq. 2.4. In duration forecasting, the EVM model consists of  $EVM(SPI^{EVM})$ , based on Eq. 2.5, and Instead, the ES model consists of  $ES(SPI^{ES})$ , based on Eq. 2.6.

The performance criteria are accuracy, precision, and timeliness.

Table 3.1 shows the  $MAE$  and  $RMSE$  values associated with the EVM, ES, DR, and IR models .

Table 3.1 EVM, ES, and best ML models overall scores

Target	Method	Model	$MAE$	$RMSE$
Cost	EVM	$EVM(CPI)$	0.0690	0.1508
	DR	SVR	0.0530	0.0976
	IR	Huber	0.0529	0.1059
Duration	EVM	$EVM(SPI^{EVM})$	0.2249	0.5079
	ES	$ES(SPI^{ES})$	0.2116	0.5453
	DR	Huber	0.1001	0.1521
	IR	SGD1cSVM	0.0990	0.1522

### **3.3 Conclusions**

The study tested the pipeline on 80 construction engineering projects. The results demonstrate that the ML models exhibit higher accuracy and precision than EVM and ES. Furthermore, some ML models show greater accuracy when developed by IR rather than DR.

# Chapter 4

## Risk Perception-Based Project Contingency Management Framework

### 4.1 Introduction

This study's objective is twofold. Firstly, it explains the relationship between short- and long-term cost overruns, how project managers perceive them, and how they can react. Secondly, it aims to model risk perception, provide the criteria for developing risk responses, and implement both within a contingency management framework (CMF) to minimize short- and long-term concerns for cost overruns throughout project execution while ensuring the complete depletion of the CR. The CMF applies the Monte Carlo (MC) method to a stochastic, discrete-event, finite-horizon, dynamic project simulation model for evaluating the combinations of initial CR level and response thresholds that optimize contingency spending. In the CMF, risk responses consist of using part of the CR to reduce the increase in the project's actual cost, thereby reducing both the perceived overrun and the cost variance at project completion.

### 4.2 Research Methodology

The study developed the CMF, more specifically, the project execution simulation model, following a simplified version of the methodologies of Law (2003) and Banks et al. (2009), consisting of the following steps:

- Problem formulation and system configurations;
- Model definition;
- Model translation;
- Pilot runs;
- Model validation;
- Output data analysis;
- Discussion of results.

## 4.2.1 Problem Formulation and System Configurations

### 4.2.1.1 Problem Formulation

The CMF is intended as a prescriptive tool for driving CM while minimizing project managers' exposure to risk perception. The CMF should determine, for each initial level of CR, the frequency and entity of risk responses.

### 4.2.1.2 System Configurations

Model verification depends on the mathematical properties of the project execution simulation model. On the other hand, model validation requires testing it in several projects, which is not feasible because of the infinite number of possible project configurations. Hence, for the purpose of validation, this study employs synthetic data representing extreme project configurations. Suppose the CMF works in such extreme configurations. In this case, the CMF can be applied in any real project that has a configuration that falls between the extreme ones. This study defines project configurations based on the schedule, cost deviation, and correlation of the tasks.

Task schedule refers to the tasks' start and end dates. Following the activity-based costing method (Lance Stephenson, 2015), the task schedule determines the cumulative cost curve, which represents the total project cost as a function of time.

A task's cost deviation is expressed using a probability density function (PDF) to account for potential risks Du et al. (2016). This function is derived by fitting a theoretical distribution to historical cost overrun data or by selecting an estimation method and gathering uncertainty factors for a sensitivity analysis. The resulting PDF values can be adjusted to accommodate additional risks that would substantially alter the task cost or incorporate subjective assumptions.

Ideally, each task should have its cost deviation PDF. However, for verification and validation purposes, this study assumed all tasks' cost deviation PDFs to be the same (in relative terms) (Barraza and Bueno, 2007; Eldosouky et al., 2014). Specifically, this study adopted the PERT distribution as PDF, i.e., a beta distribution extended to the domain [min, max] (Vose, 2008).

Let  $x$  denote the task cost deviation. Then, Eq. 4.1 provides the PDF of the PERT distribution:

$$f(x) = \frac{(x - \min)^{\alpha-1} (\max - x)^{\beta-1}}{\text{Beta}(\alpha, \beta) (\max - \min)^{\alpha+\beta-1}}, \quad (4.1)$$

where min is the  $x$  lower bound, max is the  $x$  upper bound,  $\alpha$  and  $\beta$  are the shape parameters, and Beta is the beta function described by Eq. 4.2,

$$\text{Beta}(\alpha, \beta) = \int_0^1 u^{\alpha-1} (1-u)^{\beta-1} du. \quad (4.2)$$

Eq. 4.3 provides the cumulative density function (CDF) of the PERT distribution:

$$F(x) = \frac{\text{Beta}_z(\alpha, \beta)}{\text{Beta}(\alpha, \beta)}, \quad (4.3)$$



where  $\text{Beta}_z$  is the incomplete beta function and  $z$  is evaluated as per Eq. 4.4,

$$z = \frac{x - \min}{\max - \min}. \quad (4.4)$$

A task's cost deviation correlation affects its cost deviation PDF. This study involves two correlation scenarios, namely, A and B. In Scenario A, all tasks are assumed to not be correlated. In scenario B, all tasks executed in the same time frame show the same relative cost deviation.

### 4.2.2 Model Definition

Let  $N$  denote the total number of tasks, each contributing equally to the project's progress, and let  $i$  indicate the  $i$ th task. Then, Eq. 4.5 holds.

$$\begin{aligned} dWP_i &= \frac{100\%}{N} \\ &= \frac{1}{N} \end{aligned} \quad (4.5)$$

Following EVM, progress is determined by assuming  $dEV(t) = dWP(t) \cdot \text{BAC}$ . Then, Eq. 4.6 holds.

$$dEV_i = \frac{\text{BAC}}{N} \quad (4.6)$$

Let  $n(t)$  denote the number of tasks completed at time  $t$ , determined by the profiles displayed in Fig. ???. Then,  $dEV$  is determined using Eq. 4.7.

$$\begin{aligned} dEV(t) &= n(t) \cdot dEV_i \\ &= n(t) \cdot \frac{\text{BAC}}{N} \end{aligned} \quad (4.7)$$

In contrast,  $dAC$  depends on both the task correlation and  $x_i$ , which can be randomized by applying the inverse of Eq. 4.3 to a random value generated through the uniform distribution, as in Eq. 4.8.

$$x_i(t) \sim F^{-1}(t) \quad \forall i \in [0 \dots n(t)] \quad (4.8)$$

In Scenario A, all project tasks are assumed to be independent; hence,  $dAC$  is randomized as per Eq. 4.9.

$$dAC(t) = \sum_{i=1}^{n(t)} \frac{\text{BAC}}{N} \cdot x_i(t) \quad (4.9)$$

In Scenario B, all tasks executed in the same  $t$  are assumed to show the same relative cost deviation. Therefore,  $dAC$  is randomized per Eq. 4.10.

$$dAC(t) = dEV(t) \cdot x(t) \quad (4.10)$$

The project analytical model is fit within Algorithm 1, which evaluates all combinations of  $C_0$ ,  $Th_{ST}$ , and  $Th_{LT}$  optimizing CM. The algorithm is defined as follows. The parameter  $C_0$  denotes the initial level of CR, i.e.,  $C(0) = C_0$ . The parameters  $\delta^{C_0}$ ,  $\delta^{Th_{ST}}$ , and  $\delta^{Th_{LT}}$  denote the increment in  $C_0$ ,  $Th_{ST}$ , and  $Th_{LT}$ , respectively. An upper bound corresponds each parameter ( $\max^{C_0}$  for  $C_0$ ,  $\max^{Th_{ST}}$  for  $Th_{ST}$ , and  $\max^{Th_{LT}}$  for  $Th_{LT}$ ). Iterations over  $C_0$  are repeated until it reaches  $\max^{C_0}$  or until  $Th_{ST} = Th_{LT} = 0$ , triggering the first stop criterion ( $stop^{C_0} \leftarrow 1$ ). Next, iterations over  $Th_{ST}$  are repeated until it reaches  $\max^{Th_{ST}}$ . Finally, iterations over  $Th_{LT}$  are repeated until it reaches  $\max^{Th_{LT}}$  or until the mean value of the residual  $C$  amount over the  $S$  simulations ( $\bar{C}$ ) is reduced to almost zero ( $\epsilon^l \leq \bar{C} \leq \epsilon^h$ ), triggering the second stop criterion ( $stop^{Th_{LT}} \leftarrow 1$ ).

**Algorithm 1:** Contingency management framework optimization algorithm**Data:**  $n, \mu, \sigma$ **Parameters:** S, BAC, PD,  $\max^{C_0}, \delta^{C_0}, \max^{Th_{ST}}, \delta^{Th_{ST}}, \max^{Th_{LT}}, \delta^{Th_{LT}}$ **Result:** Determine  $(C_0, Th_{ST}, Th_{LT})$  combinations that minimize  $\overline{CC}$  $C_0, stop^{C_0} \leftarrow 0, 0$ **while**  $C_0 \leq \max^{C_0}$  **and**  $stop^{C_0} = 0$  **do** $Th_{ST}, stop^{Th_{ST}} \leftarrow 0, 0$ **while**  $Th_{ST} \leq \max^{Th_{ST}}$  **and**  $stop^{Th_{ST}} = 0$  **do** $Th_{LT}, stop^{Th_{LT}} \leftarrow 0, 0$  $sol \leftarrow (C_0, Th_{ST}, Th_{LT})$  $CC_{sol} \leftarrow \infty$ **while**  $Th_{LT} \leq \max^{Th_{LT}}$  **and**  $stop^{Th_{LT}} = 0$  **do****for**  $s \leftarrow 1$  **to** S **do** $t \leftarrow 0$ **while**  $t < PD$  **do** $t \leftarrow t + 1$  $C(s, t) \leftarrow C(t - 1)$  $dEV(s, t) \leftarrow n(s, t) \cdot \frac{BAC}{N}$  $EV(s, t) \leftarrow EV(t - 1) + dEV(s, t)$  $dAC(s, t) \leftarrow \begin{cases} \sum_{i=1}^{n(s,t)} \frac{BAC}{N} \cdot x_i(s, t) & \text{Scenario = A} \\ dEV(s, t) \cdot x(s, t) & \text{Scenario = B} \end{cases}$  $AC(s, t) \leftarrow AC(t - 1) + dAC(s, t)$  $CC_{ST}(s, t) = \max\{0, dAC(s, t) - dEV(s, t) \cdot (1 + Th_{ST})\}$  $RC(s, t) \leftarrow \min\{C(s, t), CC_{ST}(s, t)\}$  $AC(s, t) \leftarrow AC(s, t) - RC_{ST}(s, t)$  $C(s, t) \leftarrow C(s, t) - RC_{ST}(s, t)$  $CC_{ST}(s, t) \leftarrow CC_{ST}(s, t) - RC_{ST}(s, t)$  $CPI(s, t) \leftarrow EV(s, t) / AC(s, t)$  $EAC(s, t) \leftarrow BAC / CPI(s, t)$  $CC_{LT}(s, t) = \max\{0, EAC(s, t) - BAC - C(s, t) \cdot (1 + Th_{LT})\}$  $RC_{LT}(s, t) \leftarrow$  $\min\left\{C(s, t), \max\left\{0, \frac{BAC[EV(s, t) - AC(s, t)] + C(s, t) \cdot EV(s, t) \cdot (1 + Th_{LT})}{EV(s, t) \cdot (1 + Th_{LT}) - BAC}\right\}\right\}$  $AC(s, t) \leftarrow AC(s, t) - RC_{LT}(s, t)$  $C(s, t) \leftarrow C(s, t) - RC_{LT}(s, t)$  $CPI(s, t) \leftarrow EV(s, t) / AC(s, t)$  $EAC(s, t) \leftarrow BAC / CPI(s, t)$  $CC_{LT}(s, t) = \max\{0, EAC(s, t) - BAC - C(s, t) \cdot (1 + Th_{LT})\}$  $CC(s, t) \leftarrow CC_{ST}(s, t) + CC_{LT}(s, t)$  $\overline{CC}(s) \leftarrow \frac{1}{PD} \sum_{t=0}^{PD} CC(s, t)$  $\overline{C} \leftarrow \frac{1}{S} \sum_{s=1}^S C(s, PD)$  $\overline{\overline{CC}} \leftarrow \frac{1}{S} \sum_{s=1}^S \overline{CC}(s)$ **if**  $0 \leq \overline{C} \leq \epsilon^h$  **and**  $\overline{\overline{CC}} \leq CC_{sol}$  **then** $stop^{Th_{LT}} \leftarrow 1$  $CC_{sol} \leftarrow \overline{\overline{CC}}$  $sol \leftarrow (C_0, Th_{ST}, Th_{LT})$ **else** $Th_{LT} \leftarrow Th_{LT} + \delta^{Th_{LT}}$  $Th_{ST} \leftarrow Th_{ST} + \delta^{Th_{ST}}$ **if**  $Th_{ST} = Th_{LT} = 0$  **then** $stop^{C_0} \leftarrow 1$ **else** $C_0 \leftarrow C_0 + \delta^{C_0}$

### 4.2.3 Pilot Runs

The algorithm was programmed from scratch in the Julia 1.8.1 programming language. Pilot runs (and later validation runs) were performed on an Intel(R) Core(TM) i7-10750H CPU; the run time was negligible.

The simulation model can be considered verified if it satisfies the central limit theorem (CLT) (Everitt and Skrondal, 2010). According to the CLT, the PDF of the project's total cost deviation should conform to a normal distribution to which mean and variance correspond to Eq. 4.11 and 4.12, respectively.

$$\begin{aligned}
 \mu &= \mu_{AC(PD)} \\
 &= N \cdot \mu_x \cdot \frac{BAC}{N} \\
 &= \mu_x \cdot BAC \\
 &= \mu_x
 \end{aligned} \tag{4.11}$$

$$\begin{aligned}
 \sigma^2 &= \sigma_{AC(PD)}^2 \\
 &= N \cdot \sigma_x^2 \cdot \frac{BAC^2}{N^2} \\
 &= \sigma_x^2 \cdot \frac{BAC^2}{N} \\
 &= \frac{\sigma_x^2}{N}
 \end{aligned} \tag{4.12}$$

The CLT holds only if the random variables are iid (i.e., independent and identically distributed). In the simulation model, the random variables correspond to the tasks' cost deviation PDFs and are iid only in the A configurations. However, if the only difference between Scenarios A and B is in the randomization of  $dAC$  and the simulation model verifies the CLT in the A scenarios, then the B configurations should also be verified.

### 4.2.4 Model Validation

Prior to conducting validation runs, the CR was bounded below zero ( $\min^{C_0} = 0$ ) and above by 0.8 ( $\max^{C_0} = 0.8$ ). Following the pilot runs, the CLT could be applied to narrow the range of  $C_0$  further to  $\mu - 3\sigma - BAC \leq C_0 \leq \mu + 3\sigma - BAC$ . This range is derived from the observation that a six-sigma interval covers approximately 99.7% of the possible outcomes. Based on this refined range of  $C_0$ , it can be predicted that the model may not provide an optimal solution for values of  $C_0$  below  $\mu - 3\sigma - BAC$ . Conversely, the CMF ensures that risks will always be mitigated for values of  $C_0$  above  $\mu + 3\sigma - BAC$ .

For the validation runs, we set the following:

- $S = 6.4 \times 10^3$ ;
- $\delta^{Th_{ST}} = \delta^{Th_{LT}} = 0.05$ ;
- $\epsilon^h = 1 \times 10^{-2}$ .

### 4.3 Conclusions

The CMF determines three intervals of initial CR based on the probability density function (PDF) of task cost deviation and their degree of correlation. The optimization algorithm then determines the thresholds for drawing on the CR in response to risk events that cause task cost overrun for each interval. Risk responses are developed based on matching two thresholds: one related to cost overruns in the ST and the other related to cost overruns in the LT. The project execution is simulated multiple times, changing the threshold levels. Given an initial CR level, the CMF optimizes the threshold levels to minimize the risk of cost overruns while ensuring the CR is used efficiently. This reduces the possibility that project managers develop suboptimal risk responses that could undermine a project's cost and schedule performance. The CMF considers the statistical properties of task cost distributions and their correlation with other tasks, providing a more realistic approach to contingency management optimization. The ability to optimize the allocation of CRs based on statistical analysis and simulation can lead to better decision-making and cost control during project execution.

# Bibliography

- Ballesteros-Pérez, P., Sanz-Ablanedo, E., Mora-Melià, D., González-Cruz, M. C., Fuentes-Bargues, J. L., and Pellicer, E. (2019). Earned Schedule min-max: Two new EVM metrics for monitoring and controlling projects. *Automation in Construction*, 103:279–290.
- Banks, J., Carson, J., Nelson, B., and Nicol, D. (2009). *Discrete-Event System Simulation*. Prentice-Hall, 5th edition.
- Barraza, G. A. and Bueno, R. A. (2007). Cost Contingency Management. *Journal of Management in Engineering*, 23(3):140–146.
- Barrientos-Orellana, A., Ballesteros-Pérez, P., Mora-Melià, D., and Cerezo-Narváez, A. (2022). Comparison of the Accuracy of Cost Prediction Methods With Earned Value Analysis. *Proceedings from the International Congress on Project Management and Engineering*, 2022-July(July):12–21.
- Barrientos-Orellana, A., Ballesteros-Pérez, P., Mora-Melià, D., Cerezo-Narváez, A., and Gutiérrez-Bahamondes, J. H. (2023). Comparison of the Stability and Accuracy of Deterministic Project Cost Prediction Methods in Earned Value Management. *Buildings*, 13(5):1–18.
- Batselier, J. and Vanhoucke, M. (2015a). Construction and evaluation framework for a real-life project database. *International Journal of Project Management*, 33(3):697–710.
- Batselier, J. and Vanhoucke, M. (2015b). Evaluation of deterministic state-of-the-art forecasting approaches for project duration based on earned value management. *International Journal of Project Management*, 33(7):1588–1596.
- Borges, W. F. and Mário, P. d. C. (2017). Five Project-Duration Control Methods in Time Units: Case Study of a Linearly Distributed Planned Value. *Journal of Construction Engineering and Management*, 143(6):05017002.
- Chang, H. K., Yu, W. D., and Cheng, T. M. (2020). A Quantity-Based Method to Predict More Accurate Project Completion Time. *KSCE Journal of Civil Engineering*, 24(10):2861–2875.
- De Koning, P. and Vanhoucke, M. (2016). Stability of earned value management do project characteristics influence the stability moment of the cost and schedule performance index. *Journal of Modern Project Management*, 4(1):9–25.
- De Marco, A., Narbaev, T., Rafele, C., and Cagliano, A. C. (2017). Integrating risk in project cost forecasting. *Proceedings of the Summer School Francesco Turco*, 2017-September:117–122.
- Du, J., Kim, B.-C., and Zhao, D. (2016). Cost Performance as a Stochastic Process: EAC Projection by Markov Chain Simulation. *Journal of Construction Engineering and Management*, 142(6):1–13.
- Eldosouky, I. A., Ibrahim, A. H., and Mohammed, H. E. D. (2014). Management of construction cost contingency covering upside and downside risks. *Alexandria Engineering Journal*, 53(4):863–881.

- Elshaer, R. (2013). Impact of sensitivity information on the prediction of project's duration using earned schedule method. *International Journal of Project Management*, 31(4):579–588.
- Everitt, B. S. and Skrondal, A. (2010). *The Cambridge Dictionary of Statistics*. Cambridge University Press, Cambridge.
- Gálvez, E., Capuz-Rizo, S. F., and Ordieres-Meré, J. (2015). Analysis of Project Duration Uncertainty Using Global Sensitivity Analysis. *Journal of Modern Project Management*, 2(3):18–25.
- Gálvez, E. D., Ordie, and Capuz-Rizo, S. F. (2017). A Method for Identification of Critical Scheduling Decisions. *Journal of Modern Project Management*, 5(1):46–61.
- Karimi, H., Taylor, T. R. B., Dadi, G. B., Goodrum, P. M., and Srinivasan, C. (2018). Impact of Skilled Labor Availability on Construction Project Cost Performance. *Journal of Construction Engineering and Management*, 144(7):1–10.
- Khamooshi, H. and Golafshani, H. (2014). EDM: Earned Duration Management, a new approach to schedule performance management and measurement. *International Journal of Project Management*, 32(6):1019–1041.
- Khamooshi, H., Mamghaderi, M., and Kwak, Y. H. (2021). Project duration forecasting: A simulation-based comparative assessment of earned schedule method and earned duration management. *Journal of Modern Project Management*, 10(1):7–19.
- Kim, J. E., Nembhard, D. A., and Kim, J. H. (2016). The effects of group size and task complexity on deadline reactivity. *International Journal of Industrial Ergonomics*, 56:106–114.
- Kim, J. H. (2019). Multicollinearity and misleading statistical results. *Korean Journal of Anesthesiology*, 72(6):558–569.
- Lance Stephenson, H. (2015). *AACE International Total Cost Management Framework: An Integrated Approach to Portfolio, Program, and Project Management*. AACE International.
- Law, A. (2003). How to conduct a successful simulation study. In *Proceedings of the 2003 Winter Simulation Conference, 2003.*, pages 66–70. IEEE.
- Liao, P.-C., O'Brien, W. J., Thomas, S. R., Dai, J., and Mulva, S. P. (2011). Factors Affecting Engineering Productivity. *Journal of Management in Engineering*, 27(4):229–235.
- Lipke, W. H. (2003). Schedule is Different. *The Measurable News*, 2:31–34.
- Narbaev, T. and De Marco, A. (2011). Cost Estimate at Completion Methods in Construction Projects. *2011 International Conference on Construction and Project Management*, 15(February 2014):32–36.
- National Defense Industrial Association (2021). A Guide to Managing Programs Using Predictive Measures. Technical report, National Defense Industrial Association, Integrated Program Management Division (IPMD), Arlington.
- Ottaviani, F. M. and De Marco, A. (2021). Multiple Linear Regression Model for Improved Project Cost Forecasting. *Procedia Computer Science*, 196(2021):808–815.
- Putnam, L. H. (1978). A General Empirical Solution to the Macro Software Sizing and Estimating Problem. *IEEE Transactions on Software Engineering*, SE-4(4):345–361.
- Santos, J. I., Pereda, M., Ahedo, V., and Galán, J. M. (2023). Explainable machine learning for project management control. *Computers and Industrial Engineering*, 180(April).

- Vanhoucke, M. (2010). Using activity sensitivity and network topology information to monitor project time performance. *Omega*, 38(5):359–370.
- Vanhoucke, M. (2012). Measuring the efficiency of project control using fictitious and empirical project data. *International Journal of Project Management*, 30(2):252–263.
- Vanhoucke, M. (2019). Tolerance limits for project control: An overview of different approaches. *Computers and Industrial Engineering*, 127(October):467–479.
- Vanhoucke, M., Coelho, J., and Batselier, J. (2016). An Overview of Project Data for Integrated Project Management and Control. *Journal of Modern Project Management*, 3(3):6–21.
- Vose, D. (2008). *Risk Analysis: A Quantitative Guide*. John Wiley & Sons, 3rd edition.
- Warburton, R. D. (1983). Managing and Predicting the Costs of Real-Time Software. *IEEE Transactions on Software Engineering*, SE-9(5):562–569.
- Warburton, R. D., De Marco, A., and Sciuto, F. (2017). Earned schedule formulation using nonlinear cost estimates at completion. *Journal of Modern Project Management*, 5(1):75–81.
- Willems, L. L. and Vanhoucke, M. (2015). Classification of articles and journals on project control and earned value management. *International Journal of Project Management*, 33(7):1610–1634.