Single- and multivariate methods for statistical project control using earned value management

Jeroen Colin
2015

Promotor: Prof. dr. Mario Vanhoucke

Promotor:

    Prof. Dr. Mario Vanhoucke

Examination commission:

Prof. Dr. Marc De Clercq (voorzitter)	Ghent University
Prof. Dr. Mario Vanhoucke (promotor)	Ghent University,
	Vlerick Business School,
	University College London (UK).
Prof. Dr. Pierre Bonnal	CERN, the European Organization for
	Nuclear Research (Switzerland)
Prof. Dr. Young Hoon Kwak	George Washington University (USA)
Prof. Dr. Broos Maenhout	Ghent University
Prof. Dr. Geert Poels	Ghent University

Ghent University
Faculty of Economics and Business Administration

Business Informatics and Operations Management
Tweekerkenstraat 2, B-9000 Ghent, Belgium

Tel.: +32 (0)9 264 35 17
Fax.: +32 (0)9 264 42 86

UNIVERSITEIT
GENT

Dankwoord

Dit boek is tot stand kunnen komen door de steun en bijdragen van enkele speciale personen. Het is dan ook gebruikelijk hier, voor de uiteenzetting van het geleverde onderzoek aanvangt, enkele woorden van dank te uiten. Ik heb hierbij niet getracht een volledige oplijsting te maken van iedereen die ik de afgelopen vijf jaar heb leren kennen en die op die manier op unieke wijzen hebben bijgedragen aan dit werk. Dat zou een onbegonnen taak zijn en wat meer is, ik zou het risico lopen mensen over het hoofd te zien. Om dat risico toch enigszins te vermijden wil ik alvast de lezer van dit boek – jou dus – bedanken. In meer of mindere mate zal jij betrokken geweest zijn in mijn leven en/of mijn onderzoek. Is het niet omwille van een interesse in mijn persoon, dan is het wel omdat je waarde hecht aan het werk dat ik geleverd heb. Ik geloof graag dat jij verder zal lezen dan dit dankwoord in dit boek. Waarvoor bedankt! Is dat niet zo, dan geloof ik graag dat jij de grote lijnen al kent, omdat je me er ergens in de afgelopen vijf jaar naar hebt gevraagd. Reden te meer dan, om jou te bedanken. Die interesse heeft mij geholpen dit werk te voltooien. Naast jou – de lezer – wil ik nog een aantal mensen in het bijzonder bedanken.

In de academische sfeer zijn dat alle mensen met wie ik mijn onderzoek hebben kunnen delen, wie voor ondersteuning hebben gezorgd en wie de werkomgeving aangenaam maakten. Kortom, de mensen die voor het kader hebben gezorgd, of daar deel van uit maakten, dat mij in staat heeft gesteld een doctoraat af te leggen. Dit gaat over de mensen aan wie ik mijn onderzoek heb kunnen presenteren (op heel diverse nationale en internationale fora) en met wie ik onderhoudende discussies heb kunnen voeren over hoofd-, bij- en randzaken. Wie mij kent, weet dat ik een geanimeerde discussie op tijd en stond weet te waarderen en dat ik daarbij als onderwerp graag dat laatste, bijna evenzeer als de eerste twee, aansnijd. In het bijzonder gaat deze paragraaf over de leden van mijn begeleidings-, lees- en examencommissie, actieve en minder actieve leden van de onderzoeksgemeenschap in operationeel onderzoek en project management en de leden van onze vakgroep. Specifiek dien ik hierbij uiteraard mijn promotor, Mario, een speciale plaats te geven. Alomtegenwoordig in alle bovenstaande categorieën, heeft hij de grootste bijdrage geleverd. Ik heb nog steeds het gevoel dat onze samenwerking, doorspekt van professionele

of pintjes in de sofa, aan ontspanning met Sofie geen gebrek. Vervolgens rest het me dan ook nog m'n ploegmakkers van "economini" te vermelden. Ook al lijkt het alsof ik de afgelopen jaren niets anders heb gedaan dan gevoetbald, ik ben van de overtuiging dat net die combinatie van kameraadschap en lichaamsbeweging een belangrijke ondersteunende factor is geweest die tot het welslagen van mijn doctoraat heeft geleid.

Deze laatste paragraaf van dit dankwoord wijd ik aan mijn familie. Hierbij begin ik graag met mijn twee petekinderen, Emiel en Bas. Samen met jullie ouders hebben jullie mij de afgelopen jaren veel ontspanning en vreugde gebracht. Daarnaast wil ik mijn broers, Jasper en Pieter, en hun respectievelijke prachtige gezinnen, Julie, Bas en Miel en Eveline, Mona en Fien, bedanken om een inspiratie te zijn voor mij. Ook mijn grootouders, Moeke, Vake en Bomma wil ik bedanken. Vervolgens neem ik graag de tijd om mijn ouders en schoonouders te bedanken. Mama, Papa, zonder jullie onvoorwaardelijke steun en liefde zou ik nooit geworden zijn wie ik nu ben. Jullie hebben me de kans gegeven om te studeren en me vervolgens steeds aangemaand om door te zetten. Rik en Veerle (en bij uitbreiding de hele familie Ryckeboer-Rens): jullie hebben ook al die tijd meegeleefd met mij en ontspanning geboden. Onze Castel di Tora trips hebben nu al een heel speciale plaats in mijn herinnering en er moeten er nog zo veel volgen. Het meest dankbaar ben ik jullie echter omwille van die fantastische dochter waarmee ik ondertussen bijna 8 jaar het leven deel. Eva, bedankt, mijn lieve schat! Ik voel mij onmachtig in woorden uit te drukken wat jij voor mij en mijn doctoraat de laatste jaren betekend hebt. Niet enkel was jij de hele tijd een voorbeeld – en zie, ik kon zelfs niet anders dan een zin uit jouw dankwoord te stelen – jij bent m'n thuis en mijn basis. Ik steel er dan nog maar een, om het af te leren – je maakt me erg gelukkig. Een dikke twee maanden geleden schonk je me bovendien dan nog eens een prachtige dochter, Elisa. Ze zal zich allicht nooit onze doctoraten herinneren en het valt nog maar af te wachten of ze ooit de interesse zal tonen om die saaie, oude studies nog maar een blik waardig te gunnen, maar op die korte tijd heeft ze op dat van mij alleszins wel een impact gehad.

Jeroen Colin
Gent, 12 April 2015

Samenvatting

Wat hebben het bouwen van een nieuw ziekenhuis, het boren van een nieuwe tunnel, het inrichten van een woning of het uitrollen van een nieuwe website gemeen? Het zijn allemaal projecten waarin in meer of mindere mate belangen spelen, geld in omgaat en die binnen een bepaalde tijd moeten kunnen worden gerealiseerd. Van dat soort projecten kan ieder nog wel een aantal voorbeelden verzinnen en het moet dan ook niet verwonderen, dat de term 'project manager' bij velen op het visitekaartje prijkt. Dit brengt ons onmiddellijk bij een eerste afbakening van het onderzoek dat in dit boek beschreven staat. Dit onderzoek richt zich op die projecten die zo groot zijn, dat een enkele persoon tijdens de uitvoering van de taken van het project niet meer bij machte is om de opvolging te verzorgen. Met andere woorden, als een project wordt uitgevoerd en er op elk moment zo veel taken in behandeling zijn dat de project manager niet meer in staat is om, hetzij telefonisch of per mail, hetzij in lijve, de voortgang ervan op te volgen. Ons onderzoek is gericht op het eenvoudiger maken van de taak van de project manager in zulke grote projecten.

Een manier waarop de dagtaak van een project manager eenvoudiger kan gemaakt worden is door op voorhand een aantal activiteiten uit het project af te zonderen die speciale aandacht vereisen. Een andere manier is om een brede kijk te houden, maar een meetsysteem te installeren dat de project manager waarschuwt als er iets mis dreigt te lopen of als er een bepaalde activiteit bijzondere aandacht moet krijgen. In de literatuur worden deze twee aanpakken als respectievelijk 'bottom-up' en 'top-down' genoemd. Wij hebben ervoor gekozen om de laatste te onderzoeken. Die aanpak 'van bovenaf' zorgt ervoor dat geen enkele activiteit wordt genegeerd en dat er op voorhand minder aannames moeten worden gemaakt. Uit vroeger onderzoek was trouwens gebleken dat een aanpak van bovenaf heel goed werkt voor projecten die vooral bestaan uit lange opeenvolgingen van activiteiten. Daartegenover staat dat een aanpak 'van onder uit' dan weer beter werkt voor projecten waarin activiteiten veelal in parallel kunnen worden uitgevoerd. De opzet was dus simpel – maar niet eenvoudig – om de werking van een aanpak van bovenaf te verbeteren zodat het op alle soorten grote projecten een bevredigend resultaat kan opleveren. Ultiem moet dit kunnen leiden tot een unieke aanpak die voor alle projecten

werkt en waardoor het leven van de project manager eenvoudiger wordt gemaakt.

Zo een meetsysteem om projecten van bovenaf op te volgen bestaat al geruime tijd. De 'Earned Value Management' (EVM) methodologie werd sinds de jaren '60 van vorig eeuw in de Verenigde Staten zelf zo populair dat het een verplichting werd voor grote overheidsprojecten en bijhorende certificaten, gebruikersbijeenkomsten en goeroes kreeg. EVM meet van bovenaf de voortgang van een project, waarbij zowel voor het tijdsaspect (zal mijn project wel op tijd klaar zijn?) als voor de kosten (gaat mijn budget niet te veel overschreden worden?) aandacht kan besteed worden. EVM werkt als een soort verkeerslicht voor de voortgang van een project. Is het licht groen, dan kan de project manager rustig achterover leunen en de zaken op hun beloop laten. Springt het licht op rood, dan is het tijd voor actie en moeten er stappen ondernomen worden of op onderzoek uitgegaan worden om het project terug op het rechte pad te brengen. Specifiek voor het tijdsaspect werd een aantal jaren terug de 'Earned Schedule' (ES) methodologie afgeleid, als uitbreiding en verbetering. Opnieuw geldt echter nog dat de opvolging van projecten met een veelal parallelle structuur soms de wensen overlaat.

Het onderzoek dat in dit boek beschreven staat is gericht op het analyseren, kwantificeren en verbeteren van de project opvolging door middel van het gebruik van de combinatie van EVM en ES, hierna EVM/ES genoemd. Onze aandacht is daarbij uitsluitend naar de opvolging van het tijdsaspect van een project gegaan. Met andere woorden, gegeven een planning voor de aanvang van een project met daarin geschatte duurtijden voor alle activiteiten, hoe weet een project manager dan wanneer actie te ondernemen of op onderzoek uit te gaan. Dit leidt tot de volgende drie onderzoeksvragen:

- Hoe goed werkt EVM/ES, zoals het wordt toegepast in de praktijk?

- Hoe goed werkt EVM/ES als we het vergelijken met alternatieven?

- Hoe kan de toepassing van EVM/ES in de praktijk verbeterd worden?

Om een antwoord te vinden op die vragen moesten we ons perspectief iets meer concreet maken. Daarom zijn we vertrokken van de formulering van het 'statistische project controle' concept. Hierin beschouwen we de uitvoering van een project ten opzichte van een vooraf gedefinieerde 'toestand van controle'. Dit betekent dat een project manager zelf kan aangeven wat hij/zij wil toelaten van variatie in het project of vroegere succesvolle projecten als referentie te gebruiken. Vervolgens konden we op die manier nagaan of de project opvolgingsmethodes die we onderzochten wel de juiste signalen geven. Als de uitvoering van het project niet overeenstemt met die vooraf gedefinieerde toestand dan moet de project
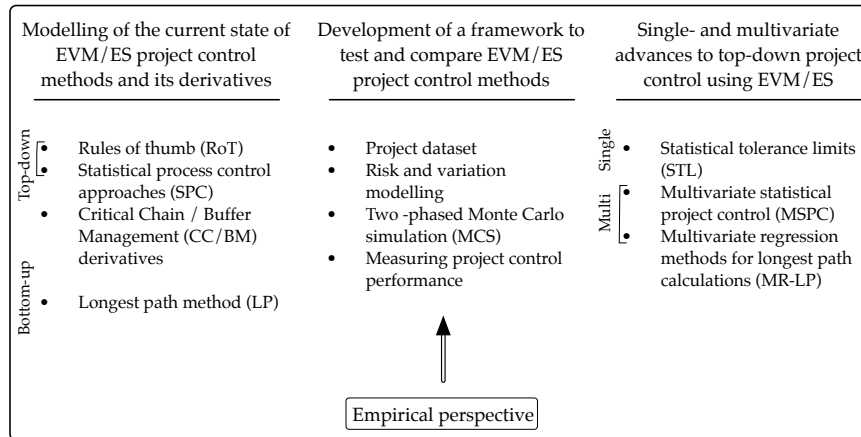
Figuur 1: Schematische weergave van de elementen die gemodelleerd, geanalyseerd en gekwantificeerd zijn in het onderzoek dat wordt beschreven in dit boek.

manager een signaal krijgen van het opvolgingssysteem (rood licht). Voldoet de uitvoering wel aan die toestand, dan moet er geen signaal worden gegenereerd (groen licht).

Figuur 1 geeft in een schematische weergave de structuur en de elementen van het onderzoek dat beschreven staat in dit boek. De linkerkolom van figuur 1 geeft de huidige staat van projectopvolging technieken zoals ze worden gebruikt in de dagdagelijkse praktijk. Vier methodes werden hiervoor gemodelleerd naar het voorbeeld van hoe ze in de praktijk worden gebruikt. Deze methodes passen het EVM/ES meetsysteem toe of bieden een alternatief. Om vanuit deze modellering een antwoord te bieden op de twee eerste onderzoeksvragen, moeten deze methodes met elkaar kunnen worden vergeleken. De tweede kolom geeft de elementen weer van het simulatiekader dat werd ontwikkeld om deze vergelijking te kunnen maken. Dit simulatiekader is gebaseerd op Monte Carlo simulaties (genoemd naar de stad waar kansspelen, geluk en ongeluk welig tieren), een grote dataset van projecten, metrieken om signalen met elkaar te vergelijken en een duale karakterisatie van hoe onzekerheid zich manifesteert in projecten die worden uitgevoerd in de praktijk. Ten slotte toont de meest rechtse kolom de drie nieuwe methodes die werden ontwikkeld om het gebruik van EVM/ES voor projectopvolging te verbeteren en bijgevolg een antwoord bieden op de derde onderzoeksvraag. In deze methodes worden simulaties niet enkel toegewend om het gebruik ervan te kwantificeren en te meten of de signalen die zij voortbrengen betrouwbaar zijn, maar de simulaties worden ook toegepast in de ontwikkeling van de methodes. De nieuw ontwikkelde methodes voor projectopvolging gebruiken statistiek

en artificiële intelligentie om op basis van simulaties van een project een meer betrouwbaar kader te creëren waartegen de echte uitvoering (middels het EVM/ES meetsysteem) kan worden vergeleken en waarna signalen kunnen worden gegenereerd. De eerste nieuwe methode past tolerantie limieten toe op basis van de kwantielen van empirische distributies, maar beschouwt de variabelen uit het EVM/ES meetsysteem afzonderlijk. In die zin is het een enkel-variabele methode. De twee andere methodes vertrekken vanuit een meer-variabele beschouwing van het EVM/ES meetsysteem, waarbij alle variabelen simultaan worden geanalyseerd en vergeleken. Samen bepalen deze drie methodes de meest opmerkelijke toevoeging van ons onderzoek aan de theorie rond project management en vormen ze de titel van dit boek: "Enkel en meer-variabele methodes voor statistische project controle, gebruikmakende van het EVM/ES systeem."

Omwille van de speciale plaats die simulaties innemen in ons onderzoek, met name voor de vergelijking van alle beschreven methodes en de ontwikkeling van de laatst genoemden, hebben we er bijzondere aandacht aan besteed. In figuur 1 is dit element afgezonderd terug te vinden onderaan de middelste kolom. Het betreft een empirische validatiestudie waarbij we de grootste beschikbare dataset van uitvoeringen van projecten onder de loep genomen hebben. Daarvoor hebben we een theoretische distributie, die reeds vroeger aan de praktijk was getoetst – maar nooit op deze schaal en met een vooropgestelde reproduceerbaarheid – losgelaten op deze dataset. De analyse van de resultaten die daaruit volgden tonen aan hoe onze simulaties kunnen worden gevalideerd met empirische data en hoe de inputs van ons simulatiemodel kunnen worden gekalibreerd aan echte data van projecten die reeds zijn uitgevoerd in de praktijk.

De voornaamste resultaten uit ons onderzoek zijn de modellering van vier methodes uit de huidige staat van projectopvolging in de praktijk, de opbouw van een omvangrijk en gevalideerd simulatiekader en de ontwikkeling van drie nieuwe project opvolgingsmethodes die gebruik maken van EVM/ES, zoals schematisch weergegeven in figuur 1. Daarnaast stelt de combinatie van die onderzoeksresultaten ons ook in staat een antwoord te bieden op de eerder gestelde onderzoeksvragen:

  - We weten nu dat het EVM/ES meetsysteem in de praktijk zal werken, maar dat het met het oog op het produceren van zo betrouwbaar mogelijke signalen best gekoppeld wordt aan controlelimieten (die de signalen voortbrengen) die gebaseerd zijn op gesimuleerde project uitvoeringen.

  - Als alternatieven voor een rechttoe-rechtaan implementatie van het EVM/ES meetsysteem kan het gebruik ervan gekoppeld worden aan statistische proces controle (SPC) kaarten, kunnen meerdere controle punten op verschillende plaatsen in de projectplanning worden geïnstalleerd

en kan EVM/ES ook 'van onder uit' worden toegepast. Deze aanpassingen komen de effectiviteit van de gegenereerde signalen ten goede, maar komen de efficiëntie van de signalen niet altijd ten goede.

- Het gebruik van het EVM/ES meetsysteem voor projectopvolging in de praktijk kan verbeterd worden (vanuit zowel een doeltreffendheid als een efficiëntie oogpunt) door middel van het gebruik van enkel en meer-variabele methodes. Deze zouden moeten worden geïmplementeerd in een informaticasysteem dat de complexiteit van de methodes afschermt van de eindgebruiker. De combinatie van simulaties en statistische analyses moet leiden tot een beter referentiekader waartegen de EVM/ES metingen voor een projectuitvoering kan worden vergeleken en waarop signalen kunnen worden gebaseerd.

Het onderzoek dat beschreven staat in dit boek werd aanvankelijk beschreven in zes papers die onafhankelijk van elkaar kunnen worden gelezen. Deze zes papers vormen nu de ruggengraat van dit boek als hoofdstukken 2 tot 7. Daarnaast voorzien hoofdstukken 1 en 8 het boek van een introductie en een conclusie. Aan het begin van elk van de hoofdstukken 2 tot 7 geeft een reproductie van figuur 1 weer welke elementen in dat hoofdstuk worden behandeld en beschreven. Bij het drukken van dit boek werden hoofdstukken 2 tot 5 reeds gepubliceerd of aanvaard voor publicatie in respectievelijk Omega – The international Journal of Management Science, Expert Systems with Applications, Construction Engineering and Management en International Journal of Project Management. Hoofdstukken 6 en 7 werden respectievelijk nog herwerkt en onder beschouwing genomen door een tijdschrift.

Table of Contents

List of Figures

List of Tables

1

Introduction

1.1 General introduction

This book is written to summarize the research that we have done on single- and multivariate methods for statistical project control using earned value management, between 2009 and 2014. In order to frame this research in the larger body of literature on project management, it is indispensable to start with a definition for project management, that we find to be appropriate. Although formal development in project management has been around since the Manhattan project in the 1940s [1], still no clear consensus can be found amongst academics, let alone practitioners, on a definition of project management [2]. Therefore we had to choose one, from which this discussion could start. We cite the characterisation that was proposed by Turner (1996) [3], who proclaimed that project management can be described as: "the art and science of converting vision into reality." Although this definition might appear vague, we believe that it appropriately covers the two main aspects that can be found in contemporary project management writings [4]. From our perspective, "the art" covers the soft skills that are required in project management [5], while "the science" adheres to the classic paradigm of operations research (OR)-related tools and techniques that are applied to manage projects [6].

The distinction between these two main aspects is also apparent in the literature on project management success [7–9] . Here, the difference between

the two aspects is translated to "project succes" and "project management success". The former deals with the psychological, organizational, geographical and cultural considerations of whoever is asked to judge the success of the conversion of the "vision into reality" [10–12]. The latter however, deals with the tangible, objective, and measurable dimensions of a project [6, 13].

From their analysis regarding project management from top management journals, Kwak and Anbari (2009) [14] concluded that the most researched fields over the preceding 50 years were strategy and portfolio management, OR and decision science, organisational behaviour and human resource management. Their findings were in part confirmed by Ingason and Jonasson (2009) [15], which show how academics from the project management discipline start focussing more and more on interpersonal competences, relationship management, resource management and strategic alignment. Our research on project management however, is in the field of OR and decisions science and therefore concurs to the prevailing paradigm of the application of hard skills in project management [4]. More precisely, we have taken up the "instrumentalist view" [16] and tried to advance the tools and techniques on how to achieve project management success, by focussing on the objective and measurable dimensions of project management.

These dimensions (time, cost and scope) are expressed in the "Iron Triangle" of project management [2]. When respondents were asked to judge whether their projects were performed successfully by White and Fortune (2002) [11], 41% claimed to have delivered on time, to budget and to specifications. For the two first dimensions Batselier and Vanhoucke (2014) [17] confirmed these findings with comparable figures. Even though algorithms have been adapted to cope with more and more added complexity in project planning (see Hartmann et al. (2010) [18] for a recent review), the stochastic environment in which projects are performed and the uncertainty associated with the estimates with which project schedules are constructed, lead to a majority of projects that fails to meet time, budget and scope specifications. This shows the need for project control in practice, and consequently justifies research on project control as the topic off this book.

More precisely we will focus, in this book, on the time aspect of project control, i.e. project schedule control. Although in personal encounters with practitioners* we more than often heard that time should be the main focus and that the other dimension could be reduced to it, we still believe that research on the control of both cost and scope is valid and beneficiary to the project management practice. We did however, not incorporate it into

---

*At any of the four last meetings of the EVM Europe conference `http://www.evm-europe.eu/`

our research. The reason for this will be described in more detail in the limitations section of the last chapter of this book.

The subject of project control has always had a strong linkage to Earned Value Management (EVM), ever since its formal conception in the 1960s as the Cost/Schedule Control Systems Criteria (C/SCSC) on US department of defence projects. The popularity of EVM has been increased ever since among practitioners and academics [17, 19–25]. Intrinsically linked to the time dimension, and the topic of project schedule control, is the recent extension to EVM that was promoted by Lipke et al. (2009) [26]: Earned Schedule (ES). In the remainder of this book we will, more often than not, consider ES as an indispensable part of the earned value methodology and we will proceed with the more inclusive acronym EVM/ES.

The use of EVM/ES for project schedule control has been extensively described in literature, where it was compared to other project schedule control methods such as the critical path method (CPM), the Project Evaluation and Review Technique (PERT) and Schedule Risk Analysis (SRA) [27]. We will not give a full review of such studies and the alternatives to EVM/ES here. This book is build up around chapters that constitute papers which are self-containing and which have been submitted to international peer-reviewed journals. It is therefore argued that the literature review should be found along the different chapters of this book.

We will however, consider the debate that was ongoing when Vanhoucke (2010, 2011) [28, 29] published his studies on top-down and bottom-up procedures for project control. A top-down control approach refers to the use of a project control system that generates project based performance metrics to give a general overview of the project performance. Warnings signals can be triggered based on these general performance metrics, which need further investigation to detect problems at the activity level. A bottom-up project control approach refers to a system in which detailed activity information needs to be available constantly during the project control process, and which consequently requires more effort [30]. At the time of the debate, Fleming and Koppelman (2005) [31] had argued that the EVM/ES system produces erroneous and misleading warning signals for project schedule control when it is implemented at a high level of the work breakdown structure (WBS) of a project. Therefore, according to them, it should only be implemented at the activity level of the project. In essence, EVM/ES would then be a bottom-up approach and would present only an alternatively stylized record of the activity level performance. Lipke et al. (2009) [26], on the other hand, claimed that this would be a time-consuming and often disruptive task for the project management team and that the EVM/ES system should be implemented at a high level of the WBS as a top-down approach.

In the research, that is described along the chapters of this book, we have

followed the advice of Lipke et al. (2009) [26] and consider EVM/ES implementations primarily as top-down approaches. In order to cope with the sometimes erroneous and misleading nature of the warning signals for schedule control, we have set it a priority to investigate and test the dynamics of the EVM/ES system during the executions of projects. Moreover, we have adopted a spectrum of machine learning methods and statistics to overcome this inherent nature of EVM/ES and to improve the schedule control process that follows from EVM/ES, while keeping a focus on the effort that is spent in that process. This was done by the conception of the statistical project control framework [32] with which all of our studies are linked.

The next section (Section 1.2) of this introductory chapter describes what this "statistical project control" framework is. Next, in section 1.3, we will introduce the elements of the research that was done for this book and how our research was structured. Finally, in section 1.4, we will present the further outline of this book and we will describe which elements of our research are addressed in which paper and consequently, what the different chapters are constituted of.

1.2   What is statistical project control?

To the best of our knowledge, we were the first to introduce the term "statistical project control" in our article on setting tolerance limits for project schedule control that was published in Omega in 2014 (Colin and Vanhoucke, 2014 [32]). For a full coverage of the concept, the reader is referred to chapter 2, where he/she will find the final version of this article, or to the cited reference in Omega. Here, we will rephrase the concept of statistical project control as *the use of simulated project progress situations in order to develop and test practically applicable approaches for project schedule control using EVM/ES*. We will proceed in this section with a discussion of the different components of this new phrasing of the concept.

> *Simulated*: We use Monte Carlo simulations to create fictitious project executions. For each project, a baseline schedule will first be constructed. Then, in the Monte Carlo simulations, the activity durations of the activities are varied with respect to the estimates that were used to construct the baseline schedule.

> *project progress situations*: The fictitious project executions are simulated as if they were really executed at an hour-by-hour basis. Therefore, we are able to get progress measurement after each review period, whether this is an hour, a week, a month or larger.

EVM/ES: These progress measurements will be formulated according to the implementation of the EVM/ES system. From the baseline schedule and the simulated progress, up to a given review period, the EVM/ES system allows schedule performance measures to be produced.

develop and test: The EVM/ES schedule performance measures are used to test both current-practice methods and new procedures. Moreover, they can specifically be deployed to developed these new procedures.

practically applicable: For the statistical project control framework, we assume that real-world executions of a project can be emulated, from which we can consequently test the performance of schedule control methods. In addition, we also propose simulations to be run prior to the execution of a real project. These simulations then serve as input in new methods which deploy the capabilities of machine learning and statistics, which can be applied during that real-life project control case.

In addition to these components from the new phrasing of the concept of statistical project control, we would like to stress an additional element.

testing against a pre-defined state of schedule control: The newly developed methods for schedule control, which deploy machine learning and statistics, are fuelled with EVM/ES data from fictitious project progress situations that are produced by Monte Carlo simulations. These simulations allow the user to define a state of schedule control to be tested against. This state of schedule control can be defined in accordance with a subjective desirable outcome, as specified by the project manager, or can be calibrated to historical records of "successful projects". Irrespective of how this state is defined, the warning signals that are produced by a schedule control method in a dynamic project progress situation can always be referenced against it. A true warning signal is then said to be produced, by the control method, if the subjected project progress situation deviates from the pre-defined state of schedule control. Alternatively, a warning signal is said to be false, when a control method produces a signal for a subjected project progress situation that does not deviate from the pre-defined state of schedule control.

We should consider a final remark, concerning our statistical project control framework. In the preceding exploration of the concept, we have only mentioned the warning signals that are produced by a control method. The discussion of project control methods, which we present in this book, is restricted to their ability to generate warning signals. We have not incorporated actions that can be taken to bring a project back on track (i.e.
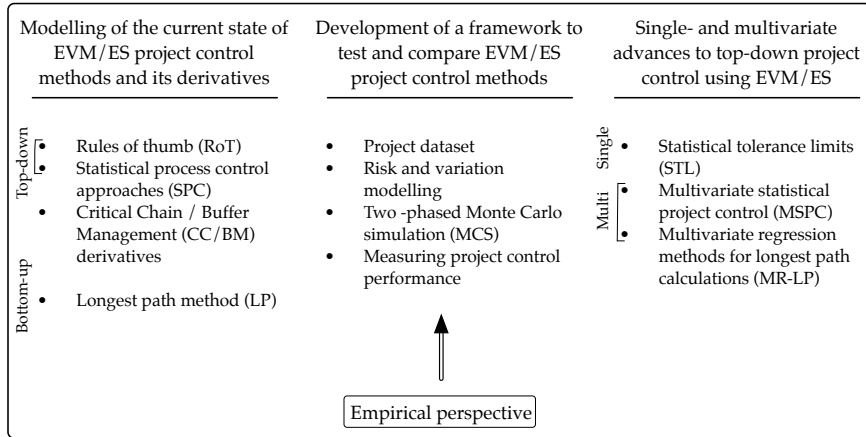
Figure 1.1: Schematic overview of the research on project schedule control that is presented in this book

bring it closer to the baseline schedule) or to protect a deadline that needs to be met. As discussed earlier, we adhere to the concept of implementing EVM/ES as a top-down schedule control approach. We therefore consider only the information (i.e. warning signals) that is made available to the project manager at a high level of the WBS. It is then reasoned that this project manager should invest time and effort into a drill-down of the WBS in order to find the cause of this warning signal. Based on the more detailed activity level view of the project he/she then obtains, the project manager should then be able to find a fitting corrective action. We will discuss the limitations of our research further in the concluding chapter of this book.

1.3 Summary and structure of the presented research on statistical project control using EVM/ES

1.3.1 Research questions

This book was not written from scratch in order to provide an extensive summary of the work that was done. Rather, it should be considered as a collection of six self-containing research articles that examine a mutual subject. These research articles constitute chapters 2 to 7 of this book. However, in order to guide the reader through this book and to stress the com-

mon ground of the research articles, two elements of structure were added, which will be discussed along the following paragraphs. Moreover, since the chapters that constitute the main part of this book are self-containing, we have not added cross-references to the chapters of this book in the text. References are provided in a list as in normal research articles. The other chapters of this book, to which a reference might be made along the lines of a certain chapter, are denoted by their title and the journal in which they are published. It is advised that the reader uses this introduction (section 1.4 in particular) to identify the appropriate chapter.

The first structural element is figure 1.1. It shows a schematic overview, in broad strokes, of the research that we did in the statistical project control framework. We will go into detail on the elements that are present in each of the three columns of figure 1.1 in the next section, but first we will provide the research questions that are addressed in this book, as the second structural element.

These research questions were only informally examined during our research and the reader should consider their discussion in this book accordingly. Given the scope of our research, as described by the definition of the statistical project control framework, the following three questions were stated:

Q1: How well does the EVM/ES system work for project schedule control?

Q2: How does the performance of the EVM/ES system compare to that of its alternatives?

Q3: How can the use of the EVM/ES system, in practice, be improved?

We will come back to these three research questions when we discuss figure 1.1 in the next section and provide a brief answer to them in the conclusions of chapter 8.

1.3.2 Structure

The structure of our research is depicted schematically in figure 1.1. We will first consider what each column of this figure represents. The further chapters of this book, and the papers that constitute these chapters, all combine a *simulation framework to test and compare EVM/ES project control methods* with either *current-state EVM/ES project control methods and derivatives* or *new single- and multivariate methods* or both.

Before proceeding, we would like to direct the reader's attention to rectangle at the bottom of figure 1.1 that states: *empirical perspective*. Since the *simulation framework to test and compare EVM/ES project control methods* is

of such great importance in the papers that constitute the chapters of this book, we have paid special attention to what inputs the Monte Carlo simulations should get in order to produce reliable and generalizable results. As we have mentioned in the previous section, a pre-defined state of schedule control should fuel our Monte Carlo simulations. This state of schedule control is always characterized by the uncertainty modelling that is employed on the activity durations. This uncertainty modelling can concur with a subjective desirable outcome, as specified by the project manager, or can be calibrated to historical records. In our *empirical perspective*, for which the paper will be presented in chapter 4, the latter is illustrated. There we take the database records of Batselier and Vanhoucke [17] and show how they can be interpreted using a statistical model from literature in order to produce a classification that can be used in a *simulation framework to test and compare EVM/ES project control methods*. Moreover, this classification is also used in the paper that is presented in chapter 5.

In the more in-detail discussion of the columns of figure 1.1 that follows, the reader should pay special attention to how the elements of these columns can combine to answer the research questions of section 1.3.1. In our research, we did just that. Specifically, we have set up a *simulation framework to test and compare EVM/ES project control methods*. By doing so, and in combination with the elements of the *current-state EVM/ES project control methods and derivatives*, we were able to answer **Q1** and **Q2**. In addition, **Q3** is answered by the elements of the *new single- and multivariate methods*.

Modelling of the current state of EVM/ES project control methods and its derivatives

The contribution of the research, that was done for this book, lies not only in the definition of the statistical project control framework and the simulation models and techniques to measure project control performance, but also in the three newly developed methods for top-down project schedule control using EVM/ES (see *Single- and Multivariate advances for top-down project control using EVM/ES* and chapters 2, 6 and 7). However, in order to show this contribution in the respective papers, we had to show how these methods outperform the current state of EVM/ES project control methods.

The modelling of this current state of EVM/ES project control methods compelled us to make a choice regarding which methods and strategies to include in our research. Since an exhaustive modelling of all methods and strategies that are applied, in practice, to EVM/ES performance measures is unobtainable, we have chosen to incorporate only those that were either described as common practice (Rules of thumb (RoT) and a CPM-like implementation of EVM/ES, i.e. the Longest path method (LP)), adapta-

tions of alternatives (Critical Chain/Buffer Management (CC/BM)) or hot prospects in literature that have not been tested extensively (Statistical process control charts (SPC)).

From these methods, only the LP method can be classified as a bottom-up approach, since it requires a fully detailed overview of the activity level performance in order to produce warning signals for schedule control. The CC/BM derivative that is presented in chapter 3 uses multiple control points, while the other strictly top-down methods (RoT, SPC) use a single point in the WBS to control the project's schedule. CC/BM is therefore classified in between top-down and bottom-up, leaning somewhat closer to top-down.

We will now proceed with providing some more details regarding the included methods of the current state of EVM/ES schedule control.

Rules of thumb (RoT) model the use of EVM/ES in practice to generate warning signals in a project schedule control process. They have been integrated into our research because of the realization that "the use of EVM/ES in practice is characterized by decision making from practical experience, rules of thumb and anecdotal evidence" [32]. The use of RoT is modelled by setting static, narrowing or widening tolerance limits for EVM/ES schedule performance measures, that act as thresholds to indicate departures from the baseline schedule performance of the project. The use of static thresholds was previously described by Anbari (2003) [19] as deployed in "Target Performance Charts". A clear motivation for the use of narrowing and widening thresholds was given by Colin and Vanhoucke (2014) [32].

Statisical process control (SPC) approaches have been used, in practice, in combination with EVM/ES schedule control measures for schedule control purposes [33–36]. The distinct methods and control charts, suggested by these authors, have all been implemented and integrated into the research that was done for this book.

Critical Chain/Buffer Management (CC/BM) has been around for almost 20 years. Although its assumptions and principals have been met with criticism in literature, records show its popularity for use in practice [37, 38]. The suggestion to compare it with the use of EVM/ES for project schedule control was made to us in a refereeing process. Therefore, we have adapted some of its principles to be used with EVM/ES. Chapter 3 will describe how this was done and how the two control methods that followed from this compare to other techniques for using EVM/ES in a project schedule control process. Although the findings for the method are new and the implementation is the product of original research, we have listed it as a current-state project control method in practice. This is justified by the obser-

vation that building-in multiple control points in a project schedule is done on a regular basis. Although this habit is not always well-documented or algorithmically supported, the usefulness of project control methods that classify in the spectrum between purely top-down and bottom-up is recognized in practice. We therefore propose our CC/BM adaptation to fill this gap in academic literature, but consider it to be part of the current state of project control practice.

Longest path (LP) The longest path method was proposed by Lipke (2012) [39] as a way to cope with the distortion of the warning signals, that are caused by non-critical activities, in the implementation of EVM/ES for schedule control in non-serial project networks. However, in essence, this method requires a full overview of the activity level performance of the project at each review period. We have therefore considered it as a bottom-up approach, closely related to CPM, which requires substantially more effort by the project manager than the previously discussed methods. The LP method requires the longest path to be calculated dynamically during the execution of the project. The EVM/ES schedule performance measures are then based only on those activities that lie on that longest path and from these, warning signals for schedule control can be produced. In chapter 7, we will show how the control effort can be significantly reduced by the application of multivariate regression methods through removal of the requirement that a detailed activity level performance needs to be obtained.

Development of a framework to test and compare EVM/ES project control methods

The multitude of the papers[†] that constitute the chapters of this book deploy a framework in which either current state methods for EVM/ES schedule control or new methods for EVM/ES schedule control or both are tested and compared. Moreover, as we have discussed in the section regarding statistical project control, the simulations that are performed in this framework are not only used to test these methods, but can also be deployed in order to fuel them and to harness the power of statistics and machine learning in a practical project control setting.

The following chapters will discuss the different elements of this framework into more detail and therefore, we will only present a short introduction here.

[†]Only chapter 4 deals with the so-called *empirical perspective* and the framework is only partly applied in the discussion section.

Project dataset: In order for any research on project management to produce results that are as reliable and generalizable as possible, the influence of the underlying network structure of the project needs to be investigated. We have therefore, in all of our studies, deployed a well-known dataset of 900 project networks that was generated using RanGen. [40]

Risk and variation modelling: The uncertainty that should be applied to the activity durations in project simulations has been subject to much debate in literature [41–43]. Without going into detail, on the precise distributional characteristics that should be applied to activity durations, we can state that we have deployed simulations models that introduce uncertainty on two levels. We will explain, along the following chapters of this book, the how and why of a modelling that considers *risk* and *variation* as two factors of uncertainty that are omnipresent in the complex stochastic environments in which projects are executed.

Two-phased Monte Carlo simulation: Since our newly developed methods, for project schedule control using EVM/ES, require the probabilistic input that is provided by simulations, it was indispensable to design experiments in two phases. The *first phase* could then be used to emulate the simulations that are needed to serve as input for the new methods, prior to a project execution. These simulations would also be needed if the new methods were to be applied in a practical project control setting. The *second phase* could then be used to test the performance of these methods, as if they were really applied for project schedule control. In this phase, we could also apply distinctly different distributional characteristics to the activity durations, which allowed us to test the robustness of our proposed methods to these characteristics and the uncertainty modelling that was applied in the first phase.

Measuring project control performance: We had to devise a way to measure the performance of the schedule control procedures that we tested in the research in this book. The majority of the papers, that constitute the chapters of this book, deploy a procedure that was lend from the literature on machine learning and classification [44]. A receiver operating characteristic (ROC) curve was therefore applied. Although we have introduced our own nomenclature for the error rates that were measured in the experiments (*Probability of overreactions*, *Detection performance*), the ROC curves allowed us to measure an *area under the curve* (AUC) that could be used for comparing the different methods with one another. Chapter 3 present an exception to the commonly used AUC for comparison, since its focus lies more on the effort that is spent by the project manager in the control pro-

cess. There, efficiency and control effort (both adapted from a definition by Vanhoucke (2012) [45]) were used to compare the tested methods for schedule control.

Single- and multivariate advances for top-down project control using EVM/ES

We declared earlier in this introductory chapter that the contribution of the research in this book lies for a large part in the development of new methods for project schedule control using EVM/ES. Chapters 2, 6 and 7 will respectively introduce *statistical tolerance limits (STL)*, *multivariate statistical project control (MSPC)* measures and *multivariate regression methods for longest path calculations (MR-LP)* that can be applied for schedule control using EVM/ES. Here, we will already provide a short introduction, but the reader is referred to the individual chapters for a full coverage on the techniques and principles behind these methods.

> *Statistical tolerance limits (STL)* are based on a simulation that should be performed prior to the execution of a project (i.e. in the *first phase*). The empirical cumulative distribution functions for the EVM/ES performance measures, that can be obtained from the simulation, then allow us to calculate sample quantiles for these measures. These sample quantiles then serve as tolerance limits (or thresholds) that can be used during the execution of the project. In chapter 2, we will present two charts (X and R), for project schedule control, using EVM/ES that apply these statistical tolerance limits.

> *Multivariate statistical project control (MSPC)* is built around the use of a principal component analysis (PCA) that analyses the correlation structure of the EVM/ES measures that are produced in *the first phase* simulation. Based on this analysis, two multivariate measures (T^2 and SPE) are produced, which can be applied dynamically during the execution of a project. We will present these multivariate measures for schedule control using EVM/ES in chapter 6.

> *Multivariate regression methods for longest path calculations (MR-LP)* allow to reduce the effort that is spent by the project manager in calculating the longest path dynamically during the execution of a project. In chapter 7, we show how principal component regression (PCR), partial least squares regression (PLSR) and their kernel counterparts (kPCR and kPLSR) can be applied to estimate the activity level performance of a project explicitly, from EVM/ES measures that are calculated at the project level of the WBS.

With respect to the title of this book, the reader should note that STL is a single- variate statistical method, whereas the MSPC and MR-LP meth-

ods utilize multivariate statistics to model the dynamics of the EVM/ES system.

1.4 Outline

We will exploit this last section of this introductory chapter to describe the outline of this book. The chapters that follow represent original research articles that can be considered to be self-sufficient. These articles are either published (chapters 2 and 3), undergoing revision (chapters 4 and 5) or recently submitted (chapter 6 and 7) to international peer-reviewed journals that are listed by Thomson Reuters Web of Science.

Chapter 2 presents the paper "Setting tolerance limits for statistical project control using earned value management" [32]. This paper introduces the STL method and compares it with the use of RoT, CC/BM and LP for project schedule control using EVM/ES.

Chapter 3 introduces the two methods (EVM-FPB and EVM-SNB) for schedule control using EVM/ES that have been derived from the CC/BM methodology. This research was presented in the paper "A comparison of the performance of various project control methods using earned value management systems" [30]. The computational experiment, in this paper, was set up to test and compare the two newly proposed methods with the STL and LP methods.

Chapter 4 presents the paper "An empirical perspective on activity durations for project management simulation studies". This paper was written in order to provide anyone, who wants to adopt our statistical project control approach, with an empirical perspective on the input to the simulation and testing framework. In this paper, we applied a distributional model from literature to the database on Batselier and Vanhoucke (2014) [17] in order to produce a classification for project executions. In the discussion section of this paper, we use this classification as input to our simulation framework for three project management studies.

Chapter 5 presents the paper "Developing a framework for statistical process control approaches in project management". In this paper, we explore the performance of the SPC approaches that have been proposed in literature [33–36] for schedule control using EVM/ES. To that end, we combine our simulation framework with the empirical classification of chapter 4, and compare the SPC approaches with respect to their performance to correctly indicate an out-of-control project progress situation.

Chapter 6 presents the paper "A multivariate approach for top-down project control using earned value management". In this paper, the multivariate schedule control measures (T^2 and SPE from the MSPC method), that are based on a PCA of the EVM/ES correlation structure in a project, are compared to the STL method.

Chapter 7 present the paper "On the use of multivariate regression methods for longest path calculations from earned value management observations". This paper explores the use of multivariate regression methods for estimating the activity durations of a project, in order to dynamically calculate the longest path (MR-LP). The performance of these multivariate regression methods is benchmarked against the LP method, which requires a physical drill-down of the WBS at each review period of the project, and inevitably places a larger burden on the project manager in terms of the effort that is spent in the control process. While the MR-LP methods drastically reduce that effort, we explore in this paper against what loss in performance (reduction in AUC) this is effectively done.

For chapters 2-7, we will reproduce figure 1.1, before the abstract of the paper is presented. In these reproductions (figures 2.1, 3.1, 4.1, 5.1, 6.1, 7.1), we will highlight which elements of the presented research are present in that particular chapter. This is done so that the reader is able to maintain an overview of the presented research, and so that he/she can relate the particular chapter to the larger body of research that is presented in this book. The choice for why a certain element is present in a certain particular chapter and not in another might not always be very clear to the reader. This is largely because the papers that constitute the chapters of this book were not written, revised or resubmitted in the order in which they are presented here. Therefore, a chapter further on in this book might not have an element included, which was presented earlier, even though it logically could have. Our choices for which element to include in which paper were purely pragmatic, based on the revision processes and the ambition to publish. A comprehensive summary study would have solved this issue. However, such a study could not be included in the timeframe in which the research for this book and the book itself was conceived.

Finally,

chapter 8 will provide a discussion of the limitations of the research that is presented in this book. In addition, it will indicate future research directions for project control and it presents conclusions on the single- and multivariate methods for statistical project control using EVM/ES.

References

[1] P. W. Morris, The management of projects, Thomas Telford, 1997.

[2] R. Atkinson, Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria, International journal of project management 17 (6) (1999) 337–342.

[3] J. Turner, Editorial: International project management association global qualification, certification and accreditation, International journal of project management 14 (1) (1996) 1–6.

[4] J. Pollack, The changing paradigms of project management, International Journal of Project Management 25 (3) (2007) 266–274.

[5] L. Crawford, J. Pollack, Hard and soft projects: a framework for analysis, International Journal of Project Management 22 (8) (2004) 645–653.

[6] W. Belassi, O. I. Tukel, A new framework for determining critical success/failure factors in projects, International journal of project management 14 (3) (1996) 141–151.

[7] A. Munns, B. F. Bjeirmi, The role of project management in achieving project success, International journal of project management 14 (2) (1996) 81–87.

[8] R. Müller, R. Turner, The influence of project managers on project success criteria and project success by type of project, European Management Journal 25 (4) (2007) 298–309.

[9] K. Jugdev, R. Müller, A retrospective look at our evolving understanding of project success, Project management journal 36 (4) (2005) 19–31.

[10] M. Hobday, The project-based organisation: an ideal form for managing complex products and systems?, Research policy 29 (7) (2000) 871–893.

[11] D. White, J. Fortune, Current practice in project managementan empirical study, International journal of project management 20 (1) (2002) 1–11.

[12] J. Fortune, D. White, K. Jugdev, D. Walker, Looking again at current practice in project management, International Journal of Managing Projects in Business 4 (4) (2011) 553–572.

[13] D. Baccarini, The logical framework method for defining project success, Project management journal 30 (4) (1999) 25–32.

[14] Y. H. Kwak, F. T. Anbari, Analyzing project management research: Perspectives from top management journals, International Journal of Project Management 27 (5) (2009) 435–446.

[15] H. T. Ingason, H. I. Jónasson, Contemporary knowledge and skill requirements in project management, Project Management Journal 40 (2) (2009) 59–69.

[16] L. A. Ika, Project success as a topic in project management journals, Project Management Journal 40 (4) (2009) 6–19.

[17] J. Batselier, M. Vanhoucke, Construction and evaluation framework for a real-life project database, International Journal of Project Management.

[18] S. Hartmann, D. Briskorn, A survey of variants and extensions of the resource-constrained project scheduling problem, European Journal of Operational Research 207 (2010) 1–15.

[19] F. Anbari, Earned value project management method and extensions., Project Management Journal 34(4) (2003) 12–23.

[20] H. Hunter, R. Fitzgerald, D. Barlow, Improved cost monitoring and control through the earned value management system, Acta Astronautica 93 (0) (2014) 497 – 500. doi:http://dx.doi.org/10.1016/j.actaastro.2012.09.010.
URL http://www.sciencedirect.com/science/article/pii/S0094576512003670

[21] Y. H. Kwak, F. T. Anbari, History, practices, and future of earned value management in government: Perspectives from NASA, Project Management Journal 43 (1) (2011) 77–90.

[22] M. Plaza, O. Turetken, A model-based DSS for integrating the impact of learning in project control, Decision Support System 47 (2009) 488–499.

[23] H. Khamooshi, H. Golafshani, EDM: Earned duration management, a new approach to schedule performance management and measurement, International Journal of Project Management 32 (6) (2014) 1019–1041.

[24] R. Elshaer, Impact of sensitivity information on the prediction of project's duration using earned schedule method, International Journal of Project Management 31 (2013) 579–588.

[25] Ö. Hazır, A review of analytical models, approaches and decision support tools in project monitoring and control, International Journal of Project Management.

[26] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[27] M. Vanhoucke, Measuring time: An earned value performance management study, The Measurable News 1 (2010) 10–14.

[28] M. Vanhoucke, Using activity sensitivity and network topology information to monitor project time performance, Omega The International Journal of Management Science 38 (2010) 359–370.

[29] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[30] J. Colin, M. Vanhoucke, A comparison of the performance of various project control methods using earned value management systems, Expert Systems with Applications 42 (6) (2014) 3159 – 3175. doi:http://dx.doi.org/10.1016/j.eswa.2014.12.007.

[31] Q. Fleming, J. Koppelman, Earned value project management. 3rd Edition, 3rd Edition, Project Management Institute, Newton Square, Pennsylvania, 2010.
URL http://books.google.be/books?id=ZMRVngEACAAJ

[32] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega The International Journal of Management Science 49 (2014) 107–122.

[33] G. T. Bauch, C. A. Chung, A statistical project control tool for engineering managers, Project Management Journal 32 (2001) 37–44.

[34] Q. Wang, N. Jiang, L. Gou, M. Che, R. Zhang, Practical experiences of cost/schedule measure through earned value management and statistical process control, Lecture Notes in Computer Science 3966 (2006) 348–354.

[35] S. S. Leu, Y. C. Lin, Project performance evaluation based on statistical process control techniques, Journal of Construction Engineering and Management 134 (2008) 813–819.

[36] R. Aliverdi, L. Moslemi Naeni, A. Salehipour, Monitoring project duration and cost in a construction project by applying statistical quality control charts, International Journal of Project Management 31 (3) (2013) 411–423.

[37] H. Steyn, Project management applications of the theory of con-
     straints beyond critical chain scheduling, International Journal of
     Project Management 20 (2002) 75–80.

[38] S. Van de Vonder, E. Demeulemeester, W. Herroelen, R. Leus, The use
     of buffers in project management: The trade-off between stability and
     makespan, International Journal of Production Economics 97 (2005)
     227–240.

[39] W. Lipke, Speculations on project duration forecasting, The Measur-
     able News 3 (2012) 3–7.

[40] E. Demeulemeester, M. Vanhoucke, W. Herroelen, Rangen: A ran-
     dom network generator for activity-on-the-node networks, Journal of
     Scheduling 6 (2003) 17–38.

[41] Y. H. Kwak, L. Ingall, Exploring monte carlo simulation applications
     for project management, Risk Management 9 (1) (2007) 44–57.

[42] T. Williams, Towards realism in network simulation, Omega The In-
     ternational Journal of Management Science 27 (1999) 305–314.

[43] C. Loch, A. De Meyer, M. Pich, Managing the unknown: A new ap-
     proach to managing high uncertainty and risk in project, John Wiley
     and Sons, Inc., New Jersey., 2006.

[44] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshi-
     rani, The elements of statistical learning, Vol. 2, Springer, 2009.

[45] M. Vanhoucke, Measuring the efficiency of project control using ficti-
     tious and empirical project data, International Journal of Project Man-
     agement 30 (2012) 252–263.

2

Setting tolerance limits for statistical project control using earned value management

Modelling of the current state of EVM/ES project control methods and its derivatives	Development of a framework to test and compare EVM/ES project control methods	Single- and multivariate advances to top-down project control using EVM/ES

Top-down
- Rules of thumb (RoT)
- Statistical process control approaches (SPC)
- Critical Chain / Buffer Management (CC/BM) derivatives

Bottom-up
- Longest path method (LP)

- Project dataset
- Risk and variation modelling
- Two -phased Monte Carlo simulation (MCS)
- Measuring project control performance

Single
- Statistical tolerance limits (STL)

Multi
- Multivariate statistical project control (MSPC)
- Multivariate regression methods for longest path calculations (MR-LP)

Empirical perspective

Figure 2.1: Schematic overview of the elements, of the larger body of research in this book, that are discussed in chapter 2.

Abstract

Project control has been a research topic since decades that attracts both academics and practitioners. Project control systems indicate the direction of change in preliminary planning variables compared with actual performance. In case their current project performance deviates from the planned performance, a warning is indicated by the system in order to take corrective actions.

Earned value management/earned schedule (EVM/ES) systems have played a central role in project control, and provide straightforward key performance metrics that measure the deviations between planned and actual performance in terms of time and cost. In this paper, a new statistical project control procedure sets tolerance limits to improve the discriminative power between progress situations that are either statistically likely or less likely to occur under the project baseline schedule. In this research, the tolerance limits are derived from subjective estimates for the activity durations of the project. Using the existing and commonly known EVM/ES metrics, the resulting project control charts will have an improved ability to trigger actions when variation in a project's progress exceeds certain predefined thresholds

A computational experiment has been set up to test the ability of these statistical project control charts to discriminate between variation that is either *acceptable* or *unacceptable* in the duration of the individual activities. The computational experiments compare the use of statistical tolerance limits

with traditional earned value management thresholds and validate their power to report warning signals when projects tend to deviate significantly from the baseline schedule.

2.1   Introduction

Project management and control have been research topics since the development of project planning approaches such as the critical path method (CPM, [1, 2]) and the Program Evaluation and Review Technique (PERT, [3]). The majority of the research endeavors published in the academic literature focuses on the construction of a project baseline schedule within the presence of limited resource constraints (see e.g. a recent survey written by Hartmann and Briskorn [4]). Ever since, the use of a project baseline schedule has been put into the right perspective, as it only acts as a roadmap for the future project performance. If the project runs into trouble, the baseline schedule can be used to detect such problems and to trigger corrective actions. The construction of a project baseline schedule should go hand in hand with the development of a risk analysis [5] to detect the weak parts in the schedule. These two sources of information should therefore be used concurrently, both before and during the execution of the project. The need for the integration of the project baseline scheduling construction, the risk analysis and management and the project control step has been recently referred to as *dynamic scheduling* [6, 7].

In this paper, the construction of a baseline schedule is assumed to be a given, and the focus lies on the presentation of a new project control system [8]. Rozenes et al. [9] have characterized a project control system by stating that it should indicate the direction of change in preliminary planning variables when compared with actual performance. If there is an *unacceptable* gap between planning and performance, a warning should be generated by the system in order to take corrective action. In the remainder of this text, we will further elaborate on this *unacceptable* gap and quantify how it can be specified.

This study mainly focuses on the generation and interpretation of warning signals during project control and hence it does not include results in the quality and input of corrective actions, nor on the use of these new methods for contingency planning.

A well-known and widely used project control system is Earned Value Management (EVM) originally developed in the '60s at the U.S. Department of Defense as a standard method to measure a project's performance. This system relies on a set of straightforward metrics to measure and evaluate the general health of a project. These metrics serve as early warning

signals to timely detect project problems or to exploit project opportunities. EVM has been investigated and used widely in project management practice and has been brought into the attention of researchers by Fleming and Koppelman [10]. Recently, the research on EVM has produced the earned schedule method (ES, [11]) and a large study on the integration of risk management and EVM/ES was conducted [12].

EVM/ES calculations are based on the costs and durations for the activities in a project. Both in the planning phase and during the execution of the project, some detail of the individual activities is lost as the individual performance is summed up into the performance metrics observed and controlled at the project level. Earned value management is known to have some shortcomings when it comes to protecting the critical path in the schedule [10]. The summation of activity earned value into the aggregate level for the project does not incorporate schedule risk analysis (SRA, [13]) data, such as criticality. In spite of the complexity of this problem, EVM/ES use in practice is characterized by decision making from practical experience, rules-of-thumb and anecdotal evidence. In the academic literature, it is reasoned that the application of a statistical process control approach could benefit the EVM/ES project control process. These previous attempts depend on the availability of historical records that may or may not exist and may or may not be valid. We therefore also allow the use of subjective estimates.

In this paper, a new *statistical project control* approach is presented. It is inspired by the practical format of *statistical process control charts* and based on *tolerance intervals* produced by a simulated sample. A controlled simulation experiment will be used to specifically produce two types of project progress situations. First, project progress situations where the variation from plan is limited to some *acceptable* margin are used to produce project control charts. Secondly, we test the discriminative power of these control charts when the variation from plan exceed the pre-defined *acceptable* margin.

The outline of this paper is as follows. In section 2.2, we present a literature review on the use of statistical control principles in project management and control. We will present two project control charts, the X chart and R chart. Section 2.3 illustrates these control charts on a fictitious project example. In section 2.4, the design of the computational test experiments is outlined based on various sources from literature. In section 2.5, test results obtained by computational experiments using Monte Carlo simulation to imitate a fictitious project progress environment are discussed. Section 2.6 draws general conclusions and highlights future research avenues.

2.2 Statistical Project Control

2.2.1 Literature overview

This literature review introduces the main concepts of *statistical process control* (SPC) as they are applied to monitor and control processes in services and manufacturing. The application of SPC techniques to earned value management is not new in literature. We introduce a *statistical project control* procedure that aims to overcome some of the issues with the previously published approaches, but which is not an implementation of SPC in the strict sense, although it reproduces some of its concepts and nomenclature. The last paragraph of this literature review is dedicated to a summation of the differences between our statistical project control procedure and the implications of applying SPC to project control in a strict sense.

In standard SPC applications a state of control is identified with a process generating samples for which the subgroup averages are approximately normal under the central limit theory. Control charts such as the Shewhart, cumulative sum (CUSUM) and exponentially weighted moving avarage (EWMA) charts serve as on-line procedures to monitor process stability, to detect assignable variation or to forecast process movements in industrial processes [14]. A process is said to be in-control when only common cause variation is present. This type of variation is characterized as coming from phenomena constantly active in a process, which can be predicted probabilistically. In his original work on process control, Shewhart [15] introduced the term chance cause. A process is said to be out-of-control if a second type of variation is present known as assignable cause variation. Assignable cause variation arises when a new, previously unanticipated phenomenon is present in the system and should cause a signal.

Figure 2.2 illustrates how a control chart applied to project control might be interpreted. Periodic observations of a performance measure for a fictitious project execution are outlined, along with an illustrative upper control limit (UCL) and lower control limit (LCL). For illustrative purposes, we will assume that the performance measure is of the type where a high value is good and a low value acts as a trigger for actions due to potential problems. The frequently encountered schedule performance index (SPI, section 2.2.3) and schedule performance index using earned schedule (SPI(t), section 2.2.3) measures in EVM/ES are of this type, although control charts of the opposite type have also been proposed where the natural logarithm of the reciprocal of SPI or SPI(t) is taken [16]. In case the observations fall outside the project control limits, the charts report a signal which could trigger actions to bring the project back to plan or to exploit opportunities. More precisely, when an observation falls below the lower control limit, the performance has likely dropped below the pre-defined *ac-*

ceptable margin on the baseline estimates. In this case, this signal should be interpreted as a warning signal to the project manager to consider taking corrective actions. When the performance indicator exceeds the upper control limit, it can be seen as a signal to exploit opportunities, in which case the current project schedule might be re-baselined to incorporate this opportunity. Different SPC control charts have been implemented for project control using earned value management by [16–21]. A short overview of their respective approaches and findings is listed below.
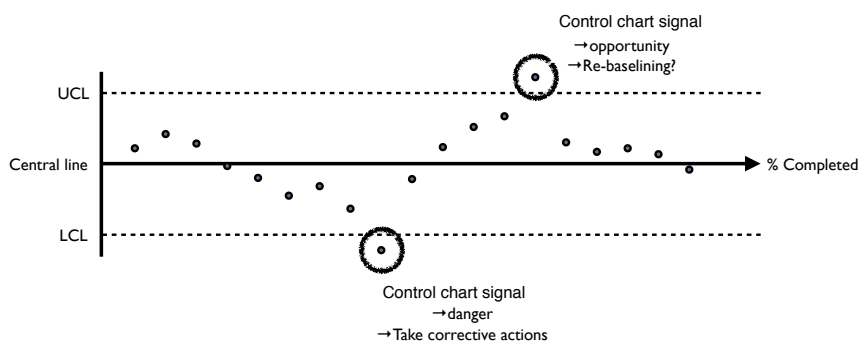
Figure 2.2: Illustrative control chart and out-of-control indications

Lipke and Vaughn [17] apply a XmR chart (individuals and moving range chart) and calculate the control limits based on a two-observation sample of earned value performance metrics. They use the reciprocal measures of CPI (cost performance index) and SPI (schedule performance index), as they have shown that these are normalized, such that many of the issues of applying SPC, such as variability and homogeneity of the data, are avoided. Bauch and Chung [18] modify the statistical process control charts proposed by Shewhart to be used with earned value management considering three major aspects. Firstly, they asses single observations relative to historical project data from 20 similar projects. Secondly, as projects span a finite time horizon, which is typically different for each project, they normalize the different time lengths of each project into a consistent number of time periods, chosen as 20 discrete measurements for the purpose of the paper. As a third modification, they adopted the CUSUM approach [22] to incorporate the increasing nature of period-to-period measurements of project performance. The typical y-axis of control charts is transformed to plot cumulative project performance values. As a result, the central line and upper and lower limits are non-stationary. Wang et al. [19] illustrate the use of SPC charts using EVM/ES on a set of more than 30 software projects where abnormal progress situations are effectively detected. [20] combine XmR charts with EVM/ES performance metrics, but implement the logarithm of the reciprocal of SPI and the cost performance index (CPI).

This was earlier proposed by Lipke [16]. Control limits are calculated from historical data of 120 projects performed by a management consulting company. Leu and Lin [20] introduce the division of different cases based on historical factors influencing project performance and quantify the XmR chart procedure for different trends observed in their data. Moreover, they accurately describe how "the quick proliferation and complexity of project performance data indicate the need for a well-organized project performance analysis process." Aliverdi et al. [21] suggest to transform CPI data when quality control charts are applied for cost control. SPC charts for both individual and moving range measurements were successfully implemented. They conclude that the integrated approach of EVM/ES and statistical quality control charts can contribute to a better and more reliable project control process.

Overall, previous research on the integration of EVM/ES with SPC techniques for projects has established that it improves project control by providing an objectively based and easily implemented real-time monitoring system. The evidence provided in the literature is based on empirical research of either post-hoc statistical analyses on projects [19] or the implementation of control limits derived from historical data [16, 20]. The dependency on historical data does not pose an immediate problem for statistical project control approaches, ideally the use of historical records is preferred over subjective estimates for project control [23]. However, the concept of "similarity" between projects, is often vaguely defined. When historical EVM/ES data is directly applied, the project-specific EVM/ES dynamics should not be ignored. [24] illustrated these project-specific dynamics with EVM/ES forecasting becoming less reliable for projects with a network close to a parallel network structure. When the concepts of SPC are directly translated to incorporate EVM/ES measurement, an assumption with respect to their distribution has to be made. In previous literature this prerequisite was fulfilled by transforming the data to appear normally distributed [17, 19] or lognormally distributed [16, 20] or appeared overlooked [21].

In section 2.2.3 we propose a different statistical project control procedure that will also include the appealing aspects of using control charts. They are easy to set up, implement and interpret. However, the imposed tolerance limits on the earned value metrics should be interpreted in the more technical sense [25]. The tolerance limits are acquired from the empirical distribution function (edf) of the EVM/ES metric under study at different points along the life-cycle of a project. This edf is obtained from a Monte Carlo simulation model that allows variation to be added at the activity level of the project. This variation can be introduced as subjective estimates of what deviations are statistically plausible during the execution of the project or from calibration with historical in-company data.

	statistical process control for project control	**statistical project control setting tolerance limits**
Data type	Process data	Simulated data
Data measurements	Data collected over time	Data collected using simulations
Number of data points	Small sample size per time unit	Large sample size per simulation point
Process	Infinite process	Finite process
Analysis	Ex post	Ex ante
Control limits	Deviations from normal progress	Deviations from desired progress
Reference value	Normal project progress (with normal variation)	Baseline schedule (with *acceptable* tolerances)

Table 2.1: Difference between SPC for projects and our statistical project control procedure

Table 2.1 summarizes the main differences between our statistical project control procedure that uses tolerance limits and a classical SPC implementation for project control. The main aspects are given along the following lines.

- *Data input*: A first fundamental difference lies in the data input necessary to construct the control limits. While process data needs to be collected at regular time intervals to monitor the behavior of a process in progress when classical SPC is implemented, our control charts rely on inputs of estimates that express the desired tolerance on the activity durations or allows calibration to historical data at the activity level. Consequently, these inputs are used to express an ex ante desired state of outcome on activity durations rather than an observed ex post outcome in the measured variable. For this reason, the activity distribution inputs can and probably will be different than the real activity distributions, and the tolerance limits are set up to detect these differences and to receive timely warnings when real activity durations exceed the desired thresholds. Unlike classical applications of SPC to project control, which aim at detecting deviations from the average on-going state of the measurable variable in progress, the project control charts using tolerance limits aim at detecting deviations from a predefined desired state in the variable defined before the start of the progress.

- *Dynamic project performance measurements*: Given the ex ante approach of setting tolerance for project control, the procedure requires a large sample of data points to predict the impact of the *acceptable* activity tolerances on the tolerance limits set for EVM/ES. Therefore, the data is collected using Monte-Carlo simulations using the desired tolerances on the activity distributions and measuring their impact on the EVM/ES output metrics such as SV, SPI, SV(t) and SPI(t). Consequently, these output metrics are used to construct tolerance limits given the desired input parameters. This is in high contrast to the on-going data collection method of observed process data used to construct control limits to detect deviations from normal or average process behavior in a classical application of SPC.

- *Control limits*: Due to the fundamental difference between the control limits of SPC for project control and the statistical tolerance limits based on the desired activity duration tolerances of our project control approach, we have decided to use the words tolerance limit rather than control limit during the construction of the control charts. Consequently, the control limits set on the EVM/ES output metrics SV, SPI, SV(t) and SPI(t) are used to define the thresholds on the tolerances in project performance behavior given the desired tolerances on the activity durations and hence act as warning signals to mea-

sure *unacceptable* project progress behavior compared to the baseline schedule. This is exactly what an EVM/ES system is trying to accomplish and this view is consistent with the definition of a project control system [9]. Project based performance metrics along the project progress are used to act as an action threshold to detect underlying problems that are responsible for the deviation between planned outcome and project progress outcome. The tolerance limits used in this paper define these action thresholds based on a statistical analysis of the simulated data resulting from a pre-specified allowable variation.

2.2.2   A new statistical project control approach

Our statistical project control approach differs from those described in the literature overview by setting tolerance limits. These tolerance limits are calculated by means of simulation and this allows us to redefine and adapt the process control principles regularly found in manufacturing and services to our requirements. Process control is generally an ongoing process where little in-depth information is known for the process, and a state of statistical control is inferred from the characteristics of the recorded process variables. Our approach, shares the use of control charts, but is fundamentally and functionally different, as outlined in the last paragraph of section 2.2.1.

Practically, these differences brings about two main aspects:

- The statistical control procedure for a project will be finite in time. This follows directly from the definition of a project as a one-time endeavour with a clearly defined start and finish. The control chart will therefore only allow a predefined number of measurements at distinct time-instances.

- Simulation allows us to apply a state of control, such as the user deems appropriate, and characterize the resulting EVM/ES project progress variables. Historical data can play a role in the input modelling of the simulation, but is no longer directly required to calculate the control limits. Historical records of EVM/ES may be used to evaluate the congruence of the simulation output with real-life executions.

The main research interest of this article is to develop and test a new project control approach for schedule control. Since schedule control can be expressed in the deviations between the actual durations and the planned durations of the activities of the project, these will be the primary focus of our simulation model. The simulation model will add uncertainty to the durations of the activities, which will result in fictitious project executions that deviate from the baseline. An effective statistical project control

chart is required to differentiate between the variation that is *acceptable* for an activity and variation that is *unacceptable*. If an activity has a deviation between its real duration and its planned durations that is unacceptably large, we wish to detect this on a project control chart using EVM/ES. If all the activity durations lie within pre-defined *acceptable* limits, we will consider the project to be executed *as planned*, and the control charts should not generate any signal.

Our approach allows the user to set tolerances on the activity level, which will result in tolerance limits for project progress variables to be used at the project level in control charts. The tolerance limits for the project progress variables will be calculated in what is referred to as the *static phase* of our procedure.

When actual project progress is recorded during the execution of a project, the observations can be plotted against the tolerance limits on the project control charts. This phase is referred to as the *dynamic phase*. In section 2.5, we will quantify the performance of the tolerance limits in the control charts using a second large Monte Carlo simulation.

2.2.3   Project control charts

The static phase

Determine the as planned situation   If the project in progress exhibits an abnormal behavior compared to a predefined situation, the project manager must be able to timely detect this. Consequently, the normal project progress situation needs to be defined by the project manager in order to construct control charts that distinguish between *as planned* and *not as planned* situations. Therefore, it is assumed that the project baseline schedule is a forecast of what might happen in the future during project progress and determines the *as planned* situation, subject to some *acceptable* variation on the activity durations and possible project-wide effects of uncertain events. This is exactly the starting point of dynamic scheduling, which proposes that the project baseline schedule needs to be used to measure the project progress. Both positive and negative deviations within a certain range are assumed to be inherent to any project and are considered as normal while abnormal deviations exceeding certain thresholds should trigger further investigations.

This variation on the activity durations, and additional uncertain events, obviously have an impact on the project performance which might differ from the initially expected project performance, as stipulated in the project baseline schedule. This *as planned* project performance deviation will be measured by Monte Carlo simulation runs.

Monte Carlo simulations   Monte Carlo simulations in project manage-
ment and control have been used widely in literature. Since its introduc-
tion [26] to analyze project networks, the methodology has been used for a
range of project management research projects and applications. In [27], a
classified bibliography of project risk management research is given where
simulation plays a central role. Ever since, Monte Carlo simulation studies
have been a respected methodology in project management and schedul-
ing. The simulation studies in this paper are based on the project control
simulation studies described in [28, 29]. The methodology used has been
initially developed by Vanhoucke and Vandevoorde [24] and is formalized
in a dynamic scheduling simulation tool, P2 Engine [30].

In the current paper, the simulation runs will be used to simulate fictitious
project progress in order to obtain periodic project performance data using
the well-known project performance indicators obtained by the EVM/ES
key metrics. Consequently, fictitious project progress is simulated for a
number nrs of Monte Carlo runs. Variation in the activity durations will
have an impact on the project performance over time and therefore, the
variable X_{sp} will be used to denote the performance indicator value from
simulation run s ($s = 1, \ldots, nrs$) measured at a certain moment in time
between the start and the finish of the simulated project progress when it
is p percent completed ($p = (1, \ldots, P) \times \Delta PC$). A total of P observations
are taken with increment ΔPC.

The following performance indicators are periodically measured during
the simulation runs

SV	Schedule variance (SV = EV - PV)
SPI	Schedule performance index (SPI = EV / PV)
SV(t)	Schedule variance using earned schedule (SV(t) = ES - AD)
SPI(t)	Schedule performance index using earned schedule (SPI(t) = ES / AD)

with

PV	Planned value
EV	Earned value
AC	Actual cost
ES	Earned schedule
AD	Actual duration
BAC	Budget at completion
PC	Percentage project completion (PC = EV / BAC) ($0\% \leq PC \leq 100\%$)
P	Number of observations taken, with an increment ΔPC

For the sake of completeness, we even include SV and SV(t) in our re-
search. Previous EVM/ES studies using SPC often neglected these perfor-
mance metrics and thus very few empirical evidence for their distributions
is known. Since we do not assume any distribution for the control mea-

sures a priori, we can incorporate them into our research. Note that, we do not incorporate the cost control metrics cost variance (CV = EV - AC) and cost performance index (CPI = EV / AC). In our simulation model, cost is always strongly correlated with the time-performance of the activity, which might not necessarily be the case in real-life. Since our focus is on controlling the time performance of a project, we restrict the analysis to those metrics specifically designed to do so.

Add tolerance limits   The simulated data is used to construct control limits for the project control variables $X_p =$(SV, SV(t), SPI, SPI(t)) at a review period p for two distinct types of control charts. The proposed X chart and the R chart resemble the Shewhart individuals (indX) and moving range (mR) control charts in their ability to inspect individual observations [31]. Control limits for the Shewhart charts are constructed based on the moving average over the range (i.e. the difference between an observation x_i and its predecessor x_{i-1}), calculated using the previous $m-1$ observations.:

$$\widehat{\text{MR}} = \frac{1}{m-1} \sum_{i=2}^{m} |x_i - x_{i-1}| \tag{2.1}$$

This concept of moving range is not adopted here to construct tolerance limits for project control, since it implies that observations are taken for a consistently operating process. On the contrary, we assume that our nrs simulation runs provide sufficient observations to reconstruct the distribution function for the variable X_p at a review period $p \in (1, \ldots, P) \times \Delta\text{PC}$. Note that we do not assume two random variables X_j and X_k (where $j, k \in (1, \ldots, P) \times \Delta\text{PC}$) to be independent, but we will construct tolerance limits for each of them independently. For any two adjacent review periods, we will look at the range $R_j = |X_j - X_{j-1}|$ ($j \in (1, \ldots, P) \times \Delta\text{PC}$) and treat this variable as a new project performance variable.

- The X chart is used to monitor individual observations for the project performance variables X_p at a review period p. Tolerance limits can be calculated from the empirical distribution function $\hat{\text{F}}_{X_p}(t)$.

$$\hat{\text{F}}_{X_p}(t) = \frac{1}{nrs} \sum_{s=1}^{nrs} \mathbb{1}\{X_{sp} \leq t\} \tag{2.2}$$

derived from a sample of independent identically distributed (i.i.d.) variables $(X_{1p}, \ldots, X_{nrs\,p})$ taken at p. $\mathbb{1}\{A\}$ represents the indicator function of an event A:

$$\mathbb{1}\{A\} = \begin{cases} 1 & \text{if } A \text{ is TRUE} \\ 0 & \text{if } A \text{ is FALSE} \end{cases} \tag{2.3}$$

By the strong law of large numbers, we know that the emperical distribution function (2.2) asymptotically converges to the real cumulative distribution function $F(t)$ for all possible values of t.

The lower and upper statistical tolerance limits at a review period p for a level α ($LTL^\alpha_{X_p}$ and $UTL^\alpha_{X_p}$) are then calculated as the αth and $(1 - \alpha)^{\text{th}}$ quantile of the distribution $F(t)$. The quantiles can be calculated from the inverse of the empirical distribution function or by estimation of the sample quantiles. Let $\hat{Q}(\alpha)_p$ for $0 < \alpha < 1$ denote the sample quantile based on a a set of independent observations $\{X_{1p}, X_{2p}, \ldots, X_{nrs\,p}\}$ from the distribution F at review period p. If the order statistics of $\{X_{1p}, X_{2p}, \ldots, X_{nrs\,p}\}$ are denoted as $\{X_{(1)p}, X_{(2)p}, \ldots, X_{(nrs)p}\}$, then $\hat{Q}(\alpha)_p$ can be estimated as:

$$\hat{Q}(\alpha)_p = (1 - \gamma)X_{(j)p} + \gamma X_{(j+1)p}$$
$$\text{where} \quad \frac{j - \alpha + 1}{nrs} \leq \alpha < \frac{j - 2 + \alpha}{nrs} \qquad (2.4)$$

with $\gamma = \alpha nrs - \alpha - j + 1$ and $j = \lfloor \alpha nrs - \alpha + 1 \rfloor$. $\lfloor u \rfloor$ is used to represent the largest integer smaller than u. We refer to [32] for a detailed comparison of different functions to calculate continuous sample quantiles.

The lower and upper statistical tolerance limits at a review period p for a level α for the X chart are:

$$LTL^\alpha_{X_p} = \hat{Q}(\alpha)_p$$
$$UTL^\alpha_{X_p} = \hat{Q}(1 - \alpha)_p \qquad \forall p \in (1, \ldots, P) \times \Delta\text{PC} \qquad (2.5)$$

The value for the level α can be chosen subsequently with respect to a total level of control over al P review periods of the project. We will illustrate this in section 2.5.1.

- The R chart monitors the range R (or difference) between two adjacent observations. This results in tolerance limits for a vector of observations which is in length one less than those for the X chart. The empirical distribution function $\hat{F}_{R_p}(t)$ for R at a period p can be found using nrs simulation runs.

$$\hat{F}_{R_p}(t) = \frac{1}{nrs} \sum_{s=1}^{nrs} \mathbb{1}\{R_{sp} \leq t\} \qquad (2.6)$$

with:

$$R_{sp} = |X_{sp} - X_{sp-1}| \qquad \forall p \in (2, \ldots, P) \times \Delta\text{PC}$$

and the tolerance limits for a level α are:

$$LTL_{R_p}^{\alpha} = \hat{Q}(\alpha)_p$$
$$UTL_{R_p}^{\alpha} = \hat{Q}(1-\alpha)_p$$

$$\forall p \in (2, \ldots, P) \times \Delta\text{PC} \qquad (2.7)$$

where $\hat{Q}(\alpha)_p$ is calculated with the order statistics $\{R_{(1)p}, R_{(2)p}, \ldots, R_{(nrs)p}\}$. The intuitive explanation for the use of the range chart is that the difference between two measurements of project performance relates to the instantaneous change of the performance. Since a project consists of activities that begin and end at discrete time instances, the transformed variable R_p should better reflect the performance of activities that are being executed at that time instance and the project control process is less cluttered by the performance of past activities.

The dynamic phase

A project needs to be monitored by periodically measuring its performance. Each periodic project performance observation can be plotted on the control chart. Y_p is used to refer to the performance indicator value during project progress at an instance p. It is different from X_{sp} in the sense that it is not a priori assumed to be drawn from the distribution of X_{sp}. Rather, the calculated tolerance limits for the control charts are used to asses whether a value Y_p is likely to have been produced under the null hypothesis (characterized by the empirical distribution function of X_p in the simulation of the *static phase*).

Classical SPC applications often reject the notion that control charts are a form of hypothesis testing. However, using tolerance limits for our statistical project control approach, we have taken the liberty to present our control charts in such a way. Control charts and hypothesis tests can be regarded as alternatives and the analogy is as presented below:

The null and alternative hypotheses of the project schedule control procedure proposed in this paper are

H_0: The project is executed *as planned*

H_a: The project is executed *not as planned*

This hypothesis is tested using periodic control charts, by checking whether the current project performance Y_p lies within the control limits or not, as follows

Accept H_0: $LTL_{X_p}^{\alpha} \leq Y_p \leq UTL_{X_p}^{\alpha}$

Reject H_0: $(Y_p < LTL_{X_p}^{\alpha}) \vee (Y_p > UTL_{X_p}^{\alpha})$

for the X chart and

$$\text{Accept } H_0: LTL_{R_p}^{\alpha} \leq |Y_p - Y_{p-1}| \leq UTL_{R_p}^{\alpha}$$
$$\text{Reject } H_0: (|Y_p - Y_{p-1}| < LTL_{R_p}^{\alpha}) \vee (|Y_p - Y_{p-1}| > UTL_{R_p}^{\alpha})$$

for the R chart.

2.3 Illustrative example

This section illustrates the calculations to construct statistical project control charts based on a fictitious project network example. Figure 2.3 displays the fictitious illustrative project network from [12] with 10 non-dummy activities. The numbers above each node are used to display the predefined baseline duration (in days) while the number below each node denotes the baseline total activity cost.
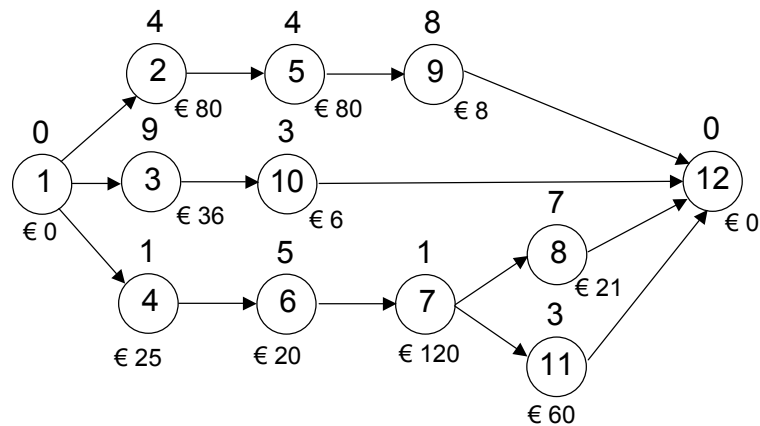
Figure 2.3: An illustrative project network (Source: [12])

Table 2.2 displays five simulation runs ($nrs = 5$) where the activity durations have been randomly chosen from a uniform distribution with a maximum relative deviation from the baseline duration equal to 30%. The resulting limits for the fictitious activity durations were rounded to the nearest integer and are shown in the first row of table 2.2. The last column of this table presents the total duration (RD) of the illustrative project for these simulation runs.

Table 2.3 displays the earned value metric EV along the project duration (in days). It is assumed that all activities follow a linear earned value accrue and, consequently, variation in an activity duration has a linear impact on the actual activity cost.

Activity	2	3	4	5	6	7	8	9	10	11	RD
$c \pm \delta^{cv}$	3-5	6-12	1-1	3-5	4-6	1-1	5-9	6-10	2-4	2-4	
1	3	8	1	5	4	1	9	6	2	4	15
2	5	6	1	5	6	1	7	6	3	3	16
3	4	7	1	4	5	1	6	7	2	3	15
4	4	12	1	3	5	1	9	8	4	2	16
5	3	11	1	3	4	1	8	10	2	4	16

Table 2.2: Normal activity variation (in days) from five simulation runs

In practice, EV measurements will be collected as imposed by contract specifications or at distinct time instances. It is easily understood that such time-instances will not always coincide with the discrete percentage complete points for which tolerance limits for control charts are calculated. This is also not the case for the project runs in our simulation software. If data are collected at fixed time-intervals, the measurements along the time-axis need to be transformed towards measurements equally spaced along the percentage complete axis. The last row of table 2.3 displays the planned value along the project duration. The transformation towards measurements equally spaced along the percentage complete axis is done by means of interpolation.

Interpolation might introduce additional variance in the observations made in both the *static phase* and the *dynamic phase*. However, this error-term can be kept within reasonable value if both the increment in percentage complete ΔPC and the periodic intervals at which data is collected are chosen sufficiently small. We illustrate this procedure with the SPI data obtained for percentage completion value of $p = \{10, 20, 30, 40, \ldots, 90\}$ for the first row of table 2.2. The SPI measurements for this fictitious project example are obtained from the data recorded for each day of table 2.3 by dividing the EV of the first row by the PV, as is illustrated in table 2.4. In order to find the SPI value for this fictitious project execution at e.g. 50% of the project complete, we need to look for the data we collected that have PC values lying the closest to 50%: in this case our measured SPI after $t_A = 5$ and $t_B = 6$ days. To approximate SPI a simple calculation through linear interpolation shows that:

$$SPI_{p=50\%} = SPI_{t_A} + (SPI_{t_B} - SPI_{t_A})\frac{50 - PC_{t_A}}{PC_{t_B} - PC_{t_A}} \qquad (2.8)$$

$$= 1.12 + (1.69 - 1.12)\frac{50 - 39.5}{70.2 - 39.5} \qquad (2.9)$$

$$= 1.31 \qquad (2.10)$$

These calculations result in the SPI measurements found in table 2.5.

Scenario	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	56	92	129	154	180	320	358	396	417	439	443	446	450	454	456	
2	47	72	98	123	148	174	195	333	374	413	437	442	446	450	455	456
3	50	79	107	136	164	193	221	364	392	416	441	446	450	455	456	
4	48	75	102	129	163	196	346	382	419	425	431	438	442	447	452	456
5	55	90	105	160	195	225	349	385	422	429	436	442	448	452	455	456
PV	49	77	105	133	161	189	333	380	408	434	440	446	450	454	455	456

Table 2.3: The Earned Value (EV) and Planned Value (PV) along the project duration

Time	1	2	3	4	5	6	7	8
EV	56	92	129	154	180	320	358	396
PC	12.3	20.2	28.3	33.8	39.5	70.2	78.5	86.8
PV	49	77	105	133	161	189	333	380
SPI	1.14	1.19	1.23	1.16	1.12	1.69	1.08	1.04

Time	9	10	11	12	13	14	15	16
EV	417	439	443	446	450	454	456	456
PC	91.4	96.3	97.1	97.8	98.7	99.6	100	100
PV	408	434	440	446	450	454	455	456
SPI	1.02	1.01	1.01	1	1	1	1	1

Table 2.4: The SPI along the first fictitious project execution's duration

PC	10%	20%	30%	40%	50%	60%	70%	80%	90%
1	1.11	1.19	1.21	1.13	1.31	1.5	1.69	1.07	1.03
2	0.96	0.93	0.92	0.79	0.66	0.76	0.85	0.91	0.95
3	1.02	1.03	1.02	1.02	0.67	0.77	0.87	0.96	0.96
4	0.98	0.97	0.98	1.03	1.04	1.04	1.04	1.02	1.03
5	1.1	1.16	1.12	1.21	1.19	1.14	1.08	1.03	1.02
$\overline{X_p}$	1.03	1.05	1.05	1.03	0.98	1.04	1.11	1.00	1.00
$LTL_{X_p}^{10\%}$	0.97	0.95	0.94	0.88	0.66	0.76	0.86	0.93	0.95
$UTL_{X_p}^{10\%}$	1.11	1.18	1.17	1.18	1.26	1.36	1.45	1.05	1.03

Table 2.5: The SPI for PC intervals after interpolation

2.3.1 X chart

To calculate the tolerance limits for the X chart, we need to find the first and last α quantiles for X_p at all $P = 9$ instances. This illustrative example only includes the SPI as a control variable. For more details regarding the calculations of the other earned value metric (SV) and the earned schedule metrics (SPI(t) and SV(t)), we refer the reader to the detailed work by Vanhoucke [12] and the original work on earned schedule by Lipke [33].

Figure 2.4 illustrates the tolerance limits obtained for an X chart for SPI. The central line (CL) is added for illustrative purposes and is the mean SPI along the project ($\bar{X}_p$), the outer lines represent the upper and lower tolerance limits ($LTL^\alpha_{X_p}$, $UTL^\alpha_{X_p}$, for $\alpha = 10\%$) and the dots illustrate how measurements made on a new run (with durations that exceed the predefined 30% margins) are categorized as *not as planned* as soon as its value exceeds a tolerance limit.
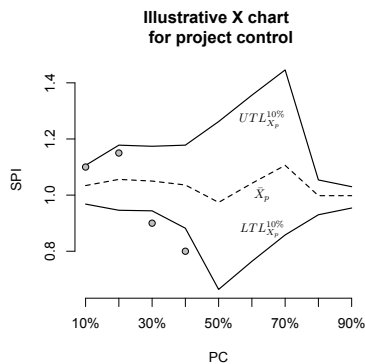
Figure 2.4: X chart control limits for the illustrative example
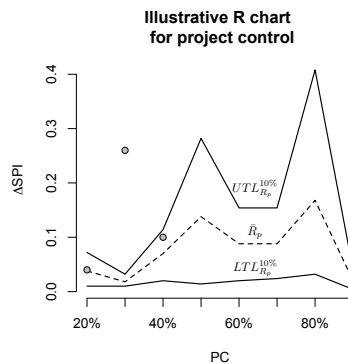
Figure 2.5: R chart control limits for the illustrative example

2.3.2 R chart

The R chart investigates the difference between two consecutive measurements. To illustrate the procedure, the SPI values from table 2.5 are transformed into the ΔSPI values of table 2.6. Afterwards, the sample mean ($\overline{R}_p$) and the quantiles are calculated from table 2.6. Figure 2.5 depicts the R control chart with the same run outlined as figure 2.4. This illustrates how both charts can show completely different dynamic behaviour of the calculated control limits, but can be read and processed in entirely the same way.

PC	20%	30%	40%	50%	60%	70%	80%	90%
1	0.08	0.02	0.08	0.18	0.19	0.19	0.62	0.04
2	0.03	0.01	0.13	0.13	0.1	0.09	0.06	0.04
3	0.01	0.01	0	0.35	0.1	0.1	0.09	0
4	0.01	0.01	0.05	0.01	0	0	0.02	0.01
5	0.06	0.04	0.09	0.02	0.05	0.06	0.05	0.01
$\overline{R_p}$	0.04	0.02	0.07	0.14	0.09	0.09	0.17	0.02
$LTL_{R_P}^{10\%}$	0.01	0.01	0.02	0.01	0.02	0.02	0.03	0
$UTL_{R_P}^{10\%}$	0.07	0.03	0.11	0.28	0.15	0.15	0.41	0.04

Table 2.6: The ΔSPI for PC intervals after interpolation

2.4 Research methodology and experimental test design

In this section, the data generation process to create a project benchmark set, the methodology used, as well as the main research questions set during the computational experiment in order to validate the project control approach are explained along the following subsections.

2.4.1 Project data

The X and R project control charts are tested on a set of 900 fictitious projects generated by a project generator RanGen [34, 35] that has been used previously in project scheduling and control studies. Vanhoucke [28] uses this dataset to connect network topology information to the optimal control procedure found to monitor the time performance of a project.

The 900 projects are chosen in accordance to the serial/parallel (SP) indicator originally proposed by Vanhoucke et al. [36] and used in previous project control simulation studies [24, 28, 29]. The SP indicator measures how close a project matches a complete serial or parallel project and is based on the maximal progressive level concept of Elmaghraby [37]. If m is the maximal progressive level, it is defined as the maximum number of activities lying on a single path in the project. If n is the number of activities in the project, it is clear that for a serial project $m = n$. The $SP = (m-1)/(n-1)$ then equals 1 for a serial project. If the largest path in a project consists of only 1 activity, the project network is said to be parallel and $SP = 0$. The project dataset consists of 100 projects for 9 intermediate SP values between 0 and 1 ($SP = \{0.1, 0.2, \ldots, 0.9\}$). If conclusions are formulated regarding a mean value over all 900 projects, it is conjectured

that they are general enough to extrapolate over all possible project networks. In the results section 2.5, we did not find a significant influence of such serial/parallel network structure on the performance of our control chart to monitor the schedule performance of a project. This result shows that our statistical project control approach provides a more robust solution than regular use of EVM/ES, which was found to improve notably for more serial project networks [28].

2.4.2 Methodology

Monte Carlo simulations are used in the *static phase* of our statistical project control procedure to construct the tolerance limits used in the X and R charts, as well as to emulate real project executions in the *dynamic phase* where fictitious project progress will be classified using these charts. A distinction is made between both phases in the form of the variation present at the activity level.

In order to generate fictitious project executions in both the *static* and *dynamic phase* of the Monte Carlo simulation, we produce activity durations under interactive and compound uncertainties [38]. Therefore, two sources of uncertainty are applied. In section 2.4.2, variation is modelled as probability distributions that are applied to the duration of the project's activities. This variation will result in fictitious project executions that are either *as planned* or *not as planned*. In section 2.4.2, we discuss the potential dependencies between a project's activities and how project progress can be affected by uncertain events.

For the construction of the tolerance limits we will allow only variation that is *acceptable* at the activity level. The projects are executed *as planned*, which represents the desired state of schedule control in the presence of uncertain project-wide events. In the *dynamic phase*, fictitious project progress is generated for which it is possible that the durations of the activities exhibit *unacceptable* variation. These fictitious project executions are then used to quantify the performance of the proposed control charts in their ability to accurately categorize whether the plan is still adequate or not. Consequently, the project control charts are used to test whether the fictitious project executions are being executed *as planned* or *not as planned*.

Project control input modelling

Since the occurrence of a project management simulation model in [39], the question has been asked how to accurately represent the uncertainty present in the duration of activities in a real-life project [40]. Since the PERT-beta distributions have been found to fall short, different authors

have suggested alternatives. Kuhl et al. [41] proposed the generalized beta distribution [28, 29]. Trietsch et al. [42] theoretically support a claim to use the Parkinson distribution, while the lognormal distribution [43], a mixture of beta and uniform distributions [44] and the doubly truncated normal distribution [45] have also been applied to study stochastic activity networks.

For the purpose of our research we have chosen to use a uniform distribution. However, the presented methodology can be reproduced with any distributional input that is able to accurately represent the stochastic nature of project activity durations. Moreover, we strongly advise a calibration to historical records for the input modelling, if our methodology would be applied in a practical setting. The uniform distribution reflects an absence of interest in the precise distributional characteristics of the project activities by the project manager. In this view on schedule control, the only interest is in the relative deviation of a duration from its baseline estimate. Most of the simulation experiments will incorporate a linear association variable (see section 2.4.2) or have *not as planned* variation added to them. Therefore, the resulting distribution for $d_i/\hat{d}_i$, where $\hat{d}_i$ represents the baseline estimate for the duration of activity i, will most likely not be uniform. Section 2.5 has a paragraph allocated to the validation of our simulation model using the recently published lognormal core theory [42]. Moreover, a reasonable fit to the proposed generalized beta distribution [41] was also found during the data analysis process. We give more details concerning the parameters used to simulate activity durations in the remainder of this section 2.4.

As planned project progress   Variation is added to the baseline duration $\hat{d}_i$ of an activity i in the project by a uniform distribution. We define the *as planned* project schedule executions in the *static phase* using the lightgrey probability distribution functions presented in figure 2.6, characterised by a choice for the maximum allowed deviation from the baseline estimate duration δ_{ap}. This distribution is presented relative to the baseline duration. The Monte Carlo samples that are drawn from this distribution therefore need to be multiplied with the baseline duration $\hat{d}_i$, in order to model the duration d_i of an activity i in a fictitious execution of the project. The maximal relative deviation δ_{ap} from the baseline estimate duration represents the subjective estimate of the variation on the baseline schedule that is regarded *as planned*.

Detect both opportunities and dangers   In the *dynamic phase*, project progress situations are simulated for which the baseline schedule might no longer be adequate. In order to simulate this, variation is added using a *not as planned*

**Uniform probability
distribution functions**

low %A_{np} high %A_{np}
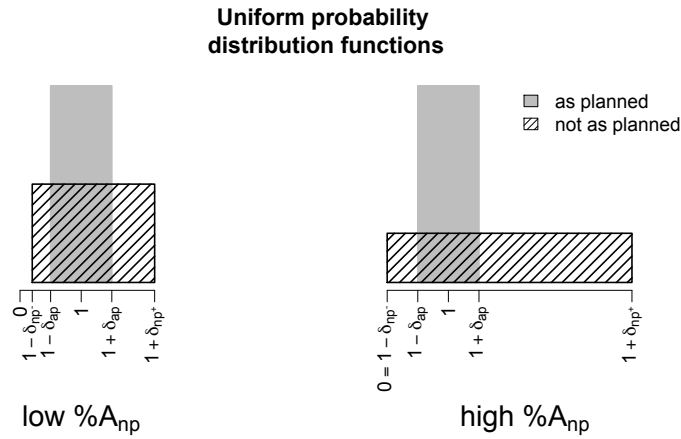
Figure 2.6: Defining the uniform input model for activity durations

uniform probability distribution that has a standard deviation larger than that of the *as planned* uniform distribution. Figure 2.6 shows two illustrative examples of this *not as planned* uniform distribution, hatched in black, for a low and a high value of %A_{np}. The variable %A_{np} expresses the percentage of the total number of activities N that has a duration that is not confined within the *as planned* margin δ_{ap}. %A_{np} is therefore proposed to provide an intuitive basis for creating *not as planned* project progress situations. The larger %A_{np}, the more the project progress can be regarded as going *not as planned* and warning signals generated by a project control procedure becomes more desirable.

The *as planned* parameter δ_{ap} and the *not as planned* parameter %A_{np} will be set in our experiment according to the values specified in table 2.7. From the δ_{ap} value, the extremes of the (lightgrey) *as planned* uniform probability distribution are calculated directly ($1 - \delta_{ap}$ and $1 + \delta_{ap}$). Indirectly, the extremes of the (hatched in black) *not as planned* uniform distribution function ($1 - \delta_{np-}$ and $1 + \delta_{np+}$) can be found using equations 2.11 and 2.12. We will discuss how these extremes are calculated along the following lines.

Let us assume that the N activities in the project have identical probability distributions and that they are sampled independently. The reader should consider the fact that we add dependencies through linear association in section 2.4.2. Let p_{np} denote the probability of drawing a value outside of the *as planned* margin. Consider this a success/failure or coin toss experiment where a draw from outside the *as planned* margin represents a success. The probability of drawing a number of successes N_{np} larger than

$\lfloor \%A_{np}N \rfloor$ from a total of N is given by the binomial distribution.

$$\mathrm{P}[N_{np} > \lfloor \%A_{np}N \rfloor] = 1 - \mathrm{P}[N_{np} \leq \lfloor \%A_{np}N \rfloor]$$

$$= 1 - \sum_{i=0}^{\lfloor \%A_{np}N \rfloor} \frac{N!}{i!\,(N-i)!} p_{np}^i (1 - p_{np})^{N-i} \quad (2.11)$$

In order to calculate p_{np} from equation 2.11, we have to assign a value to its left-hand side. This value represents the chance that a minimum of $\lfloor \%A_{np}N \rfloor$ successes is drawn. We chose this value as 95% in our experiments, in accordance with a probability of 5% that a fictitious project with less then $\lfloor \%A_{np}N \rfloor$ successes is simulated. For a simulation of $1,000$ runs, an average of 50 of such fictitious projects can then be expected to be included in the output. From equation 2.11, we proceeded with a numerical approximation to calculate p_{np}.

$$p_{np} = \mathrm{P}[|D - 1| > \delta_{ap}]$$

$$= 1 - \frac{2\delta_{ap}}{\delta_{np^+} + \delta_{np^-}} \quad (2.12)$$

From p_{np} and δ_{ap}, the extremes of the *not as planned* uniform distribution (represented using δ_{np^-} and δ_{np^+}) can then be found from equation 2.12, where D denotes the random variable that is drawn.

In order to derive a value for both δ_{np^-} and δ_{np^+}, an assumption has to be made with respect to one or both of these unknowns. We chose to implement a linear relation ($\delta_{np^+} = 1.5\,\delta_{np^-}$) which expresses the tendency of project activities to be late rather than early. However, in the cases where this relation resulted in a negative value for the lower extreme of the distribution ($1 - \delta_{np^-} < 0$), δ_{np^-} was fixed at 1 and the only unknown δ_{np^+} was found from equation 2.12. The factor 1.5 shown for the linear relation is arbitrary and different choices were also implemented during our tests.

Project progress simulation modelling

The Monte Carlo simulation generates activity durations for all activities according to the input models described in section 2.4.2. In accordance to recent literature we implemented two structures based on the linear association concept [23]. Linear association allows us to model the dependencies between activities and test how project progress is affected by this.

- Independence: The first structure can be regarded as a special case, since complete independence is assumed between activities. Activity durations are sampled independently from each other, but precedence constraints within the project assure that activities influence

one another, as expressed in real start times, and the project in its whole. Although this assumption of independence has been shown to fall short in accurately modelling project executions in real life [42, 46, 47], it has been used in both analytical [48, 49] and simulation [28, 29, 37, 50, 51] studies to produce insights into stochastic activity networks.

- Linear Association: The second structure models dependencies between activity durations. If users are required to specify correlations between activities, they face a challenging task since there is no established methodology for estimating correlation coefficients. In order to reject the independence assumption in our simulations, we employ the linear association technique [23, 42], where a systemic error B provides objective information on correlations.

  We say that N positive random variables Z_j are linearly associated if $Z_j = BX_j$ where $\{X_j\}$ is a set of n independent positive random variables and B is a positive random variable, independent from $\{X_j\}$ that can be regressed from historical data. We follow [23] and model B as a lognormal random variable, since the lognormal distribution is found appropriate to reflect both additive and multiplicative factors, ubiquitous in real-life project activities. The lognormal random variable is characterized by its mean μ_B and its coefficient of variation cv_B.

Our model assumes that the earned value (EV) for a single activity follows a linear accrue, starting from its actual start up to its budget at completion (BAC) when it is finished [12]. Planned value (PV) follows this same linear accrue from the planned start up to the planned finish time of the activity. EV and PV are calculated in P2 Engine at the project level and are compared to produce SV, SPI, SV(t) and SPI(t). The calculations for the X and R control charts and the recorded performances in section 2.5 are implemented in the statistical programming language R [52].

2.4.3   Overview of experiments

Table 2.7 gives an overview of the simulation experiments conducted for this research. A total of 400 different simulation experiments were run for each project network according to the inputs for the parameters μ_B, cv_B, δ_{ap} and $\%A_{np}$ provided in table 2.7. These parameters define respectively to what extent dependencies between activities are incorporated or not and the *as planned* and *not as planned* variation . It should be noted that although the independence assumption is retained for some scenarios (denoted as $cv_B = 0$), a constant B is still used in these to alter the mean ($\mu_B \in \{0.8, 1, 1.5, 2\}$). The 400 experiments produce 1,000 runs for

Baseline	Simulation model	Settings	
		as planned	*not as planned*
Estimates Uniform $\forall i \in \mathcal{N}: \hat{d}_i \in [8;56]$	INDEPENDENCE	$nrs = 1,000$ $\mu_B \in \{0.8, 1, 1.5, 2\}$ $cv_B = 0$	
		$\delta_{ap} \in \{20\%, ..., 80\%\}$ $\%A_{np} = 0$	$\delta_{ap} \in \{20\%, ..., 80\%\}$ $\%A_{np} \in \{20\%, ..., 80\%\}$
Fixed Cost Uniform $[0; 500]$			
Variable Cost Uniform $[700; 1500]$	LINEAR ASSOCIATION	$nrs = 1,000$ $\mu_B \in \{0.8, 1, 1.5, 2\}$ $cv_B \in \{0.3, 0.8, 2\}$	
		$\delta_{ap} \in \{20\%, ..., 80\%\}$ $\%A_{np} = 0$	$\delta_{ap} \in \{20\%, ..., 80\%\}$ $\%A_{np} \in \{20\%, ..., 80\%\}$

Table 2.7: Overview of the computational experiments

all 900 projects (section 2.4.1), for which the baseline estimates and costs are assigned according to the first column of table 2.7. Periodic EVM/ES reports are generated at 19 distinct moments in the project ($p = \{5\%, 10\%, 15\%, 20\%, \ldots, 90\%, 95\%\}$) and interpolation is done as shown in section 2.3 with a time-increment chosen sufficiently small. The durations of the activities were expressed in days, whereas the time-increment was in the order of hours.

We will use these experiments to demonstrate the value of statistical project control for project management in section 2.5 using the control efficiency measures defined in section 2.4.4. We have tried to incorporate the best practices for project control modelling as provided by the current literature, to produce results that are as generalisable to project management practice as possible. However, we will expand on the limitations of our research, and its results, in the concluding section 2.6.

2.4.4 Project control efficiency measures

In order to measure and compare the efficiency of X and R statistical project control charts we propose three simulation-based performance assessment measures [53].

The detection performance The *detection performance* equals the frequentist approximation that a certain *not as planned* situation will result in a signal during the *dynamic phase*, given the *as planned* data from the *static phase*. A signal is generated as in section 2.2.3 if the project performance Y_p or $|Y_p - Y_{p-1}|$ is not confined to the tolerance limits of respectively the X or R chart at any review period p.

The *detection performance* is calculated as:

$$\frac{1}{nrs} \sum_{s=1}^{nrs} \mathbb{1}_s\{\exists p|(Y_p < LTL_{X_p}^{\alpha}) \vee (Y_p > UTL_{X_p}^{\alpha})\} \qquad (2.13)$$

for the X chart, where $\mathbb{1}_s\{A\}$ is the indicator function of an event A for a simulation run s. Otherwise stated, we represent the occurrence of a signal for the X charts at review period p as the event $\mathcal{P}_X$:

$$\frac{1}{nrs} \sum_{s=1}^{nrs} \mathbb{1}_s\{\exists p|\mathcal{P}_X\} \qquad (2.14)$$

For the R chart, the *detection performance* is calculated analogously as :

$$\frac{1}{nrs} \sum_{s=1}^{nrs} \mathbb{1}_s\{\exists p|\mathcal{P}_R\} \qquad (2.15)$$

where:

$$\mathbb{1}_s\{\exists p | \mathcal{P}_R\} = \mathbb{1}_s\{\exists p | (|Y_p - Y_{p-1}| < LTL^{\alpha}_{R_p}) \vee (|Y_p - Y_{p-1}| > UTL^{\alpha}_{R_p})\} \tag{2.16}$$

The probability of overreactions The *probability of overreactions* approximates the probability that a signal arises even though all activities are executed *as planned*. This is calculated by 3-fold cross-validation. The runs of the *static phase* (going *as planned*) are divided into 3 equal groups and the *probability of overreactions* is calculated 3 times. Once for each group where the two other groups are used to calculate the tolerance limits. We confirm that deviations between groups never exceeded 1%, and in the remainder we will present the average over all 3 groups. Consequently, the *probability of overreactions* represents situations where project managers get a false impression that the project is being executed *not as planned*, which acts as a trigger to detect the reasons for this behavior, while there is no *unacceptable* variation present at the activity level. These false signals should be avoided since they lead to time-consuming and unnecessary overreactions made by the project manager and his/her team in charge of controlling the project performance. Each time an *not as planned* indication is reported, the project control chart is said to be overreacting since it generates a signal, while at activity level only *acceptable* variation is present.

The *probability of overreactions* for one group can be calculated using $nrs/3$ runs from the *static phase* as (2.14) and (2.15) for respectively the X and the R chart.

Area under the curve The *probability of overreactions* and the *detection performance* can both be used as approximations of the effectiveness of the proposed project control procedures. However, in the results section 2.5, the need for a more holistic representation of the effectiveness will be expressed. We find this in the concept of the calculated *area under the curve* (AUC) for a receiver operating characteristic curve (ROC). The ROC representation is widely used in classification testing and machine learning [54] and can be plotted directly from the *probability of overreactions* and the *detection performance*.

Riemann integration of the ROC curve provides us with a single unitless measure for *the area under the curve* that captures the characteristics of both the *probability of overreactions* and the *detection performance* at the same time. The AUC should be more than 0.5, which is the area under the no-discrimination line. This is the line where the true positive rate equals the false positive rate, and which can be regarded as the characteristic of a classification action that is purely based on a randomized process such as a coin toss.
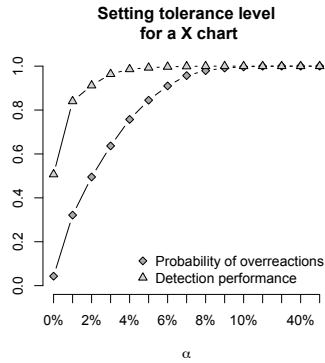
Figure 2.7: Setting the tolerance level α for a X chart, for a $(\delta_{ap},\%A_{ap})$=(20%,20%) scenario under the independence assumption $(\mu_B = 1, cv_B = 0)$.
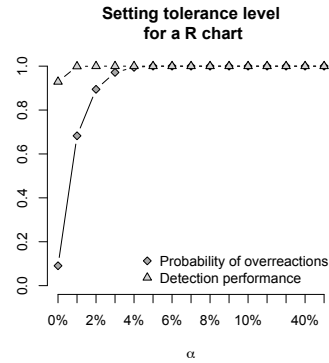
Figure 2.8: Setting the tolerance level α for a R chart, for a $(\delta_{ap},\%A_{ap})$=(20%,20%) scenario under the independence assumption $(\mu_B = 1, cv_B = 0)$.

2.5 Computational tests and results

In this section, we present our findings from our large computational experiment. First, we introduce how the concept of the tolerance level α can be used to set a desired degree of *probability of overreactions* and how the *detection performance* is affected, in section 2.5.1. Combined, these two output measures allow us to draw a ROC curve and to calculate the AUC.

Section 2.5.2 provides the validation for our project progress simulation model according to the recently postulated lognormal core theory [23, 42]. The efficiency of the described X and R statistical project control charts is explored in section 2.5.3 and is compared to a more pragmatic model of traditional EVM/ES use and how it can be implemented for different control points throughout the project in section 2.5.4

2.5.1 Close control versus overreactions - Setting a tolerance level

When the statistical project control charts are constructed, we use a parameter α as the tolerance level (see section 2.2.3). This represents the choice for the first and last α^{th} quantile of the empirical distribution function constructed in the *static phase* to form the tolerance limits of the X and R chart.
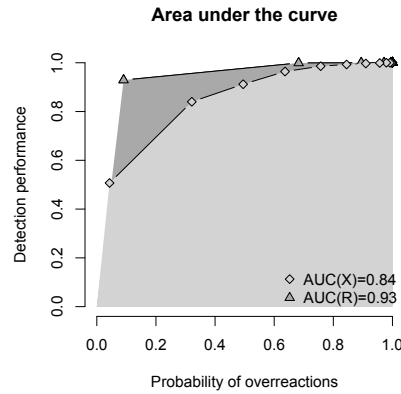
Figure 2.9: *Area under the curve* for the X and R chart, for a (δ_{ap},$\%A_{ap}$)=(20%,20%) scenario under the independence assumption ($\mu_B = 1, cv_B = 0$).

Is is clear that α will affect both the *probability of overreactions* and the *detection performance* of the X and R charts. Therefore, an appropriate choice for α should reflect both a project manager's aversion to risk and his/her willingness to invest effort in false alarms. Figures 2.7 and 2.8 illustrate the effect α has on the *probability of overreactions* and the *detection performance* for one specific scenario.

It is easily perceived that the real strength of a chart is in the relative improvement of the *detection performance* over a certain increase in the *probability of overreactions*. Optimally, we prefer the *probability of overreactions* as close to 0 as possible. This would mean that the project management team does not need to invest time and effort unnecessarily in drilling down the WBS, to find the variation at the activity level to be confined within the *acceptable* margins. Analogously, we prefer the *detection performance* as close to 1 as possible. For a certain scenario, this would imply that the *not as planned* situation gets detected somewhere throughout the lifetime of the project. This preference is accurately translated into a calculated AUC that needs to be as close to 1 as possible. The ROC curves and the corresponding AUC values for the X chart and R chart from figures 2.7 and 2.8 are depicted in figure 2.9.

2.5.2 Validation of the project simulation model

In order to validate the project progress simulation model (presented in section 2.4.2), we analysed the simulation output according to the lognormal core theory [23, 42]. Figures 2.10 and 2.11 depict boxplots for respec-
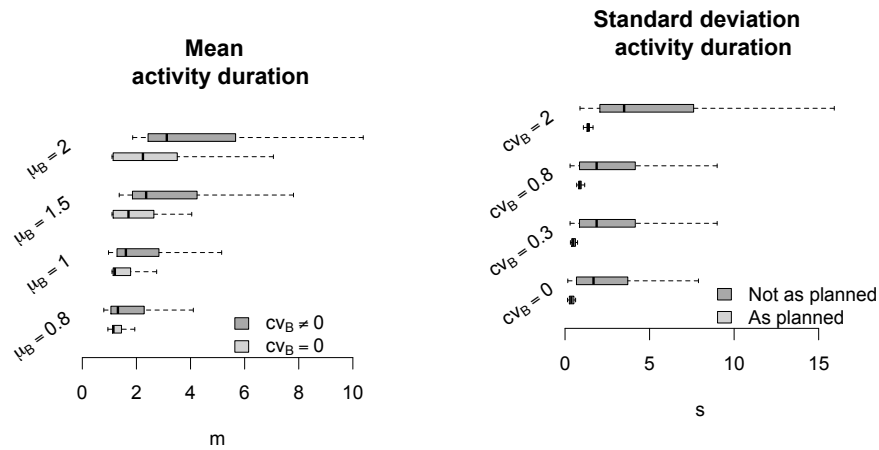
Figure 2.10: Distribution of the resulting mean in the project simulation model.

Figure 2.11: Distribution of the resulting standard deviations in the project simulation model.

tively the calculated mean and standard deviations of our simulation output.

Figure 2.10 presents the mean of the simulated activity durations. This figure clearly shows how the average activity duration in a project is affected by μ_B. If B is a random variable ($cv_B \neq 0$), rather than a constant ($cv_B = 0$), both the spread and the average of the mean duration is increased. Figure 2.11 shows the effect that the coefficient of variation cv_B has on the standard deviation of the simulated activity durations. Both the average and the spread of the standard deviation increase with increasing cv_B. This effect is noticeably larger for the *not as planned* situations.

In conclusion, we can safely state that in order to produce project progress simulations with variation in the orders of magnitude that is discussed by Trietsch et al. [42], with high variation instances having a standard deviation larger than 2, and large deviations in the mean, linear association via a lognormal variable B is preferred.

2.5.3   Project control efficiency

This part of the results section explores the efficiency of the proposed control charts using tolerance limits on EVM/ES by means of the *area under the curve* measure introduced in section 2.4.4.

Area under the curve
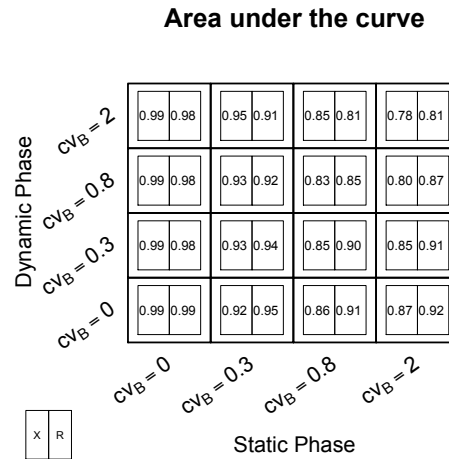
Figure 2.12: *Area under the curve* for different *static/dynamic phase* combinations

Area under the curve

The *area under the curve* representation in figure 2.12 provides a nuanced and holistic view of the project control efficiency for the X and the R chart. In general, the figure shows that the X and R chart perform better than that a randomized classification process would. All AUC values are significantly larger than 0.5, the AUC value of the no-discrimination line. Overall the X chart outperforms the R chart. This can be seen more clearly from an aggregated AUC calculated over all simulation experiments. This will be presented in section 2.5.4

For all cells along the diagonal of the *static/dynamic phase* grid, the choice for cv_B is equal in both phases and we can conclude that the R chart slightly outperforms the X chart. In other words, if an historical calibration precedes the simulations of the *static phase* and new projects can be expected to comply with the historical data, then the tolerance levels of the R chart prove to be the most efficient project control option. The cells below and above the diagonal represent the performance of the X and the R chart under parameter uncertainty [55]. When cv_B is expected to underestimate the real coefficient of variation of the linear association variable B during the *dynamic phase* (above the diagonal in the static/dynamic grid), then the X chart outperforms the R chart, although both produce very high AUC values. If however, the coefficient of variation of B is chosen too high (below the diagonal), the R chart will be more favourable for efficient project control during the *dynamic phase*.
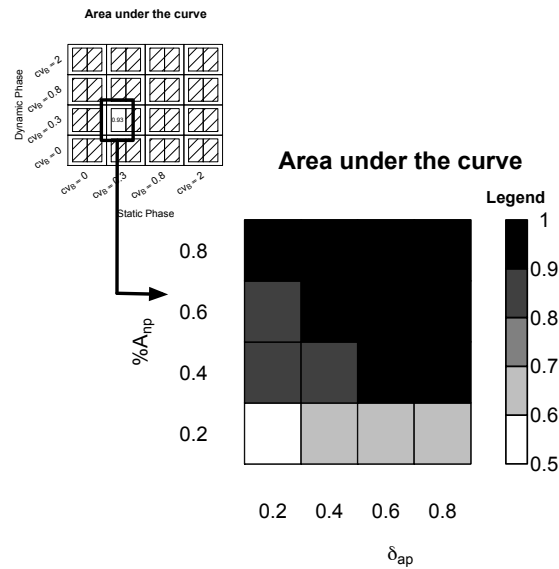
Figure 2.13: *Area under the curve* for different *not as planned* situations

Influencing factors for the X and R chart AUC

We proceed with a discussion in more detail for the AUC shown in figure 2.12. First, we provide an intuitive basis for the observed variation within each cell of the static/*dynamic phase* grid. Secondly, we investigate the effect that a change in the mean or coefficient of variation of the observed distribution of $d_i/\hat{d}_i$ has on the AUC.

Intuitive scenario representation
Without loss of generality, we will illustrate the intuitive basis on a single cell of the static/dynamic grid. We assume that the project management team has made the right choice for $cv_B = 0.3$ and $\mu_B = 1$ in the *static phase* from calibration to historical data recorded within the organisation. Moreover, we also assume that the real-life execution, for which the statistical project control charts are calculated, can be correctly modelled using the same random variable B. Figure 2.13 shows the expected AUC over a subset of *not as planned* scenarios.

In order to improve the readability of figure 2.13, we applied a color coding where the lightest shades of grey correspond to the lowest values and the darker to the highest values. The depicted scenarios correspond to *as*

planned situations for $\delta_{ap} \in \{20\%, 40\%, 60\%, 80\%\}$ and *not as planned* scenarios with $\%A_{np} \in \{20\%, 40\%, 60\%, 80\%\}$.

The color-coded variation in figure 2.13 was found to be identical for both the X and the R chart, so only that of the X chart is shown. Figure 2.13 largely confirms the intuition behind the scenario representation. The higher the percentage of activities $\%A_{np}$ that is likely to have a duration that is not confined to the *as planned* margin δ_{ap}, the more likely this scenario will produce a clear signal that can be interpreted using the statistical project control charts. For any $\%A_{np} \geq 20\%$, an increase in AUC can also be observed for increasing values of the *as planned* margin δ_{ap}. From equation 2.12, it can be seen that if p_{np} remains fixed (resulting from a fixed $\%A_{np}$), then $(\delta_{np^-} + \delta_{np^+})$ needs to increase if δ_{ap} increases. The extremes of the *not as planned* uniform thereby diverge, which results in a larger standard deviation and mean of the distribution. In the next section, it will be shown that increasing values for the mean and standard deviation lead to increasing AUC values.

Change in mean or standard deviation
This paragraph explores the effect the observed underlying distribution of $d_i/\hat{d}_i$ has on the *area under the curve*. We investigate the impact of a change in mean and standard deviation of the observed distribution of $d_i/\hat{d}_i$ for the *dynamic phase* relative to the *static phase* (which is used to calculate the statistical tolerance limits) for the X and R control charts. The relative change in mean $\Delta m = m_{\text{dynamic}} - m_{\text{static}}$ is the difference between the calculated mean of the activity durations in the *dynamic phase* (m_{dynamic}) and the mean of the activity durations in the *static phase* (m_{static}). Similarly, $\Delta s = s_{\text{dynamic}} - s_{\text{static}}$ is defined as the difference between the standard deviation of these two phases.

Figure 2.14 presents the effect a relative change in the mean of the underlying distribution has on the efficiency of the project control process. The x-axis represents the different ranges of values that are found for Δm, while the change in the standard deviation is kept close to zero between the *dynamic phase* and the *static phase* ($\Delta s \leq \pm 0.1$). In figure 2.14, we observe an increasing trend for the AUC for increasing values of Δm up to 1.2. If the mean of the *static phase* is e.g. 1, then increasing values for the AUC can be expected for a mean in the *dynamic phase* up to 2.2. Both the X and the R chart exhibit this behaviour, while the R chart significantly outperforms the X chart.

The effect of a change in the standard deviation Δs of the underlying distribution is shown in figure 2.15 for approximately equal mean values ($\Delta m \leq \pm 0.1$). The x-axis represents the ranges of values found for Δs. If the standard deviation of the underlying distribution of $d_i/\hat{d}_i$ decreases
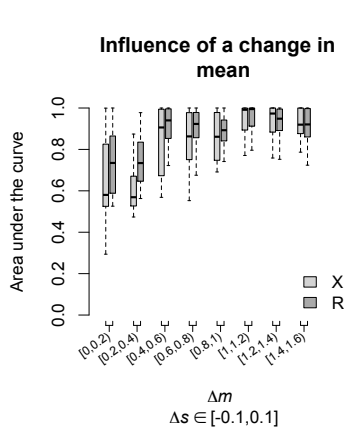
Figure 2.14: The effect a change in
the mean has on the AUC.
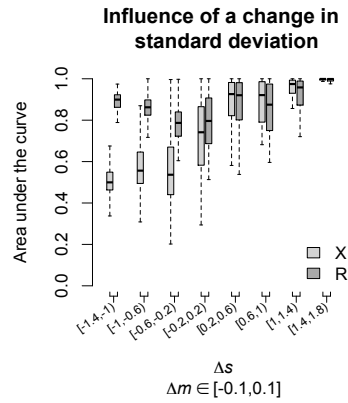
Figure 2.15: The effect a change in
the standard deviation has on the
AUC.

between the static and the *dynamic phase* ($\Delta s < 0$), we observe decreasing values for the AUC of the X control chart and increasing values for the AUC of the R control chart. Moreover, the R chart is preferred in order to accurately detect *not as planned* situations if $\Delta s < 0$. For positive Δs values, the AUC of both the X chart and the R chart increases. In these situations, the X chart performs better than the R chart.

2.5.4   Comparing statistical project control to traditional EVM/ES

In this section we revisit the discussion on the comparison of the use of project control charts using statistical tolerance limits (**STL**) to the more traditional use of EVM/ES metrics and some recent advances from literature for project schedule control. Although practical use of EVM/ES is typically characterized by on the spot decision-making from practical experience, we implement four techniques to compare our statistical control charts with.

- **Rules-of-thumb (RoT):** This project schedule control approach combines the use of project control charts with rules-of-thumb. Instead of our statical tolerance limits for the X chart, these charts employ either static, widening or narrowing tolerance limits for schedule control purposes. The tolerance limits, represented by $a + b\,p$, are implemented as symmetrical lines around a percentage complete axis. Therefore, the median of each performance metric $\hat{Q}(0.5)_p$ calculated in the *static phase*, is subtracted for all percentage project complete p.

- *Static tolerance limits* mimic the "Target Performance Chart" [56]. The slope a equals 0, for these tolerance limits

- *Widening tolerance limits* should be used in combination with SV and SV(t). These variance-based performance metrics grow larger in absolute figures during project progress and therefore, an increasing upper tolerance limit should be used. Since the lower tolerance limit is symmetrical, overall the tolerance limits can be said to be widening towards the end of the project.

- *Narrowing tolerance limits* should be used in combination with SPI and SPI(t). Simple simulations indicate that the variation for these index-based performance metrics decreases along the percentage complete axis. Consequently, narrowing tolerance limits are preferred during project progress.

Without loss of generality, we assume that these control strategies represent a best-case decision making process from practical experience, since they are calibrated to the *static phase* simulation data, much as the X and R control charts. This calibration is done for the slope a and the intercept b, which are optimized using a coarse search algorithm over a large solution space, in order to result in *probabilities of overreactions* that are competitive with those chosen for the X and R chart. In practical decision-making from experience, this calibration is impossible and the real-life *probabilities of overreactions* can be expected to be much higher. We therefore consider this RoT control strategy to be best-case EVM/ES decision making without the use of statistical tolerance limits.

- **EVM/ES on multiple control points:** The use of EVM/ES control metrics applied at different (and multiple) control points in the project schedule has recently been discussed by Colin and Vanhoucke [57]. Inspired by the concepts of the critical chain/buffer management (CC/BM) methodology, different control points throughout the project are suggested at which EVM/ES control charts should be monitored. The two alternative approaches are discussed in the original paper and can be briefly summarized as follows

  - *Feeding paths (FP):* The concept of feeding paths is adopted from the CC/BM literature. EVM/ES schedule control metrics are calculated for those paths in the project network where a buffer should be placed according to CC/BM in order to mitigate risks and to protect the critical path. The schedule control metrics are then monitored on a control chart against rules-of-thumb tolerance levels.

  - *Sub-networks (SN):* The concept of sub-network control points expands on the concept of feeding paths. Schedule control met-

Project schedule control using EVM/ES
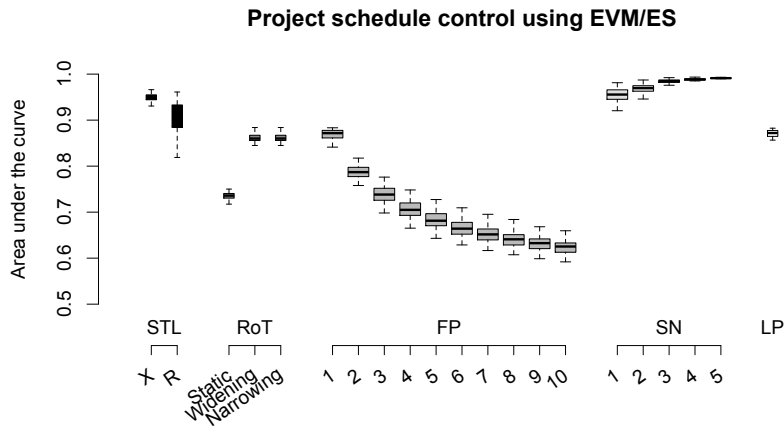
Figure 2.16: *Area under the curve* comparison of the X and R charts (**STL**) with other schedule control approaches using EVM/ES

rics are calculated for the collection of paths that enter the critical path at a given point. Again, these EVM/ES metrics are referenced against tolerance limits on control charts. The tolerance limits are calculated using sample quantiles, in a manner similar to that presented in section 2.2.3.

It should be noted that the differnece between the two techniques mainly lies in the number of control points, and figure 2.16 only gives a summary overview of the main results to compare the alternative project control methods.

- **Longest path monitoring (LP):** Inspired by the fact that the EVM/ES schedule control metrics are fully reliable for serial networks but have a decreasing performance for more parallel networks [28], the longest path control approach was proposed by Lipke [58]. In this method, SPI(t) is referenced against a static tolerance limit on a single project control chart. The SPI(t) is however not calculated for all activities of the project, but only for those that lie on the longest path in the project network. This longest path is updated dynamically during project progress with actual durations for those activities that have already been finished and the baseline estimate durations for the activities that are not yet started.

Figure 2.16 depicts the calculated *area under the curve* as boxplots for the statistical project control charts proposed in this paper and the alternative schedule control procedures outlined in this section. This comparison is based on all the data obtained from the simulations in the *static/dynamic*

phase grid of figure 2.12. A detailed discussion on the merits and the distinct characteristics of the procedures presented in this section falls outside of the scope of this paper. In short, we conclude that the X chart outperforms the R chart, based on the aggregated AUC over all performed simulation experiments. Both these charts, using statistical tolerance limits should be preferred over all other tested schedule control procedures for the following four reasons

- The RoT strategies display a significantly lower AUC. The narrowing tolerance limits and the widening tolerance limits can however be considered as useful alternatives.

- The FP method is shown for a combination of maximally 10 feeding paths monitored during project progress. The maximal number of feeding paths can however be much larger, even for small projects. The x-axis in figure 2.16 under FP depicts the number of control points that are used. The first control point is always the critical path, while the others are used to monitor the performance of the feeding paths. When not all feeding paths are monitored, information about certain activities is lost. This results in an increased *probability of overreactions*, which affects the *detection performance* and consequently the calculated AUC. Figure 2.16 shows that this produces a decreasing schedule control performance for an increasing number of included feeding paths.

- The SN method shows the opposite behaviour. AUC increases to the optimal unity area under the curve for 5 sub-network control points included in the schedule control procedure. When its performance is compared to the X chart, it shows to be a valuable alternative. However, while the X control chart relies on only one control point, the SN approach needs up to 5 control points to only slightly outperform the X chart. Moreover, the SN approach also relies on the STL procedure which is the topic of this paper.

- The LP method [58] is found to be outperformed by both the X and the R chart. Using our definition and measures for schedule control it is not able to produce similar results. However, this method was developed foremost to produce project duration forecasts, rather than to detect *not as planned* project progress.

2.6 Discussion and conclusions

In this paper, a project control system is presented which is inspired by the well-known statistical process control charts widely used to monitor manufacturing processes but is fundamentally different in its use of the

progress data to construct tolerance limits. The use of earned value management performance metrics in a control chart, based on simulated random variation, allows to discriminate between *as planned* and *not as planned* project performance variation. This paper measures the performance of the two proposed charts, using a wide range of fundamentally different project networks in a large computational experiment. The first chart (X chart) mimics a Shewhart chart for process control with adaptations to fit the finite nature of a project. The X chart uses the commonly known schedule performance metrics provided in an earned value management system. The second chart (R chart) proposes an adjustment on these metrics to focus on the instantaneous changes in project performance. The assumption is that by performing differential calculations, the accumulated performance of past activities is filtered and only active tasks are being monitored.

Overall the X chart shows to be the most promising to be used in practice to monitor deviations from the baseline schedule. Altogether, both show very comparable performance. The results of various computational experiments show the relevance and usefulness of this approach in a project control setting, although all the results are subject to the specific definition of *as planned* and *not as planned* project progress and the characteristics of the simulation model as applied in this research. In order to assess the performance of the statistical tolerance limit approach discussed in the paper, the X and R chart's performances are compared to add-hoc rules-of-thumb strategies that would typically be used in project control using EVM/ES and some recently developed EVM/ES extensions. Overall, the X and R chart can defensibly be preferred over all tested procedures

Obviously, the results obtained in this study should be put in the right perspective. The use of simulated data often shows some weaknesses and the results should therefore be interpreted with care. A first assumption in our simulation study lies in the dependence structure of activity durations during sampling. While the assumption of independence was invalidated in previous research [42, 59], it has nevertheless been used in earlier reports on project control [28, 29]. We also expanded our project progress simulation model with the concept of linear association [23]. However, the input for our simulation experiments depends largely on subjective estimates for activity durations, where in practice calibration to historical data should always be preferred.

A second restriction of our simulated project control study is the strict focus on time performance of projects. Although EVM/ES takes both a time and cost focus during project performance management, this paper puts a strong and restricted focus on the time dimension of a project in progress in order to validate the use of schedule performance metrics in a statistical project control approach. Activity costs have been assumed to be linearly

dependent on the activity duration time, which is restricted to the use of periodic costs. In the future, our model could be adjusted to incorporate other situations. Furthermore, other relevant project performance dimensions, such as quality control or project scope control could not be taken into account in this study. Future research will build on this paper, stretching its scope to incorporate the restrictions and expanding on the idea of using process control procedures for project control.

Finally, it should be mentioned that a detailed comparison between the statistical project control approach of this paper and the alternative control approaches available in literature is not within the scope of this paper. A comparison between the quality of methods such as Critical Path Method (CPM, [1]), the Programme Evaluation and Review Technique (PERT, [60]), its novel extension to PERT21 [23], the bottom-up approach using Schedule Risk Analysis [29] and the Critical Chain Project Management (CCPM, [61, 62]) is therefore considered as an interesting future research avenue to further improve and optimize the knowledge on project control.

This study should be relevant for both practitioners and academics for the following reasons. Performance indicators from earned value management/earned schedule have been used widely by practitioners in controlling projects. However, the lack of guidelines towards reasonable tolerance limits to discriminate between *acceptable* and *unacceptable* performance variation has often classified this technique as secondary to the necessary skills of intuition, existing experience and knowledge a project manager must have. The methodology and control approach proposed in this paper quantifies the use of tolerance limits. This paper shows that the discriminative power between *as planned* and *not as planned* variation in a project is much better when a customized control chart is used rather than when manual and intuitive thresholds (represented through the rules-of-thumb) are used. The on the spot decision-making during project control is thereby assisted. Next to the practical relevance, we also believe that this topic will contribute to new research challenges where extensions of this approach might lead to an increased discriminative power and better project control.

References

[1] J. Kelley, M. Walker, Critical path planning and scheduling: An introduction, Mauchly Associates, Ambler, PA, 1959.

[2] J. Kelley, Critical path planning and scheduling: Mathematical basis, Operations Research 9 (1961) 296–320.

[3] D. Malcolm, J. Roseboom, C. Clark, W. Fazar, Application of a tech-

nique for a research and development program evaluation, Operations Research (1959) 646–669.

[4] S. Hartmann, D. Briskorn, A survey of variants and extensions of the resource-constrained project scheduling problem, European Journal of Operational Research 207 (2010) 1–15.

[5] M. Madadi, H. Iranmanesh, A management oriented approach to reduce a project duration and its risk (variability), European Journal of Operational Research 219 (3) (2012) 751–761.

[6] E. Uyttewaal, Dynamic Scheduling With Microsoft Office Project 2003: The book by and for professionals, Co-published with International Institute for Learning, Inc., 2005.

[7] M. Vanhoucke, Project Management with Dynamic Scheduling: Baseline Scheduling, Risk Analysis and Project Control, Vol. XVIII, Springer, 2012.

[8] S. Rozenes, G. Vitner, S. Spraggett, Project control: literature review, Project Management Journal 37 (2006) 5–14.

[9] S. Rozenes, G. Vitner, S. Spraggett, MPCS: Multidimensional project control system, International Journal of Project Management 22 (2004) 109–118.

[10] Q. Fleming, J. Koppelman, Earned value project management. 3rd Edition, Newtown Square, PA: Project Management Institute, 2005.

[11] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[12] M. Vanhoucke, Measuring Time - Improving Project Performance using Earned Value Management, Vol. 136 of International Series in Operations Research and Management Science, Springer, 2010.

[13] D. Hulett, Schedule risk analysis simplified, Project Management Network 10 (1996) 23–30.

[14] Y. Fang, J. Zhang, Performance of control charts for autoregressive conditional heteroscedastic processes, Journal of Applied Statistics 26 (6) (1999) 701–714.

[15] W. A. Shewhart, Economic control of quality of manufactured product, Vol. 509, ASQ Quality Press, 1931.

[16] W. Lipke, A study of the normality of earned value management indicators, The Measurable News 4 (2002) 1,6,7,12–14,16.

[17] W. Lipke, J. Vaughn, Statistical process control meets earned value, CrossTalk: The Journal of Defense Software Engineering June (2000) 16–20,28–29.

[18] G. T. Bauch, C. A. Chung, A statistical project control tool for engineering managers, Project Management Journal 32 (2001) 37–44.

[19] Q. Wang, N. Jiang, L. Gou, M. Che, R. Zhang, Practical experiences of cost/schedule measure through earned value management and statistical process control, Lecture Notes in Computer Science 3966 (2006) 348–354.

[20] S. S. Leu, Y. C. Lin, Project performance evaluation based on statistical process control techniques, Journal of Construction Engineering and Management 134 (2008) 813–819.

[21] R. Aliverdi, L. Moslemi Naeni, A. Salehipour, Monitoring project duration and cost in a construction project by applying statistical quality control charts, International Journal of Project Management.

[22] E. Page, Continuous inspection schemes, Biometrika (1954) 100–115.

[23] D. Trietsch, K. R. Baker, Pert 21: Fitting PERT/CPM for use in the 21st century, International Journal of Project Management 30 (4) (2012) 490–502.

[24] M. Vanhoucke, S. Vandevoorde, A simulation and evaluation of earned value metrics to forecast the project duration, Journal of the Operational Research Society 58 (2007) 1361–1374.

[25] G. Burrows, Statistical tolerance limits–what are they, Tech. rep., Knolls Atomic Power Lab., Schenectady, NY (1962).

[26] R. Van Slyke, Monte Carlo methods and the PERT problem, Operations Research 11 (1963) 839–860.

[27] T. Williams, A classified bibliography of recent research relating to project risk management, European Journal of Operational Research 85 (1995) 18–38.

[28] M. Vanhoucke, Using activity sensitivity and network topology information to monitor project time performance, Omega The International Journal of Management Science 38 (2010) 359–370.

[29] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[30] M. Vanhoucke, Integrated Project Management and Control: First come the theory, then the practice, Management for Professionals, Springer, 2014.

[31] D. C. Montgomery, W. Woodall, Research issues and and ideas in statistical process control, Journal of Quality Technology 31 (4) (1999) 376–387.

[32] R. J. Hyndman, Y. Fan, Sample quantiles in statistical packages, The American Statistician 50 (4) (1996) 361–365.

[33] W. Lipke, Statistical methods applied to EVM ...the next frontier, The Measurable News Winter.

[34] L. Tavares, J. Ferreira, J. Coelho, The risk of delay of a project in terms of the morphology of its network, European Journal of Operational Research 119 (1999) 510–537.

[35] E. Demeulemeester, M. Vanhoucke, W. Herroelen, Rangen: A random network generator for activity-on-the-node networks, Journal of Scheduling 6 (2003) 17–38.

[36] M. Vanhoucke, J. Coelho, D. Debels, B. Maenhout, L. Tavares, An evaluation of the adequacy of project network generators with systematically sampled networks, European Journal of Operational Research 187 (2008) 511–524.

[37] S. Elmaghraby, On criticality and sensitivity in activity networks, European Journal of Operational Research 127 (2000) 220–238.

[38] S. Wang, G. Huang, An integrated approach for water resources decision making under interactive and compound uncertainties, Omega The International Journal of Management Science 44 (2014) 32–40.

[39] T. Williams, What are PERT estimates?, Journal of the Operational Research Society 46 (1995) 1498–1504.

[40] M. E. Kreye, Y. M. Goh, L. B. Newnes, P. Goodwin, Approaches to displaying information to assist decisions under uncertainty, Omega The International Journal of Management Science 40 (6) (2012) 682–692.

[41] M. E. Kuhl, E. K. Lada, N. M. Steiger, M. A. Wagner, J. R. Wilson, Introduction to modeling and generating probabilistic input processes for simulation, in: S. Henderson, B. Biller, M. Hsieh, J. Shortle, J. Tew, R. Barton (Eds.), Proceedings of the 2007 Winter Simulation Conference, New Jersey: Institute of Electrical and Electronics Engineers, 2007, pp. 63–76.

[42] D. Trietsch, L. Mazmanyan, L. Govergyan, K. R. Baker, Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation, European Journal of Operational Research (2012) 386–396.

[43] S. Mohan, M. Gopalakrishnan, H. Balasubramanian, A. Chandrashekar, A lognormal approximation of activity duration in pert using two time estimates, Journal of the Operational Research Society 58 (6) (2007) 827–831.

[44] E. Hahn, Mixture densities for project management activity times: A robust approach to PERT, European Journal of Operational Research 188 (2008) 450–459.

[45] T. Kotiah, N. D. Wallace, Another look at the pert assumptions, Management Science 20 (1) (1973) 44–49.

[46] C. Ragsdale, The current state of network simulation in project management theory and practice, Omega The International Journal of Management Science 17 (1) (1989) 21–25.

[47] R. Schonberger, Why projects are "always" late: A rationale based on manual simulation of a PERT/CPM network, Interfaces 11 (1981) 65–70.

[48] D. L. Fisher, D. Saisi, W. M. Goldstein, Stochastic pert networks: Op diagrams, critical paths and the project completion time, Computers & operations research 12 (5) (1985) 471–482.

[49] B. Dodin, M. Sirvanci, Stochastic networks and the extreme value distribution, Computers & Operations Research 17 (4) (1990) 397–409.

[50] T. Williams, Practical use of distributions in network analysis, Journal of the Operational Research Society (1992) 265–270.

[51] F. Acebes, J. Pajares, J. M. Galán, A. López-Paredes, A new approach for project control under uncertainty. going back to the basics, International Journal of Project Management.

[52] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2013). URL http://www.R-project.org

[53] T. Ponsignon, L. Mönch, Simulation-based performance assessment of master planning approaches in semiconductor manufacturing, Omega The International Journal of Management Science 46 (2014) 21–35.

[54] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshirani, The elements of statistical learning, Vol. 2, Springer, 2009.

[55] S. Lim, A joint optimal pricing and order quantity model under parameter uncertainty and its practical implementation, Omega The International Journal of Management Science 41 (6) (2013) 998–1007.

[56] F. Anbari, Earned value project management method and extensions., Project Management Journal 34(4) (2003) 12–23.

[57] J. Colin, M. Vanhoucke, A comparison of the performance of various project control methods using earned value management systems, Expert Systems with Applications 42 (6) (2015) 3159–3175.

[58] W. Lipke, Speculations on project duration forecasting, The Measurable News 3 (2012) 3–7.

[59] T. Williams, The contribution of mathematical modelling to the practice of project management, IMA Journal of Management Mathematics 14 (1) (2003) 3–30.

[60] W. Fazar, Program evaluation and review technique, The American Statistician 13 (1959) 10.

[61] E. Goldratt, Critical Chain, North River Press, Great Barrington, MA., 1997.

[62] R. C. Ash, P. H. Pittman, Towards holistic project scheduling using critical chain methodology enhanced with pert buffering, International Journal of Project Organisation and Management 1 (2) (2008) 185–203.

3

A comparison of the performance of various project control methods using earned value management systems

Modelling of the current state of
EVM/ES project control
methods and its derivatives

Development of a framework to
test and compare EVM/ES
project control methods

Single- and multivariate
advances to top-down project
control using EVM/ES

Top-down

- Rules of thumb (RoT)
- Statistical process control
  approaches (SPC)
- Critical Chain / Buffer
  Management (CC/BM)
  derivatives

Bottom-up

- Longest path method (LP)

- Project dataset
- Risk and variation
  modelling
- Two -phased Monte Carlo
  simulation (MCS)
- Measuring project control
  performance

Single

- Statistical tolerance limits
  (STL)

Multi

- Multivariate statistical
  project control (MSPC)
- Multivariate regression
  methods for longest path
  calculations (MR-LP)

Empirical perspective

Figure 3.1: Schematic overview of the elements, of the larger body of research in
this book, that are discussed in chapter 3.

Abstract

Recent literature on project management has emphasized the effort which
is spent by the management team during the project control process. Based
on this effort, a functional distinction can be made between a top down
and a bottom up project control approach. A top down control approach
refers to the use of a project control system that generates project based
performance metrics to give a general overview of the project performance.
Actions are triggered based on these general performance metrics, which
need further investigation to detect problems at the activity level. A bot-
tom up project control system refers to a system in which detailed activ-
ity information needs to be available constantly during the project control
process, which requires more effort. In this research, we propose two new
project control approaches, which combines elements of both top down
and bottom up control control. To this end, we integrate the earned value
management/earned schedule (EVM/ES) method with multiple control
points inspired by critical chain/buffer management (CC/BM). We show
how the EVM/ES control approach is complementary with the concept of
buffers and how they can improve the project control process when clev-
erly combined. These combined top down approaches overcome some of
the drawbacks of traditional EVM/ES mentioned in the literature, while
minimally increasing the effort spent by the project manager. A large com-
putational experiment is set up to test the approach against other control
procedures within a broad range of simulated dynamic project progress
situations.

3.1 Introduction

In this paper we focus our research on the control process during the execution of a project. Among the different approaches published in the project management literature, substantial distinctions exist with respect to the work breakdown structure (WBS) level at which the control process is performed, and consequently the effort which is spent during the process and the accuracy of potential actions triggered by such a process. We will restrict our attention to schedule control in this research.

Ultimately, the objective of the control process is finishing the project within a given deadline. It is assumed that the level of detail that has to be available for the project manager during project control corresponds to the effort spent during the control process. This research will not focus on the possible actions to be taken to bring the project back on track, we therefore refer to [16], [4] and [43] for an illustration of possible actions that can be incorporated into a dynamic project control experiment. Rather, we will discuss the project control process itself and analyse its performance based on whether or not it produces correct warning signals. We consider the project baseline schedule to be a given and will not discuss different objective functions that can be taken into account during project planning [22] under the availability of limited resources. The reader is referred to a recent survey written by [15] on that topic.

Figure 3.2 shows a classification of project control procedures according to the effort invested by the project manager during the project control process. The purpose of this figure is not to give an exhaustive list of the control procedures published in literature or to provide a bullet-proof classification for these control methods. Rather, we wish to express the reduced effort spent by a project manager when only high WBS level information needs to be recorded and processed at each review period during the top down project control process. This *top down* and *bottom up* classification was previously used in the research of [43]. A *bottom up* procedure requires an intensive and detailed control on all the activities at the lowest WBS level. With this information, a reliable estimate for the final project duration can be calculated and actions can be taken accordingly to meet a given project deadline. Alternatively, a *top down* procedure will only consider a single aggregate performance metric calculated at the top level of the WBS during project control. Only if necessary, additional effort can be spent by drilling down the WBS, in search of those activities that need actions, to ensure a timely completion of the project.

At the bottom of figure 3.2, the critical path method (CPM) is classified as requiring much project control effort by the project manager. CPM is one of the earliest reported approaches for project planning which can also be used in practice to form a basis for prediction of the total project duration

Top down	
EVM	[18], [12]
⋮	
EVM-1PB	[25]
EVM-STL	[9]
EVM-SNB	Section 3.2.2
CC/BM	[13], [16]
EVM-FPB	Section 3.2.2
EVM-LPB	[23]
⋮	
CPM	[19]
Bottom up	
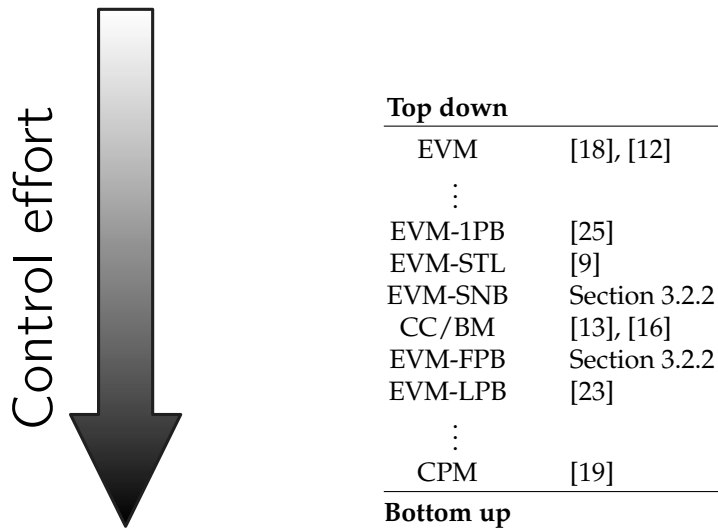
Figure 3.2: Control process effort classification

and to check progress against a produced baseline during project execution [19]. At each review period during project control, progress details on all the activities need to be reported. If the CPM algorithm is updated with actual durations for those activities that have been finished, expected finish times for the activities that are in progress and the baseline estimates for the other activities, it produces a reliable estimate for the final project duration. Updating the information for all individual activities can become a cumbersome and disruptive task for project teams in projects with a large number of activities [25].

At the top of figure 3.2, earned value management is classified as demanding less project control effort by the project manager. EVM was originally developed in the 60's by the U.S. Department of Defence as a project cost and schedule control procedure that evaluates and reports performance metrics calculated at high levels of the WBS. [12] brought EVM under the attention of researchers, and recent publications have produced the earned schedule method (ES; [25]), new project cost and duration forecasting methods using artificial intelligence [5, 6, 47] and fuzzy logic [28], and dynamic EVM systems for monitoring [21] and visualisation [7] of the performance of a project. [18] argue that an aggregate look of the project performance at the highest level of the WBS might lead to misinterpretations of the real project performance and errors in the reported warning signals, and state that the EVM performance measures should be used at lower WBS levels, obviously leading to an increased effort for the project manager. On the contrary, [25] argue that the use of EVM on lower levels of the

WBS is a cumbersome and often disruptive tasks for the project manager and EVM/ES needs to be applied at high levels of the WBS. These conflicting views on the optimal level for project control using EVM/ES have inspired the work in this paper.

More precisely, we will propose control points for a project at a level of the WBS in-between the top level of traditional EVM and the bottom level of the CPM. These can be interpreted as intermediate levels of the WBS at which the EVM/ES performance measures are calculated. In doing so, they replace the use of control accounts (CA) in an EVM system. Control accounts are natural management points for planning and control, since they represent the work assigned to one responsible organizational element in the WBS. However, since the WBS ordering in control accounts is brought forth by organizational or practical considerations, these control accounts have little or no correspondence to the baseline schedule, i.e. the logic followed during the execution. The proposed control points in this research can also be interpreted as locations in the project activity network. We will therefore frequently refer to the placement of a control point in the network, as calculated from the baseline schedule. The reader should note, that the timing of control points is not discussed in this research [29, 32]. Each discussed control approach will have an equal number of observations, distributed uniformly over the duration of the project. While the research on the timing of control points is concerned with minimising the number of observation points during the execution of the project [14], our objective is to investigate the effect of grouping activities into subsets. These subsets are then controlled separately, which should minimise the probability that a deviation from the baseline schedule goes unnoticed and endangers the project deadline. We will propose control points for different control approaches in this research, while incorporating the concept of buffers from the Critical Chain / Buffer Management (CC/BM) methodology into the EVM/ES control process to include project baseline information in a structured manner:

- Two new EVM/ES approaches will be introduced in this paper. Similar to the CC/BM approach, buffers are added as control points to the project in each feeding path that enters the critical path (EVM-FPB). EVM/ES performance measures will then be used to monitor the progress of both the critical path and all the feeding paths. While this can lead to a high number of control points, a second approach will also be presented to reduce the number of control points by adding buffers on subnetworks instead of on all feeding paths (EVM-SNB).

- We will test these newly proposed control approach against three additional EVM/ES procedures found in the literature. The traditional EVM/ES control methodology makes use of a single control point at the top level of the WBS, and is therefore labelled as EVM-1PB (1

project buffer). This control methodology can be extended by statistical tolerance limits (EVM-STL) as proposed by [9]. [42] has shown that the EVM methodology shows a reliable behaviour when project networks are close to a serial network, and therefore, [23] has proposed to monitor the project performance on the dynamic longest path of the project (EVM-LPB) to improve its performance. This approach requires the calculation of the critical path at each review period and lies closer to the traditional CPM method in figure 3.2.

It should be noted that the control approaches discussed and tested in this paper are inspired by the CC/BM control mechanisms without having the intention to completely follow the CC/BM methodology initially proposed by [13]. The Critical Chain/Buffer Management project planning and control approach can also be characterized as an in-between top down and bottom up control approach in terms of the control effort on figure 3.2. CC/BM is Goldratt's theory of constraints translated to the field of project management [31]. Shortly after Goldratt's initial formulations, CC/BM was found to be popular in both project management practice and literature [34, 39]. However, more than a purely algorithmic matter, CC/BM is believed to be an important strategic decision and its underlying assumptions and principles have all been challenged in literature. Primarily, the focus of CC/BM is to protect the makespan of the project. From a project control stance, CC/BM accepts that the project in progress is subject to considerable uncertainty, but tries to mitigate risks through focusing on the critical chain and by building in buffers.

Buffers are added to the baseline schedule in three forms. A *project buffer* is added at the end of the project as a way to merge the build-in contingencies that are usually found in individual activities, ultimately reducing the total project duration. The main argument for creating this project buffer is not that there is in general too much contingency built into schedules but rather that it is built in at the wrong place [33]. In order to reduce the contingencies at the activity level, it is conjectured that the activity durations need to be cut aggressively, as much as halving (50% rule) the original estimates [13, 16]. The second type of buffer added in CC/BM, the *feeding buffer*, has also been the subject of debate. [39], [35] and [38] tested different methods to accurately estimate the optimal buffer size to be added to all non-critical paths that feed into the critical chain. The activities on these non-critical paths are scheduled as late as possible to minimize the work in progress (WIP, the amount of partially finished work in the project) and to decrease the chance of rework if design problems are discovered. Moreover, gating tasks (i.e. tasks that have only a dummy start node as a predecessor) should not start before the scheduled start time, while non-gating tasks should be started as soon as they can, when work becomes available. This last principle is denoted the *roadrunner mentality*. [16] question this practice as it is not sure whether this is always beneficial with respect to

WIP reduction. Furthermore, the implementation of this roadrunner mentality forces the user to maintain two different schedules, i.e. a baseline schedule and a projected schedule. The reader is referred to [16] for a detailed discussion on the merits and pitfalls of the CC/BM methodology. A third type of buffer, the *resource buffer*, is placed whenever a resource is assigned to an activity on the critical chain and the previous task on the critical chain is performed by another resource. Resource buffers usually take the form of an advance warning signal or idle time that can be created around the resource to create some kind of protective capacity. Since resources are not taken into account in this paper, we will not discuss resource buffers any further.

In this research, we did not have the intention to completely follow the CC/BM methodology. Rather than building buffers into the baseline schedule to mitigate risks, we partly adopt the nomenclature and schedule characteristics of the buffers, from the CC/BM methodology, to define control points throughout the project. At these control points, the schedule performance of the project is checked dynamically during project progress to act as an early warning signal to trigger potential actions. The first control point will always be situated at the project buffer, and will monitor the schedule performance of different subsets of the project activities depending on which project control approach is being used (i.e. EVM-1PB, EVM-STL, EVM-FPB, EVM-SNB or EVM-LPB). Additional control points can be added to monitor the consumption of slack for feeding path buffers (for EVM-FPB) or for subnetwork buffers (EVM-SNB). The use of these buffers will be formalized in section 3.2. The number of control points utilised in each project control approach can be interpreted as the effort that needs to be spend by the project manager during the project control process. The EVM-1PB and EVM-STL method employ only one control point, while supplementary control points can be formulated for the EVM-FPB and EVM-SNB control approaches. However, it will be shown further in this paper that the use of EVM/ES performance measures on feeding or subnetwork buffers can lead to an improved ability to protect the project deadline without increasing the number of control points too dramatically. Discussing the number of control points for the EVM-LPB control approach directly, might give the reader the wrong impression that the control effort spent in EVM-LPB is similar to EVM-1PB and EVM-STL. EVM-LPB however, requires calculations of the CPM at each review period and therefore assumes activity level progress data, even though the schedule performance is interpreted at the project buffer. In terms of control effort the EVM-LPB method exceeds the other discussed control approaches. We will test all five control approaches over a broad range of project progress simulations in the computational experiments outlined in section 3.3. The results for these experiments will be discussed in section 3.4 and general conclusions and a discussion will be given in section 3.5.
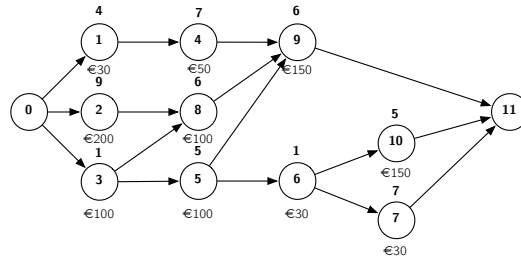
Figure 3.3: An illustrative project network

3.2 Methodology

In this section, the EVM-FPB and EVM-SNB procedures will be introduced using a small fictitious project example. In section 3.2.1, we introduce the notations used in the proposed methodology. We will then demonstrate our methodology in section 3.2.2, by illustrating how EVM/ES is integrated within the concept of buffers that act as control points in the schedule. We will formalize the EVM-FPB approach in section 3.2.3 and the EVM-SNB approach in section 3.2.4.

3.2.1 Notation and project scheduling

Figure 3.3 is used to present the EVM-FPB and EVM-SNB control procedures in this section. The activity network of a small project with 10 non-dummy activities is represented in an activity-on-the-node diagram. The baseline estimates for the activity durations are expressed in days as shown above the nodes, while the daily costs for the activities are denoted below the nodes in the network. Further notations for the network are formalized in table 3.1.

The total planned cost for an activity can be found by multiplying the periodic cost below the node with its baseline estimate duration above the node. For activity i, the total planned cost is given by its budget at completion, $\mathrm{BAC}^{\{i\}} = c_i * \widehat{d_i} = (30\,€) * 4 = 120\,€$. Without loss of generality, the planned value for an activity i is assumed to accrue linearly from 0 at the starting time to $\mathrm{BAC}^{\{i\}}$ at the planned finish time. The earned value for an activity i is conjectured the follow this same linear accrue between its real starting time and its real finish time. This distribution of value over the activity duration is believed to model most realistic situations where costs are expressed in man-hours and was used previously in studies by [41] and [8, 9]. We refer to the papers by [12] and [43] for further details on the calculations of EVM, and to [24] for the Earned Schedule (ES) calcula-

Project characteristics

$G = (\mathcal{N}, \mathcal{A})$	Project activity network
$\mathcal{N}$	Set of activities (nodes) in the project
$\mathcal{A}$	Precedence relations (arcs) in the project
$\delta_{\mathcal{N}}$	Deadline for the project (in days)
$\mathcal{P}_i$	Set of predecessors of activity i: $\{j \in \mathcal{N} | (j, i) \in \mathcal{A}\}$
$\widehat{d_i}$	Baseline estimate for the duration of activity i (in days)
d_i	Real activity duration of activity i (in days)
c_i	Periodic cost for the activity i

Earned value management

Project EVM/ES key metrics

BAC^x	Budget at completion for the set of activities x
PV_t^x	Planned value for the set of activities x at period t
EV_t^x	Earned value for the set of activities x at period t
ES_t^x	Earned schedule for the set of activities x at period t

Project EVM/ES performance metrics

$\mathrm{SV}_t^x = \mathrm{EV}_t^x - \mathrm{PV}_t^x$	Schedule variance for the set of activities x at period t
$\mathrm{SPI}_t^x = \mathrm{EV}_t^x / \mathrm{PV}_t^x$	Schedule performance index for the set of activities x at period t
$\mathrm{SV(t)}_t^x = \mathrm{ES}_t^x - t$	Schedule variance using earned schedule for the set of activities x at period t
$\mathrm{SPI(t)}_t^x = \mathrm{ES}_t^x / t$	Schedule performance index using earned schedule for the set of activities x at period t

with:

$t = 1, \ldots, T$	Current time period
T	Total duration of the project

Table 3.1: Project network and EVM notations

tions. A summary of the EVM/ES notations used in this paper is given in table 3.1.

For the illustrative project network of figure 3.3, the critical path is found by a single forward/backward pass through the network. Figure 3.4 shows the critical path of the example network and the resulting earliest start schedule in a Gantt chart. The critical path consists of activities 2-8-9, resulting in a planned project makespan of 21 days. During the execution of the project, the real duration d_i of an activity i can be different from its baseline estimate $\widehat{d_i}$. The slack present in the baseline schedule can then be consumed and non-critical paths might become critical, influencing the total project duration. In section 3.2.2, we will illustrate and validate a project control model using EVM/ES that monitors both the performance of the activities on the critical path as well as the consumed slack by non-critical activities feeding into the critical path. This consumption of the slack can then act as a signal to trigger potential actions. To that end, we present 10 fictitious project executions of the example project in table 3.2 in which each activity has a real duration d_i that might differ from its baseline estimate $\widehat{d_i}$.

3.2.2   Illustration of the project control model

In this section, we will demonstrate the new project control approach using EVM/ES. We will improve the schedule control performance of EVM/ES by calculating the performance metrics at multiple control points in the baseline schedule, instead of calculating them for all activities at the highest WBS level. These control points are not chosen arbitrarily, but are a direct consequence of the critical path baseline schedule of the project. The first control point will monitor the critical path, and will be situated at the project buffer. This will be discussed in section 3.2.2. Non-critical activities on a path that feeds into the critical path can also become critical when their slack is consumed. In order to monitor the consumption of slack throughout the project, the EVM-FPB method considers every feeding path as a control point for which EVM/ES calculations are to be made. Section 3.2.2 discusses this approach. EVM-FPB is likely to lead to a large number of control points in the project since each feeding path is controlled individually, and correspondingly, the project control effort will be high. The EVM-SNB procedure tries to reduce the control effort through reduction of the number of control points in the project. To this end, feeding paths are combined to form a subnetwork. Section 3.2.2 shows how the slack of such a subnetwork can be controlled dynamically.
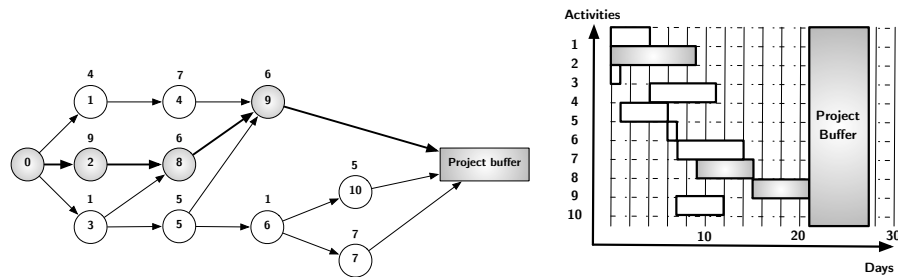
Figure 3.4: Critical chain for the illustrative project network

Project buffer

The dummy end node of figure 3.3 is transformed into a single project buffer in figure 3.4. This project buffer will act as a control point using EVM to predict whether the project deadline is likely to be met. The deadline is chosen in this example as a fixed percentage i.e. 30% of the project planned duration, as shown in the Gantt chart.

The project buffer can act as a control point for EVM/ES in different ways, dependent on which control approach is being used. Let Y_t^x denote the EVM/ES measure Y calculated at a time period t for a set of activities x. Y can then represent each of the project EVM/ES performance metrics shown in table 3.1. At the project buffer, the EVM/ES performance metric Y_t^x will be controlled dynamically to evaluate whether the project is likely to meet the deadline. The specific EVM control approach which is used, determines the set of activities x for which Y is calculated at each time period t. The EVM control approaches introduced in the following sections (EVM-FPB and EVM-SNB) will only control the critical activities in the project buffer. The other EVM approaches (EVM-1PB, EVM-STL and EVM-LPB) discussed in this paper have a different set x but all of them are controlled at this project buffer control point. We will provide more details on the set of activities x for those control approaches later in this paper.

Feeding path buffer

In order to present the EVM-FPB control procedure, we will restrict our attention to the non-critical path 3-5-6-7 that feeds into the critical path. This path has an expected duration of 14 days and 7 days of slack. This slack is used as the feeding path buffer F_1 as shown in figure 3.5. Table 3.3 introduces the notations used to discuss feeding path buffers in the EVM-FPB control approach and subnetwork buffers in the EVM-SNB control approach.

Table 3.2: Activity durations for fictitious project executions

Simulation run	Activities										Path 3-5-6-7
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**3-5-6-7**
e_i	4	9	1	7	5	1	7	6	6	5	14
1	6	17	3	10	6	2	8	11	10	10	19
2	6	10	7	13	8	3	7	7	4	9	25
3	3	9	1	13	9	3	10	9	9	6	23
4	5	14	2	12	4	1	5	10	7	8	12
5	8	9	1	5	10	1	12	8	12	3	24
6	3	15	1	5	4	1	14	9	10	8	20
7	6	9	1	10	4	1	8	9	10	9	14
8	7	11	2	7	9	1	5	4	6	3	17
9	6	15	1	7	5	1	8	11	6	8	15
10	7	9	1	8	5	1	10	6	5	5	17

Note: $\forall i \in \mathcal{N} : d_i \in \mathbb{N}$
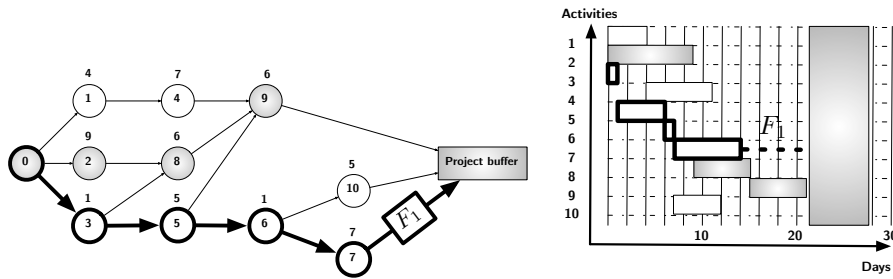
Figure 3.5: Path 3-5-6-7 as the feeding path buffer F_1

Consider a dynamic project control approach that calculates the expected path length ($\widehat{\mathrm{RD}}_{F_j}$) as the sum of the real activity durations for the activities that are finished, the expected duration of the activities in progress and the baseline estimate duration for all activities in the future. The expected buffer penetration can then be calculated for each time period as

$$100\frac{\widehat{\mathrm{RD}}_{F_j} - \mathrm{PD}_{F_j}}{s_j^*} \tag{3.1}$$

This percentage can be less than 0% when the path is performed ahead of schedule, between 0% and 100% when the existing slack is expected to cope with the delays, or more than 100% when the slack will be consumed entirely and the critical path is potentially endangered.

Figure 3.6 shows the buffer penetration of three example runs (1, 2 and 3) from table 3.2 along the duration of the feeding path. More precisely,

Notations for a buffer F_j

$\mathcal{F}_j$	Set of activities of the feeding path/subnetwork
$\mathcal{A}_{F_j}$	Set of precedence relations $(i,j) \in \mathcal{A}|i,j \in \mathcal{F}_j$
s_j^*	Slack of the last activity of the feeding path/subnetwork
PD_{F_j}	Planned duration of the feeding path/subnetwork
RD_{F_j}	Real duration of the feeding path/subnetwork
$\widehat{\mathrm{RD}}_{F_j}$	Expected real duration of the feeding path/subnetwork

Baseline schedule information

s_i	Slack of activity i in the project baseline schedule
FT_i	Planned finish time of activity i in the project baseline schedule
ST_i	Planned start time of activity i in the project baseline schedule

Table 3.3: EVM-SPB and EVM-SNB buffer notations and project baseline schedule information

the x-axis denotes the current path duration and the y-axis denotes the expected buffer penetration. As an example, activity 3 is finished after day 3 in run 1, and given the expected remaining durations of the other activities, the expected path duration equals 16. Consequently, the expected buffer penetration after day 3 is 100*(16-14)/7 = 28.57%. The final buffer penetration at day 19 for the project run 1 for F_1 is 100*(19-14)/7 = 71.42%. As long as the final buffer penetration is less then 100%, the slack is sufficient to protect the critical path against delays in the feeding path 3-5-6-7. For the second and the third fictitious project executions, this is clearly not the case as they both end beyond 100% buffer penetration. We have added a suggestion for a straightforward tolerance limit to trigger early warning signals that could indicate the need to take actions, during the execution of the feeding path. This is done by the lightgrey and darkgrey shaded areas on the graph in figure 3.6. The line which divides the two areas assumes that the 100% buffer is linearly spread over the path duration. From figure 3.6, we notice that this straightforward tolerance limit would trigger a signal for all of the three presented example runs. Since the final consumption of the slack for run 1 is lower than 100%, the warning signal could be interpreted as false. The signals for run 2 and 3 however, express a genuine need for corrective actions since they both end with more than 100% of the slack consumed.

This simple tolerance limit approach can be easily replicated using the standard EVM/ES performance metrics. EVM/ES provides a birds-eye view on the project performance by calculating schedule performance metrics at the top levels of the WBS. Moreover, it is known from literature [43] that these performance metrics perform best when measuring the schedule performance of a project with a network that lies close to a completely serial network. Consequently, it is an obvious step to consider our feeding path buffer, which acts as a control point for a 100% serial non-critical feeding path, as an EVM/ES performance control point, since it has the guarantee to provide reliable schedule estimates along the complete lifetime of the feeding path.

If Y_t^x denotes the EVM/ES measure Y calculated at a time period t for a set of activities x, then $ES_t^{\mathcal{F}_1}$ is the earned schedule calculated at a period t for the activities in the buffer F_j. If $ES_t^{\mathcal{F}_1}$ and $SPI(t)_t^{\mathcal{F}_1}$ are calculated for the feeding path, the first three runs of table 3.2 result in the lines depicted in figure 3.7. The x-axis denotes the feeding path percentage complete, which always equals $ES_t^{\mathcal{F}_1}/PD_{F_1}$, and the y-axis denotes the calculated $SPI(t)_t^{\mathcal{F}_1}$. Let us now evaluate $SPI(t)_t^{\mathcal{F}_1}$ for run 1 at the same days as we did when we discussed the expected buffer penetration. After 3 days ($t = 3$), only the work that was scheduled to be performed in one day ($ES_3^{\mathcal{F}_1}=1$) has been accomplished. Therefore, after 7.14% (1/14) of the feeding path planned duration, the $SPI(t)_t^{\mathcal{F}_1} = 1/3 = 0.33$. At the completion of the feeding
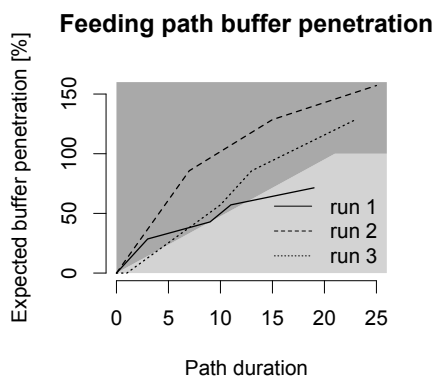
Figure 3.6: Expected buffer penetration of F_1
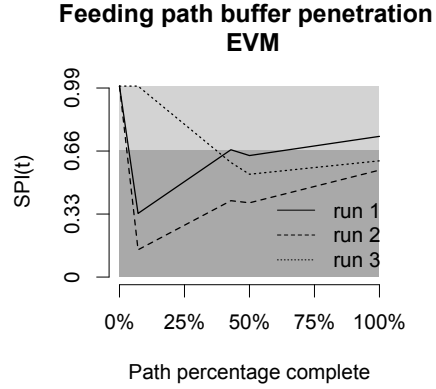
Figure 3.7: Dynamic SPI(t) control of path F_1

path, it has taken ($t = 19$) days to complete the work that was scheduled to be done in 14 days, and hence the final $\mathrm{SPI(t)}_1^{\mathcal{F}_1}$ equals 0.73. Instead of depicting the expected buffer penetration along the duration of the path and controlling it against a linear tolerance limit as was done in figure 3.6, the $\mathrm{SPI(t)}_t^{\mathcal{F}_1}$ can be measured directly against a static threshold. In this approach, the expected path length ($\widehat{\mathrm{RD}}_{F_1}$) is calculated using the formula of [24] as:

$$\widehat{\mathrm{RD}}_{F_1} = \frac{\mathrm{PD}_{F_1}}{\mathrm{SPI(t)}_t^{\mathcal{F}_1}} \tag{3.2}$$

This formula assumes that the current SPI(t) is representative for the schedule performance of the rest of the feeding path. Given a maximum buffer consumption of 7 days and a maximum feeding path duration of 21 days, the minimum allowable SPI(t) value is equal to $14/21 = 0.67$. The SPI(t) is not allowed to drift below this value to avoid more than 100% of the slack being consumed. This minimal value is depicted by the border between the lightgrey and darkgrey area in figure 3.7. This is the same as distributing the buffer over the length of the path linearly, such as was done in the graph of figure 3.6 and consequently, the same conclusions can de drawn for the fictitious project executions depicted in table 3.2.

Subnetwork feeding buffer

In the previous section, the slack of a single non-critical path feeding into the critical path was controlled using an EVM/ES measurement system.
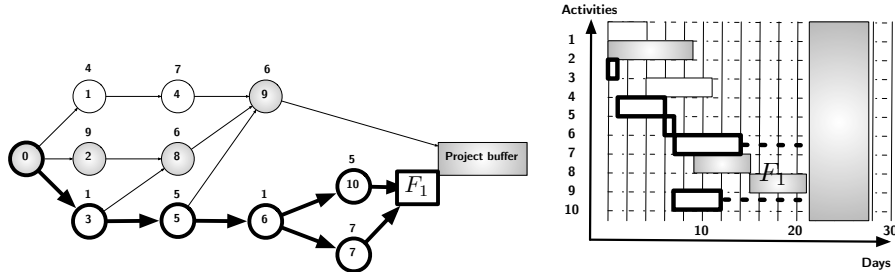
Figure 3.8: F_1 consisting of the sub-network 3-5-6-7-10

This provides the management with a reliable estimate whether or not this feeding path is likely to become critical at one point. If this approach is followed for all the non-critical paths that feed into the critical path, the total number of control points can become very large. In this section, we will present an alternative control approach (EVM-SNB) aiming to reduce this high number of control points. This approach combines all the non-critical paths feeding into the critical path in a subnetwork. EVM/ES performance metrics will be used to control all the activities of the subnetwork instead of the individual feeding paths. Calculating the slack for such a subnetwork is no longer trivial, since there might be more than one activity that feeds into the critical paths at the same place in the network. Consequently, tolerance limits are difficult or impossible to derive analytically for the slack present in the baseline schedule. When the subnetwork buffer includes more than one path, we will no longer use the analytical procedure to calculate tolerance limits followed in figure 3.7. However, we will show how an alternative approach using statistical tolerance limits will lead to a figure that is as easily interpretable.

Figure 3.8 shows the subnetwork buffer F_1 that now consists of all of the activities of a subnetwork that feeds into the critical path at the project buffer, i.e. $\mathcal{F}_1 = \{3, 5, 6, 7, 10\}$. The subnetwork buffer F_1 is composed out of the two feeding paths 3-5-6-7 and 3-5-6-10 that both feed into the critical path at the same node, with each a different slack value. Path 3-5-6-7 is constrained by the project makespan and has a slack of 7 days, as depicted in figure 3.8. Path 3-5-6-10 is planned to finish two days earlier and its slack is 9 days. Both paths can become critical when they consume more than their slack during the execution of the project. Instead of controlling each path individually using the approaches outlined in section 3.2.2, we wish to reduce the number of control points by combining them into a subnetwork buffer. To that end, statistical tolerance limits for a performance measure $Y_p^{\mathcal{F}_1}$, calculated for the subnetwork, should be derived. The tolerance limits for $Y^{\mathcal{F}_1}$ should now be calculated for each percentage project complete p, ranging between 10% and 90% instead of for each time instance
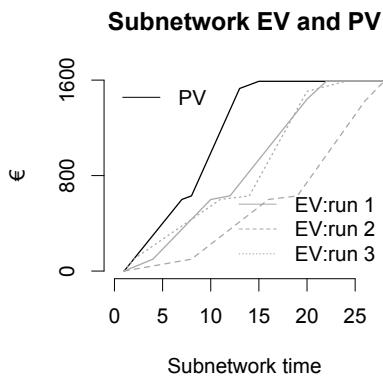
Figure 3.9: PV and EV curves for the subnetwork FB_1
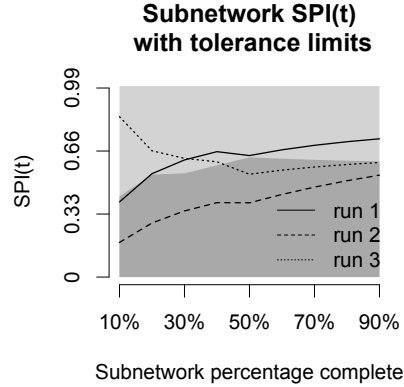
Figure 3.10: Dynamic SPI(t) tracking of FB_1, with tolerance limits

t [9].

In order to calculate these statistical tolerance limits, we need a reference set of project executions. The right side of table 3.4 shows the duration of the two paths in our subnetwork. In order for the subnetwork not to infer with the critical path, the two paths have to finish in less than 21 days after the start of the project. When we remove all project progress runs from table 3.4 that consume more than their predefined slack for any of the two paths, the remaining project progress runs (with * in table 3.4) can be used to calculate sample quantiles for $\mathrm{SPI(t)}^{\mathcal{F}_1}$ that act as thresholds that should not be exceeded, just as the minimal SPI(t) tolerance limit depicted in figure 3.7. The $\mathrm{SPI(t)}_p^{\mathcal{F}_1}$ values for all 10 fictitious project executions of table 3.2 for p varying between 10% and 90%, with increments of 10%, are also presented in table 3.4. The calculation of these $\mathrm{SPI(t)}_p^{\mathcal{F}_1}$ values was not as straightforward as in the case of the single feeding path, since the subnetwork in F_1 is not serial. Instead, they required the $\mathrm{PV}_t^{\mathcal{F}_1}$ and $\mathrm{EV}_t^{\mathcal{F}_1}$ curves to be calculated first. The $\mathrm{PV}_t^{\mathcal{F}_1}$ and $\mathrm{EV}_t^{\mathcal{F}_1}$ curves for the first three fictitious project executions are shown in figure 3.9. From the $\mathrm{PV}_t^{\mathcal{F}_1}$ and $\mathrm{EV}_t^{\mathcal{F}_1}$ curves, $\mathrm{ES}_t^{\mathcal{F}_1}$ can then be calculated and consequently, the $\mathrm{SPI(t)}_p^{\mathcal{F}_1}$ values can be found.

As an example, figure 3.10 shows two grey shaded areas divided by the 0% quantile (minimum) of the empirical distribution of $\mathrm{SPI(t)}^{\mathcal{F}_1}$. This sample quantile is also presented in table 3.5. The lightgrey area represents the sample values for the runs indicated by * in table 3.4. The darkgrey area represents all $\mathrm{SPI(t)}^{\mathcal{F}_1}$ values which are lower than the 0% sample quantiles found from the set of runs indicated with *. The three lines in figure

Simulation run	Subnetwork percentage complete									Path duration	
	10%	20%	30%	40%	50%	60%	70%	80%	90%	3-5-6-7	3-5-6-10
* 1	0.36	0.45	0.55	0.64	0.64	0.54	0.51	0.56	0.61	19	21
2	0.16	0.24	0.31	0.38	0.39	0.38	0.39	0.43	0.47	25	27
3	0.97	0.86	0.74	0.63	0.54	0.49	0.49	0.53	0.57	23	19
* 4	0.54	0.68	0.82	0.96	1.00	0.90	0.81	0.85	0.89	12	15
5	0.96	0.84	0.71	0.58	0.58	0.60	0.60	0.60	0.59	24	15
* 6	1.02	1.07	1.13	1.18	1.17	0.78	0.65	0.66	0.68	20	14
* 7	1.02	1.07	1.13	1.18	1.17	0.79	0.69	0.77	0.85	14	15
* 8	0.50	0.52	0.53	0.54	0.58	0.59	0.63	0.70	0.76	17	15
* 9	1.00	1.00	1.00	1.00	1.00	0.65	0.65	0.75	0.84	15	15
* 10	1.00	1.00	1.00	1.00	1.00	0.82	0.76	0.78	0.80	17	12

* These runs are used to produce the sample quantile of table 3.5

Table 3.4: SPI(t) calculated for the subnetwork in F_1

3.10 again represent the first three fictitious project executions of table 3.4. The first fictitious project execution uses 100% of its slack for the path 3-5-6-10, but does not consume all of the slack for the path 3-5-6-7. Since there is not more than 100% consumption of the subnetwork slack, the critical path is not endangered in this fictitious project execution, as shown by the full line ending in the lightgrey area in figure 3.10. The second and the third fictitious project execution consume more than 100% of the slack in either the path 3-5-6-7 or the path 3-5-6-10. They are correctly classified by our tolerance limits, since they end up below the 0% sample quantile in the darkgrey area. The use of the 0% sample quantile would have correctly signalled that F_1 was likely to interrupt or endanger the critical path for runs 2 and 3. When figure 3.10 is compared to figures 3.6 and 3.7, we notice that the false warning signal for run 1 has been removed through the use of statistical tolerance limits on the subnetwork buffer. The use of the feeding path buffer could have possibly led to an overreaction, while the use of the subnetwork buffer would have only shown genuine warning signals, with respect to endangering the critical path of the project.

We have illustrated in this section that the statistical tolerance approach to setting limits on EVM/ES metrics allows the project manager to control the consumed slack within a subnetwork. In doing so, the number of control points throughout the project, and correspondingly the project control effort, can be reduced. Applying the bird's eye perspective of EVM/ES on these specific subnetworks in the project, alongside the critical path, will improve project schedule control. We will further formalize the EVM-FPB and EVM-SNB approaches in section 3.2.3-3.2.4 and test their performance in the remainder of this paper.

3.2.3 Combining EVM with feeding path buffers (EVM-FPB)

In this section, we formalize the EVM-FPB procedure to combine the use of EVM with feeding path buffers in two parts. In section 3.2.3, a recursive method is introduced to determine the number of feeding path buffers and their place in the project schedule. Section 3.2.3 calculates the tolerance levels for the EVM/ES performance metrics to be used, while controlling the project buffer and the feeding path buffers.

Table 3.5: SPI(t) sample quantiles for the subnetwork in F_1

Sample quantile	Subnetwork percentage complete								
	10%	20%	30%	40%	50%	60%	70%	80%	90%
0%	0.42	0.54	0.54	0.58	0.63	0.62	0.61	0.61	0.61

Algorithm 1: $AddFeedingPath(i)$

$\mathcal{E} = \mathcal{E} \setminus \{i\}$
```
// Non-critical predecessors
```
for $k \in \mathcal{P}_i \cap (\mathcal{E} \setminus \mathcal{C})$ **do**
 if $(FT_k \neq ST_i) \wedge (\mathcal{F}_j \neq \{\})$ **then**
  $j = j + 1$
  $\mathcal{F}_j = \{\}$
 end
 $\mathcal{F}_j = \mathcal{F}_j \cup \{k\}$
 $AddFeedingPath(k)$
end
if $(\mathcal{E} \setminus \mathcal{C} \neq \{\}) \wedge (\mathcal{F}_j \neq \{\})$ **then**
 $j = j + 1$
 $\mathcal{F}_j = \{\}$
end
```
// Critical predecessors
```
for $k \in \mathcal{P}_i \cap \mathcal{C}$ **do**
 $AddFeedingPath(k)$
end

Recursive addition of feeding path buffers

The algorithm starts with an earliest start schedule of the project network $G(\mathcal{N}, \mathcal{A})$ in which 1 is used as the index for the dummy start node and N is the index for the dummy end node. For each activity i in the earliest start schedule, the slack s_i is calculated as the difference between its latest start time and its earliest start time. Consequently, a recursive search is invoked to determine the number and place of the feeding path buffers. $\mathcal{C} = \{i \in \mathcal{N} | s_i = 0\}$ will be used to denote the set of activities that lie on the critical path and the set $\mathcal{E}$ is used to store the activities that are still eligible for being passed as argument to the recursive function presented in algorithm 1.

Algorithm 1 shows the recursive function to add a total of J feeding path buffers F_j, which should be controlled using EVM/ES performance metrics. After initialisation ($j = 1$; $\mathcal{F}_j = \{\}$; $\mathcal{E} = \mathcal{N} \setminus \{1\}$) the dummy end node should be given as argument to the function $AddFeedingPath$ ($AddFeedingPath(N)$). Recursively, each node i in the project network will then be visited and non-critical activities will be added to an appropriate feeding path buffer F_j. The algorithm first visits all non-critical predecessors of the node i. The predecessor k will be added to the current feeding path F_j if the start time of i is equal to the finish time of k. Otherwise, a new empty buffer F_{j+1} will be created and the predecessor activity

k will be added to this new buffer. Secondly, all critical activities will also be visited in the current recursive level in order to run backwards along the critical path until the dummy start node is reached. For our experiments we will always choose the non-critical predecessor k that leads to the path with the longest sequence of activities first. More precisely, this is recursively done by selecting the predecessor $k \in \mathcal{P}_i \cap (\mathcal{E} \setminus \mathcal{C})$ that will lead to the longest path when continuing the recursive search. Searching for the longest possible paths for each feeding buffer leads to a reduction in the number of feeding path buffers in comparison with the random selection of a path for each feeding buffer. However, the recursive search can be applied differently without loss of generality, possibly leading to a higher number of buffers, and hence, a higher number of control points during project progress. Figure 3.11 shows the addition of $J = 3$ feeding path buffers to the project example network using our recursive search algorithm.

Setting tolerance limits for EVM-FPB

The EVM-FPB project control approach extends the use of the traditional EVM/ES project control approach to control points on each buffer, aiming at monitoring the schedule progress measured by the SV(t) and SPI(t) performance metrics. The first control point is always situated at the project buffer (or dummy end node) and is used to control the critical path. To that end, either $\mathrm{SV(t)}_t^{\mathcal{C}}$ or $\mathrm{SPI(t)}_t^{\mathcal{C}}$ should be monitored dynamically during the project execution against a tolerance level to act as a threshold for early warning signals. Suppose we have a project deadline that can be written as:

$$\frac{(100 + \%pb)}{100} \mathrm{PD}_C \qquad (3.3)$$

with $\%pb$, the project buffer, i.e. the maximum allowable delay expressed as a percentage of the planned duration of the project.

Based on the $\%pb$ value, the tolerance limits for $\mathrm{SV(t)}_t^{\mathcal{C}}$ can be expressed as a function of the time t, where t varies between the starting time of the first activity on the critical path until the finish of the last activity of the critical path, as

$$\frac{-\%pb}{100 + \%pb} t \qquad (3.4)$$

Likewise, the tolerance limit for $\mathrm{SPI(t)}_t^{\mathcal{C}}$ can be set as

$$\frac{100}{100 + \%pb} \qquad (3.5)$$
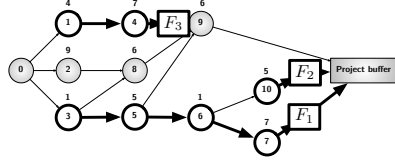
Figure 3.11: Feeding path buffers for the example project network
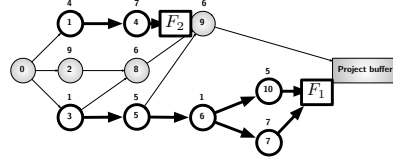
Figure 3.12: Subnetwork buffer for the example project network

which shows that its value is independent from the time t.

The feeding path buffers act as additional control points to be controlled alongside the critical path. Analogously to the project buffer, if the slack of a feeding path F_j is written as a percentage of the planned duration of the feeding path: $\%fpb = 100 s_j^* / \mathrm{PD}_{F_j}$, then the tolerance limits are $-\%fpb/(100 + \%fpb)t$ and $100/(100 + \%fpb)$ for $\mathrm{SV(t)}_t^{\mathcal{F}_j}$ and $\mathrm{SPI(t)}_t^{\mathcal{F}_j}$ respectively, where t is the time since the planned start of the first activity of F_j.

3.2.4   Combining EVM with subnetwork buffers (EVM-SNB)

In this section, we formalize the EVM-SNB procedure using control points on subnetworks. In section 3.2.4, a recursive method is introduced to determine the subnetwork buffers and their place in the baseline schedule. Section 3.2.4 calculates the tolerance levels for the EVM/ES performance metrics to be used while controlling the critical path and the subnetwork buffers.

Recursive addition of subnetwork buffers

Similar to the EVM-FPB approach the procedure starts with an earliest start schedule, and relies on the recursive search of algorithm 2 to determine and add the subnetwork buffers.

After initialisation ($\mathcal{F}_1 = \{\}$, $\mathcal{E} = \mathcal{N} \setminus \{1\}$) the dummy end node and $j = 1$ should be given as arguments to the function $AddSubNetwork$ ($AddSubNetwork(N, 1)$) to produce a total of J^* subnetwork buffers. For a node i in the activity network, the recursive search will first visit all non-critical predecessors k and will add them in a subnetwork buffer F_j. If all non-critical predecessors have been visited, the search is continued along the critical path and a new subnetwork buffer F_{j+1} is created as long as there are still non-critical activities eligible, given by $\mathcal{E} \setminus \mathcal{C} \neq \{\}$, and $\mathcal{F}_j$ is

---

Algorithm 2: $AddSubNetwork(i, j)$

$\mathcal{E} = \mathcal{E} \setminus \{i\}$;
`// Non-critical predecessors`
for $k \in \mathcal{P}_i \cap (\mathcal{E} \setminus \mathcal{C})$ **do**
   | $\mathcal{F}_j = \mathcal{F}_j \cup \{k\}$
   | $AddSubNetwork(k, j)$
end
`// Critical predecessors`
for $k \in \mathcal{P}_i \cap \mathcal{C}$ **do**
   | **if** $(\mathcal{E} \setminus \mathcal{C} \neq \{\}) \wedge (\mathcal{F}_j \neq \{\})$ **then**
      | $\mathcal{F}_{j+1} = \{\}$
      | $AddSubNetwork(k, j+1)$
   | **else**
      | $AddSubNetwork(k, j)$
   | **end**
end

---

not empty. Figure 3.12 shows how $J^* = 2$ subnetwork buffers are added to the example project network.

The number of subnetwork buffers J^* will represent a significant decrease in the number of control points, compared to the number of feeding path buffers J as calculated by algorithm 1. This difference will be the largest for project network structures close to parallel. In the extreme, for a complete parallel project, the procedure outlined in algorithm 2 will result in $J^* = 1$, while algorithm 1 will produce $J = N - 2$ feeding path buffers.

Setting tolerance limits for EVM-SNB

The tolerance limits to control the subnetwork buffers in the EVM-SNB procedure are calculated from sample quantiles of a Monte Carlo simulation performed prior to the project progress as illustrated in section 3.2.2. These sample quantiles can be calculated for any EVM/ES performance metric Y and for any α^{th} quantile of the empirical distribution function approximated from the simulation. We refer to [17] and [9] for more detail on the calculations of the αth quantile from the simulated set of observations $\{Y_{i,p}^x | i \in \mathcal{M}\}$, where $Y_{i,p}^x$ is used to denote the observation of an EVM/ES performance metric Y, calculated for a set of activities x in a run i $(1 \leq i \leq n)$ at p percentage of the project complete. The αth quantile $\hat{Q}(\alpha)_p$ at a percentage complete p, is the number q for which the probability of drawing a value from Y_p less than q is at most α.

In order to find appropriate runs to form the reference set $\mathcal{M}$, a Monte

Carlo simulation is performed. From the simulation output, all runs for which the delays in the subnetwork buffers exceed the slack of that subnetwork are removed, and the remaining simulation runs form the reference $\mathcal{M}$ from which statistical tolerance limits can be calculated. In the EVM-SNB approach, tolerance limits need to be calculated for the set of activities x, equal to $\mathcal{C}$ for the critical path and x equal to $\mathcal{F}_j$ for the subnetwork buffers ($j \in \{1, \ldots, J^*\}$).

The distributional input to the Monte Carlo simulation should be chosen such that it accurately reflects the real uncertainty that is encountered in practice in project activity durations. Probability density functions for all activities can be obtained from expert judgement or from calibration to historical data and the specific settings for the uncertainty modelling in our study will be introduced in section 3.3.

3.3   Computational experiments

In this section, we introduce an extensive simulation experiment in which we compare the performance of the EVM-FPB and EVM-SNB control approaches against EVM-1PB, EVM-STL, EVM-LPB. Section 3.3.1 will formally present the EVM-1PB, EVM-STL, EVM-LPB control approaches. Monte Carlo simulation experiments are well established in the project management research literature [4, 40, 48]. The project dataset and the dynamic project progress model for these experiments will be introduced in section 3.3.2. Section 3.3.3 will discuss the input modelling for the activity durations used in the. Finally, we will elucidate the measures used to quantify and compare the performance of the project control approaches in section 3.3.4.

3.3.1   EVM project control procedures

The EVM-FPB and EVM-SNB procedures combine the bird's eye perspective of EVM with cleverly assigned project control points to incorporate baseline schedule information. In our computational experiments, we investigate whether this combination is likely to improve the performance of the project control process in protecting the project makespan, without excessively increasing the effort spent in the process. We will therefore compare the introduced project control procedures with three alternative EVM control methods encountered in literature. We will first give a short overview of the alternative approaches in section 3.3.1 and second, we will present the main characteristics of the five project control procedures and the details on their implementation in this study in section 3.3.1.

Introduction from the project control approaches from literature

The traditional **EVM-1PB** procedure relies on the earned schedule performance measures $SV(t)$ and $SPI(t)$ calculated for the complete project. It is labelled as EVM-1PB, since it uses only one control point that is situated at the project buffer. However, unlike the EVM-FPB and EVM-SNB procedures, this control point includes the performance of all the activities (i.e. the set $\mathcal{N}$) and not only those that lie on the critical path. If the project deadline is written as in equation 3.3, then tolerance limits can be calculated for $SV(t)_t^{\mathcal{N}}$ and $SPI(t)_t^{\mathcal{N}}$ using equations 3.4 and 3.5. A warning signal for schedule control is said to be produced by the EVM-1PB procedure at a time t if $SV(t)_t^{\mathcal{N}}$ or $SPI(t)_t^{\mathcal{N}}$ are lower than their tolerance limits. This control approach for projects was previously discussed by [2] as the use of "Target Performance Charts", and implemented by [44] in his study on top-down project control using EVM/ES.

The **EVM-STL**, previously described by [9], also uses only one control point at the project buffer that includes the performance of all activities (the set $\mathcal{N}$). However, the difference lies in the calculation of the tolerance limits on the EVM/ES performance metrics at each p percentage project completion ($SV_p^{\mathcal{N}}$, $SPI_p^{\mathcal{N}}$, $SV(t)_p^{\mathcal{N}}$ and $SPI(t)_p^{\mathcal{N}}$). Similar to the procedure outlined in section 3.2.4, this is done by estimating sample quantiles ($\hat{Q}(\alpha)_p$) from the Monte Carlo simulation runs in a set $\mathcal{M}$, prior to the start of the execution phase of the project. A warning signal is said to be produced when at any percentage complete p during the execution of the project, the measured EVM/ES performance metrics are lower than their corresponding tolerance limits.

Inspired by the observation that the EVM/ES metrics are fully reliable for serial networks, the longest path EVM (**EVM-LPB**) control approach was proposed by [23]. It also employs only one control point at the project buffer, but the EVM/ES performance metrics are now only calculated for the activities lying on the dynamic longest path $\widehat{\mathcal{C}}_t$. This dynamic longest path can be perceived as a dynamic (i.e. dependent of the time t) estimate of the real critical path of the project, and therefore might be different for different values of t. It assumes that at each time instance the critical path method calculations are made to establish the longest path in the schedule, given the baseline estimates for the durations of the activities that have not been finished and the real durations for those that have been finished. This procedure require full knowledge at the activity level of the project and is therefore classified as closest to the CPM method in figure 3.2 with respect to the control effort. If the project deadline is written as in equation 3.3, then tolerance limits can be calculated for $SV(t)_t^{\widehat{\mathcal{C}}_t}$ and $SPI(t)_t^{\widehat{\mathcal{C}}_t}$ using equations 3.4 and 3.5. A warning is said to be signalled by the EVM-LPB

control procedure if the current schedule performances are lower than their tolerance limits.

Implementation and comparison

Table 3.6 summarizes the main characteristics of the five project control procedures that are compared in this computational experiment. For EVM-1PB, EVM-STL and EVM-LPB the main references are provided in the second column and for EVM-FPB and EVM-SNB, we refer to the corresponding algorithms presented this paper. The third column of table 3.6 presents the number of control points $J^{\#}$ for each control procedure. For the EVM-LPB approach, this amounts to N, which indicates that at each review period $k = 1 \ldots K$ all N activities of the project need to be consulted in order to produce a warning signal.

In order to present an objective comparison of the five project control procedures in this paper, we have restricted their implementation such that they use a single EVM/ES performance metric, so that for all of them Y^x equals $\mathrm{SPI(t)}^x$. Moreover, a total number of control periods K over the lifetime of the project was kept fixed for each project control procedure. Therefore, the time t at which the $\mathrm{SPI(t)}^x$ is calculated for a set of activities x is chosen such that it coincides with $K = 19$ distinct intervals along the project complete axis ($t = t(p), \forall p = 5\%, 10\%, \ldots 95\%$). The fourth column of table 3.6 presents the sets of activities x for each control procedure and the corresponding tolerance limits are shown in the fifth column.

The reader should note that an additional index α has also been added to the tolerance limits of the EVM-1PB, EVM-FPB and EVM-LPB project control procedures. $\%pb_{\alpha}$ and $\%fpb_{\alpha}$ thereby respectively represents a "virtual" project and feeding path buffer that allows the tolerance limits for these control procedures to be calculated with more flexibility. This was done to ensure a maximal *effectiveness* for all control procedures in our computational experiment, in order to allow us to compare them based on their *efficiency* and *control effort*. We will go into more detail concerning these performance measures in section 3.3.4.

3.3.2   Dynamic project progress model

The simulation experiment, deployed in this computational experiment, uses the project network dataset of [43] to test the performance of the control approaches. The 900 networks were generated using RanGen [10, 46], and ensure a maximal diversity in terms of network structure within the dataset. The project progress for this research is performed using Monte

Name	References/Algorithm	$J^{\#}$	Sets of activities	Tolerance limits
EVM-1PB	[2, 44]	1	$x = \mathcal{N}$	$\dfrac{100}{100 + \%pb_\alpha}$
EVM-STL	[9]	1	$x = \mathcal{N}$	$\hat{Q}(\alpha)_p$
EVM-FPB	$AddFeedingPath$	$J+1$	$x = \begin{cases} \mathcal{C} & j=1 \\ \mathcal{F}_j & j=2\dots J+1 \end{cases}$	$\left\{\begin{array}{l} \dfrac{100}{100+\%pb_\alpha} \\ \dfrac{100}{100+\%fpb_\alpha} \end{array}\right.$
EVM-SNB	$AddSubNetwork$	$J^{*}+1$	$x = \begin{cases} \mathcal{C} & j=1 \\ \mathcal{F}_j & j=2\dots J^{*}+1 \end{cases}$	$\hat{Q}(\alpha)_p$
EVM-LPB	[23]	N	$x = \hat{\mathcal{C}}_t$	$\dfrac{100}{100+\%pb_\alpha}$

Table 3.6: Overview of the control procedures discusses in the computational experiments

Carlo simulations in P2 Engine [45], a scripting-enabled project scheduling and control tool, previously used by [43] and [8, 9]. Project progress is simulated from fictitious project executions, where actual durations for the activities deviate from the baseline estimate durations given the applied modelling of the input uncertainty. The EVM/ES control data calculated during the fictitious project executions are based on the assumption that both PV and EV accrue linearly for an activity during its execution.

3.3.3   Simulation input modelling

In order to test and validate the performance of the different control approaches, two Monte Carlo simulations are needed for each project. After establishing the subnetwork buffers through algorithm 2, a first simulation will be used to calculate the tolerance limits for the EVM-SNB approach as described in section 3.2.4. Subsequently, a second simulation allows us to calculate the performance of the project control approach, as will be discussed in section 3.3.4. Both these simulations are run with appropriate settings for the distributions to model uncertainty on activity durations.

In this research, we have chosen to model uncertainty at the activity level by both variation and risk. Variation on the activity level of a project can be expressed by applying probability density functions on the real activity durations and will be discussed in section 3.3.3. In addition to variation, project management literature often recounts other types of uncertainty occurring during execution [26]. Risk, as discussed in section 3.3.3, is regularly used to denote (un)likely events that can have a potentially disastrous impact on multiple activities in the project and the overall project objectives. This type of uncertainty is often overlooked in project control simulation, since there is no generalised way to model correlation structures or the occurrence of events. A solution to overcome this shortcoming is provided by the linear association approach that looks at statistical dependencies between activities in the project [37]. In our experimental model, we will incorporate both variation in the form of activity-specific probability density functions for the duration and risks by using linear association, as will be discussed in section 3.3.3.

Variation model

We will model variation on the activity durations through probability density functions from the generalised beta distribution family. Generalised beta functions have long been associated with PERT estimates in project management [27] for their ability to accurately mimic the behaviour of the

random input processes driving the system [20]. The probability density function for a random variable X is

$$f_X(x|a, b, \theta_1, \theta_2) = \begin{cases} \frac{\Gamma(\theta_1+\theta_2)}{\Gamma(\theta_1)\Gamma(\theta_2)} \frac{(x-a)^{\theta_1-1}(b-x)^{\theta_2-1}}{(b-a)^{\theta_1+\theta_2-1}} & \text{if } a \le x \le b \\ 0 & \text{if } x < a \lor b < x \end{cases} \quad (3.6)$$

where $\Gamma()$ denotes the gamma function and θ_1 and θ_2 are the shape parameters. [1] suggest an approximation for fitting the beta distribution to historical data or subjective estimates, when uncertainty is characterized by a long tail, as often described in project management. This approximation produces estimates for the beta shape parameters through a single auxiliary parameter, calculated from PERT-style three point estimates. In [43], these approximations have been used to test the performance of EVM time-forecasting methods on a large fictitious dataset. However, since large variation instances for fixed values for the average of the activity duration can not be produced using the approximation of [1], we propose the use of a parameter vector $\boldsymbol{\omega} = (a, b, m, \mu)$, with estimates for the minimum, the maximum, the mode and the mean of the distribution expressed as a fraction of the baseline estimate duration $\widehat{d_i}$. In doing so, we can write the probability density function of the random duration D_i of activity i as:

$$f_{D_i}\left(d_i|a\widehat{d_i}, b\widehat{d_i}, \theta_1(\boldsymbol{\omega}), \theta_2(\boldsymbol{\omega})\right) = f_X\left(x|a, b, \theta_1(\boldsymbol{\omega}), \theta_2(\boldsymbol{\omega})\right)\widehat{d_i} \quad (3.7)$$

$$= \beta(\boldsymbol{\omega})\widehat{d_i} \quad (3.8)$$

with $x = d_i/\widehat{d_i}$ and

$$\begin{cases} \theta_1(\boldsymbol{\omega}) = -\frac{(b+a-2m)(a-\mu)}{(m-\mu)(a-b)} & (3.9a) \\ \\ \theta_2(\boldsymbol{\omega}) = \frac{(b+a-2m)(b-\mu)}{(m-\mu)(a-b)} & (3.9b) \end{cases}$$

where the shape parameters (equations 3.9) are independent from the baseline estimate $\widehat{d_i}$ and can be found by solving the set of equations 3.10 through substitution.

$$\begin{cases} \mu = \frac{\theta_1 b + \theta_2 a}{\theta_1 + \theta_2} & (3.10a) \\ \\ m = \frac{(\theta_1-1)b + (\theta_2-1)a}{\theta_1 + \theta_2 - 2} & (3.10b) \end{cases}$$

Risk model

We have chosen to model risk using the linear association approach of [37]. In the absence of a generalised way to model the potential impact of events

on multiple activities through a correlation matrix, [37] and [36] suggested to use a positive random variable B to model dependencies between activities. The bias term B is easily perceived as a consistent over- or underestimation of activity duration, or a project-wide effect of uncertain events. In a critical chain context, [35] investigated the effect of a systemic error on setting an optimal project buffer. A set of positive random variables Z_i are said to be linearly associated if $Z_i = BY_i$, where $\{Y_i\}$ is a set of positive random variables with the same cardinality. [37] show that the inclusion of a bias term B is indispensable when a project model is calibrated to historical data of nine projects. According to [37], B is in practice influenced by both additive and multiplicative causes and is therefore assumed to follow a lognormal distribution. We reason that the generalised beta family of distributions has an advantage in simulation input modelling over the lognormal distribution. It is a more intuitive distribution, which originates from practical applications, and which allows $f(x) = 0$ for $x < \infty$. The lognormal distribution has $\lim_{x \to \infty} f(x) = 0$, a theoretical limit that is not really intuitive (every positive duration having non-zero probability of occurring), nor is it practically reproducible in a computerized simulation experiment.

Combined input model

For each run in the simulation experiment, numbers will be generated from the generalised beta distribution to model both activity variation and risk. To that purpose, a bias term B will first be sampled for each run in the simulation experiment from a generalised beta probability distribution $\beta(\boldsymbol{\omega}_R)$. Subsequently, for all activities $i = 1 \ldots N$ in this fictitious project execution, an activity duration d_i is drawn from the distribution $\beta(\boldsymbol{\omega}_V)B\widehat{d_i}$. This distribution can be seen as a generalised beta distribution $\beta(\boldsymbol{\omega}_V)$ with a independently chosen parameter vector that represents the variation $\boldsymbol{\omega}_V$, which is applied to a scaled baseline estimate ($B\widehat{d_i}$) of the real activity duration.

These assumptions and characteristics allow the use of a limited number of parameter vectors $\boldsymbol{\omega}$ (either $\boldsymbol{\omega}_R$ or $\boldsymbol{\omega}_V$) in our computational experiment. Table 3.7 depicts the selection of parameter vectors from which $\boldsymbol{\omega}_R$ and $\boldsymbol{\omega}_V$ are chosen in the simulations. Preliminary calculations and visualisation in the statistical programming language R [30] provided validation for the suggested parameter vectors of table 3.7 during our research. The first set of parameter vectors ($\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$ and $\boldsymbol{\omega}_3$) are chosen to emulate practical distributions, with realistically shaped tails that represent scenarios were the average activity duration is lower than, equal to or higher than its baseline duration. The standard deviations of $\boldsymbol{\omega}_1$, $\boldsymbol{\omega}_2$ and $\boldsymbol{\omega}_3$ equal 0.38, 0.30 and 0.38 respectively and hence, an appropriate choice for both $\boldsymbol{\omega}_R$ and $\boldsymbol{\omega}_V$

General performance experiment			Sensitivity experiment					
			Mean			Standard deviation		
ω	(a, b, m, μ)	σ	ω	(a, b, m, μ)	σ	ω	(a, b, m, μ)	σ
ω_1	$(0.1, 1.2, 0.7, 0.6)$	0.38				$\omega_{2_{s1}}$	$(0.2, 4, 0.40, 1)$	0.50
			$\omega_{2_{\mu1}}$	$(0.2, 4, 0.51, 0.7)$	0.3	$\omega_{2_{s2}}$	$(0.2, 4, 0.70, 1)$	0.42
			$\omega_{2_{\mu2}}$	$(0.2, 4, 0.90, 1.0)$	0.3	$\omega_{2_{s3}}$	$(0.2, 4, 0.85, 1)$	0.35
ω_2	$(0.2, 4, 0.9, 1)$	0.30	$\omega_{2_{\mu3}}$	$(0.2, 4, 1.22, 1.3)$	0.3	$\omega_{2_{s4}}$	$(0.2, 4, 0.92, 1)$	0.28
			$\omega_{2_{\mu4}}$	$(0.2, 4, 1.57, 1.6)$	0.3	$\omega_{2_{s5}}$	$(0.2, 4, 0.96, 1)$	0.20
			$\omega_{2_{\mu5}}$	$(0.2, 4, 1.89, 1.9)$	0.3	$\omega_{2_{s6}}$	$(0.2, 4, 0.98, 1)$	0.15
ω_3	$(0.9, 4, 1.3, 1.4)$	0.38						

Table 3.7: Parameter vectors for the generalised beta distribution

from this set of parameter vectors is sufficient to simulate large samples for $d_i/\widehat{d_i}$ with standard deviations up to 0.5. Samples of activity durations with standard deviation up to 0.5 were found to adequately represent the largest variation instances from the sizeable collection of empirical data provided by [3]. For more than 90% of the projects found in the database of [3], the sample standard deviation was found to be less than 0.5. This database currently consists of 51 real-life projects with actual progress data and is very diverse with respect to sector, planned duration and budget at completion characteristics of the projects. The second set of parameter vectors ($\boldsymbol{\omega}_{2\mu_1}, \ldots \boldsymbol{\omega}_{2\mu_5}$) is derived from $\boldsymbol{\omega}_2$ to test the influence of a change in the mean of the distribution. If the mean is changed, the mode needs to change accordingly to keep a realistic shape for the distribution function, and to ensure that the standard deviation remains the same as for $\boldsymbol{\omega}_2$, $\sigma = 0.3$. The last set of parameter vectors ($\boldsymbol{\omega}_{2s_1}, \ldots \boldsymbol{\omega}_{2s_6}$) will be used to explore the influence of a change in the standard deviation. Is it only necessary to change the mode of the generalised beta distribution to effectively change the standard deviation, while keeping the mean $\mu = 1$ unchanged.

3.3.4   Performance measurement

In the computational results section, we will compare the *efficiency* of all project control methods under such conditions that the effectiveness is always equal to 100%. To that purpose, we also measure the *control effort* of each project control method as a proxy of the amount of time a project manager has to spend in monitoring project progress using each method. Both the efficiency and control effort are quantified in the following two subsections.

Efficiency

The efficiency of project control methods have been defined earlier by [44] using a corrective actions model that is used when projects are in danger. However, since in the current study no corrective actions are implemented, we have adopted the efficiency definition of project control. Instead, we rely on a model that assumes that each control method is always able to detect deviations from the plan. More precisely, we have set the α parameter, used for setting the tolerance limits of the EVM-1PB, EVM-FPB and EVM-LPB methods using virtual buffers or as sample quantiles for the EVM-STL and EVM-SNB methods, to such a value that the *detection performance* of each control method is equal to 100% . The detection performance

has been used in the study of [9] and is equal to:

$$\mathrm{P}[S|RD > \delta_{\mathcal{N}}] = \frac{\sum\limits_{i=1}^{n} \Bbbk_i(RD > \delta_{\mathcal{N}})\Bbbk_i(S)}{n}$$

Here, $\Bbbk_i(A)$ denotes the identity operator, which returns 1 if statement A is true and 0 if statement A is false for simulation run i. Moreover, a signal can be produced by any of the control points in the selected control procedure, so S is a logical disjunction over all control points:

$$S = \bigvee_{j=1}^{J^{\#}} S_j$$

where $J^{\#}$ represents the number of control points in the selected control procedure and S_j represents whether or not a signal is produced in control point j. Similarly, we can calculate the *probability of encountering false warning signals* [9] as:

$$\mathrm{P}[S|RD \leq \delta_{\mathcal{N}}] = \frac{\sum\limits_{i=1}^{n} \Bbbk_i(RD \leq \delta_{\mathcal{N}})\Bbbk_i(S)}{n}$$

The efficiency of a control procedure expresses the probability that the project duration will exceed its final deadline, given the event that a signal S is reported. Therefore, Bayes' theorem allows us to calculate the efficiency of the project control procedure from the detection performance and the probability of of encountering false warning signals as:

$$\begin{aligned} \mathrm{P}[RD > \delta_{\mathcal{N}}|S] &= \frac{\mathrm{P}[S|RD > \delta_{\mathcal{N}}]\,\mathrm{P}[RD > \delta_{\mathcal{N}}]}{\mathrm{P}[S]} \\ &= \frac{\mathrm{P}[S|RD > \delta_{\mathcal{N}}]\,\mathrm{P}[RD > \delta_{\mathcal{N}}]}{\mathrm{P}[S|RD > \delta_{\mathcal{N}}]\,\mathrm{P}[RD > \delta_{\mathcal{N}}] + \mathrm{P}[S|RD \leq \delta_{\mathcal{N}}]\,\mathrm{P}[RD \leq \delta_{\mathcal{N}}]} \end{aligned}$$

(3.11)

where $\mathrm{P}[RD > \delta_{\mathcal{N}}]$ and $\mathrm{P}[RD \leq \delta_{\mathcal{N}}]$ depend on the probabilistic outcome of the project simulations. This is affected by both the given the simulation input modelling (section 3.3.3) and the underlying project network structure.

Control effort

For each of the control procedures discussed in this computational experiment, we will also directly measure the control effort. This measure counts

General Performance
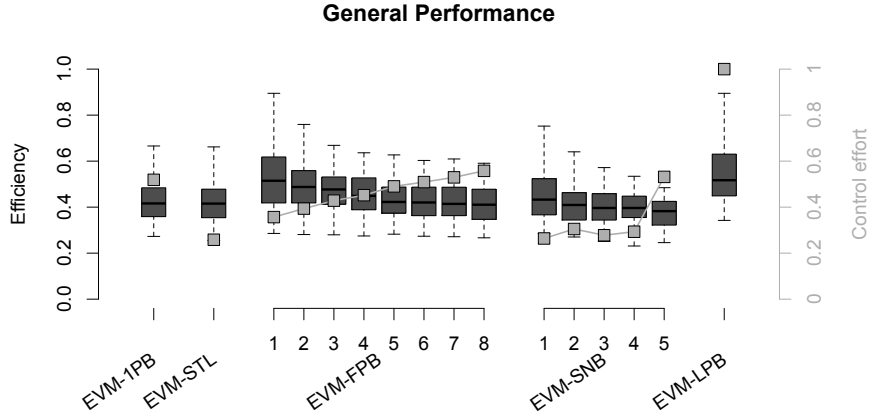
Figure 3.13: General comparison of the project control procedures (for $\boldsymbol{\omega}_R = \boldsymbol{\omega}_V = \boldsymbol{\omega}_2$)

the relative number of times a signal is reported by the project control approaches, and consequently the relative number of times that the activity level performance needs to be controlled during the execution of the project by drilling down the WBS [44]. For each project control approach, we therefore calculate the control effort as:

$$\frac{\sum_{i=1}^{n} \sum_{k=1}^{K} \bigvee_{j=1}^{J^{\#}} S_{ijk}}{nK} \tag{3.12}$$

where $J^{\#}$ again represents the number of control points in the selected control procedure and S_{ijk} represents whether or not a signal is produced in simulation run i, for control point j at a time period k.

3.4 Results

Section 3.4.1 shows the general performance of the five project control approaches on the 900 project networks dataset. The network structure of a project can influence the performance of each of the project control approaches, as explored in section 3.4.2. The robustness of the different approaches is discussed in section 3.4.3.

ω_R and ω_V chosen from:

$E[d_i/\widehat{d_i}]$ $sd[d_i/\widehat{d_i}]$	(ω_1, ω_2) 0.7 0.38			ω_2 1.0 0.3			(ω_2, ω_3) 1.3 0.38		
	$\overline{J\#}$	Eq. 3.11	Eq. 3.12	$\overline{J\#}$	Eq. 3.11	Eq. 3.12	$\overline{J\#}$	Eq. 3.11	Eq. 3.12
EVM-1PB	1	23%	27%	1	43%	51%	1	78%	61%
EVM-STL	1	12%	31%	1	43%	27%	1	78%	23%
EVM-FPB	3.1	50%	26%	2.9	54%	37%	1.9	83%	43%
EVM-SNB	1.1	12%	33%	1.2	46%	27%	1.4	80%	25%
EVM-LPB	1	58%	100%	1	55%	100%	1	82%	100%

Table 3.8: General performance experiment comparison of project control procedures

3.4.1   General performance

The *general performance experiment* parameter vectors of table 3.7 were applied to the 900 project networks dataset and the simulation results (with $n = 10,000$ Monte Carlo simulation runs on each project and for every scenario) were analysed. The parameter vectors $(\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \boldsymbol{\omega}_3)$ were assigned to model both the dependencies between activities $(\boldsymbol{\omega}_R)$ and the variation $(\boldsymbol{\omega}_V)$ in the simulation experiment. We combined $\boldsymbol{\omega}_2$ with a parameter vector chosen from $\{\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \boldsymbol{\omega}_3\}$ and assigned one of them to $\boldsymbol{\omega}_R$ and the other to $\boldsymbol{\omega}_V$. In a second experiment, the role of the parameter vectors are switched and table 3.8 shows the averaged results over these simulations.

The mean and standard deviation for the resulting distribution of $d_i/\widehat{d_i}$ are given in the upper part of table 3.8. The lower part of table 3.8 presents the average number of control points used in each project control procedure $(\overline{J^\#})$, and the efficiency (equation 3.11) and control effort (equation 3.12) that has been recorded for each control procedure. For all project networks and each general performance simulation scenario, we have chosen the deadline $\widehat{RD}$ as 1.3PD. The "virtual" buffers $\%pb_\alpha$ and $\%fpb_\alpha$ and the αth sample quantiles were chosen such that the detection performance for all project control approaches could be kept fixed at 100%. The average number of control points used in a project control procedure $\overline{J^\#}$, is also calculated. For the EVM-FPB and EVM-SNB procedures, we have set the number of feeding path buffers and subnetwork buffers that are controlled simultaneously to those values such that the efficiency of the project control approach is maximised.

A general trend that can be seen from table 3.8 is that the efficiency of project control procedures seems to increase drastically for an increasing mean of the activity duration distribution. This observed increase is the most profound for the control procedures that deploy project specific statistical tolerance limits, calculated from the additional simulation runs prior to the execution of the project (EVM-STL, EVM-SNB). At the same time, the control effort for these project control approaches decreases, while the effort increases for the other control approaches (EVM-1PB, EVM-FPB) or remains stable (EVM-LPB). We observed that given an increased mean in activity durations, the probability of meeting a given deadline will decrease, since the mean of the distribution of the real project duration can be expected to shift to larger values [11]. In this scenario, it is then advantageous to use statistical tolerance limits, in order to produce warning signals at the right moments in the project, which ultimately reduces the project control effort.

Table 3.8 shows that the EVM-LPB method outperforms the other project control approaches in terms of the efficiency. This is due largely to a re-

duced probability of encountering false warning signals, which means that the EVM-LPB method is less likely to produce a signal when the project has a makespan that is smaller than the project deadline. This can be expected since the EVM-LPB approach uses dynamic activity level data to calculate the expected project duration at each time instance during the project execution. However, the increased efficiency comes at a high cost of maximal control effort (100%) because the activity level performance needs to be updated at each control period during the execution of the project. Consequently, from table 3.8, we conclude that the EVM-FPB approach is able to approximate the level of efficiency of the EVM-LPB approach, with much less the control effort. The maximal number of control points for the EVM-FPB approach can become very large, although table 3.8 shows that on average only 2-3 control points need to be monitored simultaneously to produce these results.

Figure 3.13 shows the observed efficiency and control effort for the second general performance scenario (where ω_R and ω_V are equal to ω_2). The number of control points is varied from low to high values for both the EVM-FPB and EVM-SNB methods. The boxplots show the distribution of the observed values of the efficiency, for which the axis is depicted to the left of the figure. The control effort is displayed using lightgrey boxes and is referenced against the axis on the right of figure 3.13. Overall, the same conclusions can be drawn from these as from table 3.8. The EVM-FPB approach approximates the efficiency of the EVM-LPB approach with a smaller corresponding control effort. For the EVM-FPB and EVM-SNB approaches, the performance in shown in relation to the number of control points used, denoted along the x axis. Figure 3.13 shows that the control effort increases for both the EVM-FPB and EVM-SNB approaches for increasing number of control points, while their efficiency decreases. This decrease in efficiency is due to the higher observed probability of encountering a false warning signal when multiple control points are used. For EVM-FPB, the increase in control effort is immediate, as soon as an additional control point is used alongside the control point placed at the project buffer to control the critical path. For the EVM-SNB approach, this effect only shows when more than 4 control points are used in combination with the control point at the project buffer.

In conclusion, table 3.8 and figure 3.13 show that for the two control procedures, EVM-FPB and EVM-SNB, an optimal efficiency and control effort is obtained when only the deterministic critical path is monitored at the project buffer. For the EVM-FPB approach, the efficiency is comparable to the best efficiency observed in our study, which required dynamic calculations of the longest path in the project (EVM-LPB), for only a fraction of the control effort. However, the recorded efficiency and the control effort for all studied control approaches were produced using an index α. This α allowed us to ensure that the project control approaches all have

**Impact of network structure on
the efficiency and control effort**
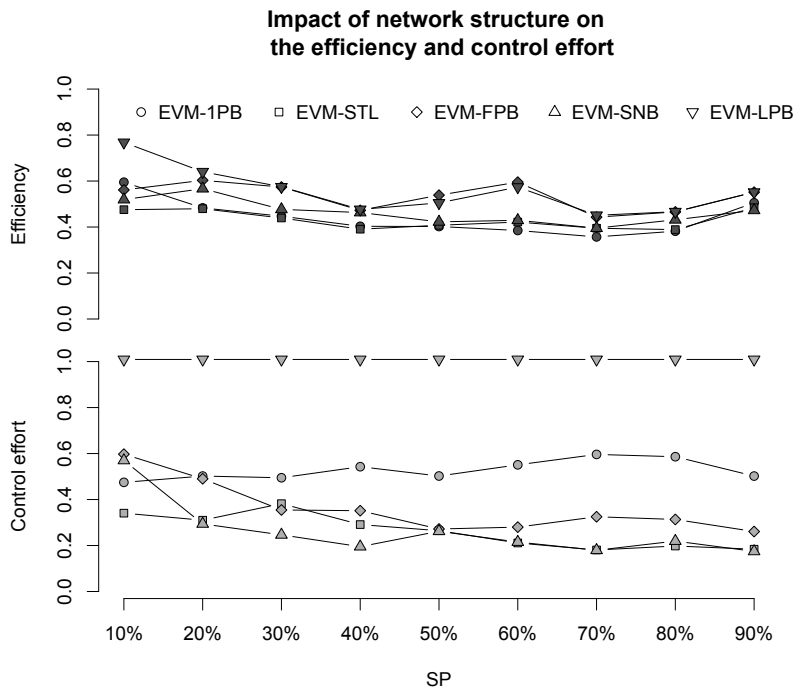
Figure 3.14: Influence of the serial/parallel indicator on the project control
procedures

a detection performance that equals 100%, while potentially reducing the
probability of encountering false warning signals. Thereby, the efficiency
of the control approaches could be maximised. The distributional choice
made for ω_R and ω_V in the first stage simulation however, might influ-
ence the sensitivity of the control methods in the latter stage. Consider a
real-life practical environment, where historical data is used to fuel a first
simulation and consequently, through statistical analysis, tolerance limits
(indexed by α) are calculated. If these tolerance limits are to be used in a
practical dynamic control process, it is of utmost importance to know how
sensitive these limits are to possible under- or overestimations in the dis-
tributional characteristics of the activity durations. In order to check this
sensitivity, or rather the robustness against the distributional assumptions
made, we conducted the robustness experiment in section 3.4.3.

3.4.2 Impact of network structure

The impact of the network structure of a project on the project control performance is studied in this section. The data from figure 3.13 is restructured in figure 3.14 to show the relation of the efficiency and control effort to the underlying network structure of the project. Therefore, the serial/parallel (SP) indicator is denoted on the x axis. The efficiency for all control approaches is shown at the top of figure 3.14 and the control effort is show at the bottom. The closer a project network is to a full serial network, the closer the SP is to 100%. If all activities are allowed to be executed in parallel, the project network is said to resemble a full parallel network and the SP is equal to 0%. Figure 3.14 confirms the earlier reported characteristic of EVM that it becomes less efficient to obtain activity level performance data (EVM-LPB) for more serial network structures [42]. The efficiency of the other project control approaches remains relatively stable for all SP values. The control effort however, remains stable only for the EVM-1PB approach, and decreases for all other project control approaches (EVM-STL, EVM-FPB, EVM-SNB) for increasing SP values. This also confirms the findings of [42, 44] with respect to the improvement in the project control capabilities of the EVM/ES system for more serial network structures.

3.4.3 Robustness experiment

In this robustness experiment, the sensitivity of the project control approaches to possible under- or overestimations of the characteristics of the underlying distribution for the activity durations is tested. All the presented results for the efficiency and the control effort are calculated using tolerance limits which are indexed by α which optimises the efficiency, while keeping the detection performance fixed at 100%. Therefore, the tolerance limits are subject to the result of a simulation run, prior to the project execution. Consequently, errors in the estimates for the standard deviation and the mean of the activity durations might have an impact on the performance on the tolerance limits derived from these estimates. Figure 3.15 shows the efficiency and project control effort for the EVM-1PB, EVM-STL, EVM-FPB, EVM-SNB and EVM-LPB control approaches. Two x-axes represent the changes in the mean and the standard deviations, which we will discuss separately in the remainder of this section. The two axis on the left of figure 3.15 are respectively used to reference the efficiency (at the top) and the control effort (at the bottom) against.

Mean The left side of figure 3.15 shows the influence of a change in the mean on the performance of the project control procedures. The tolerance limits (indexed by α) are calculated from a statistical analysis of the

Robustness of the project control approaches
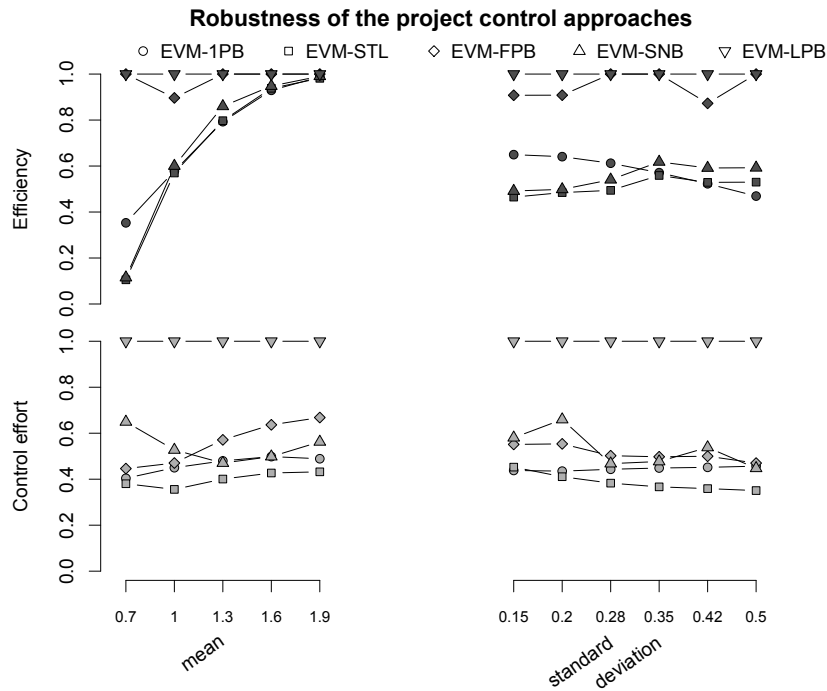
Figure 3.15: Influence of a change in the mean and the standard deviation on the project control procedures

simulation output where $\boldsymbol{\omega}_{2\mu_2}$ ($\mu = 1$) is assigned to both $\boldsymbol{\omega}_R$ and $\boldsymbol{\omega}_V$. The second stage simulation has another parameter vector ($\boldsymbol{\omega}_{2\mu_1}$, $\boldsymbol{\omega}_{2\mu_2}$, $\boldsymbol{\omega}_{2\mu_3}$, $\boldsymbol{\omega}_{2\mu_4}$ or $\boldsymbol{\omega}_{2\mu_5}$, with corresponding means $\mu = 0.7$, 1, 1.3, 1.6 or 1.9) assigned to model either the risk $\boldsymbol{\omega}_R$ or the variation $\boldsymbol{\omega}_V$. Consequently, figure 3.15 shows the averaged performance where either the risk or variation estimates are wrong. Figure 3.15 confirms the earlier observation that the overall efficiency increase when the mean of the activity duration distribution increases. The project control procedures (EVM-STL and EVM-SNB) that require the calculation of sample quantiles prior to the execution of the project perform significantly worse than those that use the "virtual" buffers to derive tolerance limits (EVM-FPB and EVM-LPB). This effect shows most clearly when the mean of the activity distribution decreases.Moreover, we can again conclude that the EVM-FPB approach approximates best the observed efficiency of the EVM-LPB approach with significantly less control effort. This control effort for the EVM-FPB approach decreases with decreasing mean, and increases with increasing mean of the activity duration distribution.

Standard deviation   In order to test the potential effect of an under- or overestimation of the standard deviation of the distributions used to calculate the tolerance limits, the previous procedure was replicated with the parameter vectors $\boldsymbol{\omega}_{2s_1}, \boldsymbol{\omega}_{2s_2}, \boldsymbol{\omega}_{2s_4}, \boldsymbol{\omega}_{2s_5}$ and $\boldsymbol{\omega}_{2s_6}$, with a standard deviation of $\sigma = 0.15, 0.2, 0.28, 0.35, 0.42$ or 0.5. The tolerance limits where calculated using a distribution ($\boldsymbol{\omega}_2$) that has a standard deviation of 0.3, assigned to both the risk ($\boldsymbol{\omega}_R$) and the variation ($\boldsymbol{\omega}_V$) uncertainty modelling. Consequently, the simulations with $\boldsymbol{\omega}_{2s_1}, \boldsymbol{\omega}_{2s_2}$ or $\boldsymbol{\omega}_{2s_3}$ are used to model underestimates of the real standard deviation of the activity duration distribution, while the simulations with $\boldsymbol{\omega}_{2s_4}, \boldsymbol{\omega}_{2s_5}$ or $\boldsymbol{\omega}_{2s_6}$ are used to model overestimates. Figure 3.15 again shows how that the EVM-FPB method approximates the efficiency of the EVM-LPB approach for varying values of the real standard deviation of the activity duration distribution. The observations for the other project control approaches are described along the following lines. The EVM-1PB project control approach seems to be less efficient for overestimates of the standard deviation and more efficient for underestimates. For the control procedures that require the calculation of sample quantiles prior to the execution of the project (EVM-STL and EVM-SNB), an opposite effect is observed. These control procedure are more efficient for overestimates, while their efficiency decreases for the underestimates of the real standard deviation of the activity duration distribution. Figure 3.15 also shows that the control effort for the EVM-1PB, EVM-STL, EVM-FPB and EVM-SNB procedures decreases slightly when the real standard deviation of the activity duration distribution increases.

3.5   Discussion and conclusions

In order to minimise the effort spent by the project manager during the project control process, a top down earned value management/earned schedule (EVM/ES) method can be applied. However, the precise level of the WBS at which the EVM/ES control process should be conducted has been the subject of debate in literature. In this paper, we investigate five project control procedures using EVM/ES with different control points in the project. EVM-1PB calculates EVM/ES performance measures at the top WBS level and corresponds to the traditional use of EVM/ES. The EVM-STL control approach uses the same control point, but compares actual performance to statistical tolerance limits. The EVM-LPB control approach calculates the expected project duration from EVM/ES performance measures calculated for the dynamic longest path at each time instance during the execution of the project. In this research, we test two additional EVM/ES control approaches, inspired by the concept of buffers from the critical chain/buffer management (CC/BM) methodology. Buffers, which include baseline information in a structured manner will act as control

points at which EVM/ES performance measures are calculated. Control points are added at each non-critical path feeding into the critical path in the EVM-FPB control approach. Since this can lead to a high number of control points, a second approach is presented to reduce the number of control points by adding control points on subnetworks buffers in the EVM-SNB approach. For both the EVM-FPB and EVM-SNB control approaches, we presented a recursive search algorithm to determine the number of control points and to find their place in the baseline schedule. Subsequently, a procedure was proposed to formulate tolerance limits for both control approaches. These tolerance limits were used in dynamic project progress simulations to produce warning signals, which indicated whether the project was likely to meet a given deadline or not. All the project control approaches were tested using a large computational experiment that includes a broad range of simulated dynamic project progress situations. They were then compared based on their efficiency and the required control effort.

The research methods used in this paper are either in line with recent literature or provide a valuable extension. Three main strengths of the proposed method are discussed along the following lines. First, the computational experiment was performed on a well known, large dataset of projects, which allows us to produce generalisable results. Second, the dynamic project progress simulations, performed for this research, deployed an uncertainty modelling for the activity durations in two stages. Linear association was chosen to represent the risk factor, in addition to variation modelling through probability density functions. Third, an extensive comparison of both newly developed project control approaches (EVM-FPB and EVM-SNB) and approaches from the literature (EVM-1PB, EVM-STL, EVM-LPB) was presented. The weaknesses of the research methods with which this study was conducted are summarised along the following lines. In order to allow a practical implementation of the proposed control approaches, historical data should be available within an organization. It is also required that this data can be interpreted and calibrated to the uncertainty modelling that we propose in this paper. In addition, a minimum of theoretical knowledge on project scheduling theory should be present in the organization, in order to calculate a baseline schedule for the project that needs to be performed. Finally, the results of in this paper are believed to be generalisable, but are subjected to some restrictions (i.e. results are only shown for the SPI(t)) and some adaptations that were made to the methods available in literature (i.e. the concept of "virtual" buffers is applied to ensure that all the control approaches have a detection performance that is equal to 100%).

The contribution to the theory on project management and control is threefold. First, we presented an introduction of the theoretical basis to the CC/BM methodology to the practice of project control using EVM/ES sys-

tems. Second, we have characterised two newly proposed control approaches (EVM-FPB and EVM-SNB) and compared them against alternatives. Moreover, this study also presents the first characterisation of the EVM-LPB method in a large computational experiment. Third, we have shown the mathematical basis on how to apply EVM/ES at different control points in a project baseline schedule and different ways in which tolerance limits can be calculated to produce warning signals for project control.

The practical implications of our study should also be given some thoughts. The results of our study shows that all EVM/ES project control procedures require some calibration (either by using the concept of "virtual" buffers or by the calculation of sample quantiles) to some prior data of project executions. This translates in this study to a detection performance that is equal to 100%. This means that not a single project execution, that does not meet the predefined deadline, goes unnoticed by the project control approaches. We have shown that the EVM-LPB control approach is the most efficient. However, this approach requires a dynamic recalculation of the longest path at each review period during the execution of the project. It therefore places a heavy burden on the project manager in terms of control effort. In order to reduce this control effort, our study shows that the EVM-FPB method should be preferred. Even when only the critical path is monitored during the execution of the project, the efficiency of the EVM-FPB method already approximates that of the EVM-LPB method. Especially when is dealt with projects with a more serial network structure, it becomes more beneficial to incorporate only the critical activities on a single EVM/ES control chart.

In order to draw further conclusions on the practical implications from this study, we advise that future research expands on the introductory experiment that is presented in this paper. A model that incorporates corrective actions could refine the measure for the efficiency that is proposed in this paper. This model could then investigate whether the warning signals that are produced by the control approaches lead to a better decision making in the project control practice. Moreover, a case study could be implemented to test whether, from an EVM/ES accounting perspective, it would be attainable to switch from the control account-oriented approach to a structure that is dictated by the baseline schedule of the project. In addition, the preliminary findings on the concept of "virtual" buffers for the critical path and the feeding paths could be expanded. It could then be tested whether these can be translated to physical buffers in order to build contingencies into the baseline schedule.

References

[1] AbouRizk, S., Halpin, D., and Wilson, J. (1994). Fitting beta distributions based on sample data. *Journal of Construction Engineering and Management*, 120:288–305.

[2] Anbari, F. (2003). Earned value project management method and extensions. *Project Management Journal*, 34(4):12–23.

[3] Batselier, J. and Vanhoucke, M. (2014). Construction and evaluation framework for a real-life project database. *International Journal of Project Management*.

[4] Bowman, R. A. (2006). Developing activity duration specification limits for effective project control. *European Journal of Operational Research*, 174(2):1191–1204.

[5] Cheng, M.-Y. and Roy, A. F. (2010). Evolutionary fuzzy decision model for construction management using support vector machine. *Expert Systems with Applications*, 37(8):6061–6069.

[6] Cheng, M.-Y. and Wu, Y.-W. (2009). Evolutionary support vector machine inference system for construction management. *Automation in Construction*, 18(5):597–604.

[7] Chou, J.-S., Chen, H.-M., Hou, C.-C., and Lin, C.-W. (2010). Visualized evm system for assessing project performance. *Automation in construction*, 19(5):596–607.

[8] Colin, J. and Vanhoucke, M. (2014a). A multivariate approach to statistical project control using earned value management. *Submitted to an international journal (working paper version available at www.projectmanagement.ugent.be/sites/default/files/files/ wp/ColinVanhoucke2014d.pdf)*.

[9] Colin, J. and Vanhoucke, M. (2014b). Setting tolerance limits for statistical project control using earned value management. *Omega The International Journal of Management Science*, 49:107–122.

[10] Demeulemeester, E., Vanhoucke, M., and Herroelen, W. (2003). Rangen: A random network generator for activity-on-the-node networks. *Journal of Scheduling*, 6:17–38.

[11] Elmaghraby, S. (2000). On criticality and sensitivity in activity networks. *European Journal of Operational Research*, 127:220–238.

[12] Fleming, Q. and Koppelman, J. (2010). *Earned value project management. 3rd Edition*. Project Management Institute, Newton Square, Pennsylvania, 3rd edition edition.

[13] Goldratt, E. (1997). *Critical Chain*. North River Press, Great Barrington, MA.

[14] Golenko-Ginzburg, D. and Laslo, Z. (2001). Timing control points via simulation for production systems under random disturbances. *Mathematics and computers in simulation*, 54(6):451–458.

[15] Hartmann, S. and Briskorn, D. (2010). A survey of variants and extensions of the resource-constrained project scheduling problem. *European Journal of Operational Research*, 207:1–15.

[16] Herroelen, W. and Leus, R. (2001). On the merits and pitfalls of critical chain scheduling. *Journal of Operations Management*, 19:559–577.

[17] Hyndman, R. J. and Fan, Y. (1996). Sample quantiles in statistical packages. *The American Statistician*, 50(4):361–365.

[18] Jacob, D. and Kane, M. (2004). Forecasting schedule completion using earned value metrics? Revisited. *The Measurable News*, Summer:1, 11–17.

[19] Kelley, J. and Walker, M. (1959). *Critical path planning and scheduling: An introduction*. Mauchly Associates, Ambler, PA.

[20] Kuhl, M. E., Lada, E. K., Steiger, N. M., Wagner, M. A., and Wilson, J. R. (2007). Introduction to modeling and generating probabilistic input processes for simulation. In Henderson, S., Biller, B., Hsieh, M., Shortle, J., Tew, J., and Barton, R., editors, *Proceedings of the 2007 Winter Simulation Conference*, pages 63–76. New Jersey: Institute of Electrical and Electronics Engineers.

[21] Lee, S. H., Peña-Mora, F., and Park, M. (2006). Dynamic planning and control methodology for strategic and operational construction project management. *Automation in construction*, 15(1):84–97.

[22] Liang, T.-F. (2010). Applying fuzzy goal programming to project management decisions with multiple goals in uncertain environments. *Expert Systems with Applications*, 37(12):8499–8507.

[23] Lipke, W. (2012). Speculations on project duration forecasting. *The Measurable News*, 3:3–7.

[24] Lipke, W. and Vaughn, J. (2000). Statistical process control meets earned value. *CrossTalk: The Journal of Defense Software Engineering*, June:16–20,28–29.

[25] Lipke, W., Zwikael, O., Henderson, K., and Anbari, F. (2009). Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes. *International Journal of Project Management*, 27:400–407.

[26] Loch, C., De Meyer, A., and Pich, M. (2006). *Managing the unknown: A new approach to managing high uncertainty and risk in project*. John Wiley and Sons, Inc., New Jersey.

[27] McBride, W. J. and McClelland, C. W. (1967). Pert and the beta distribution. *Engineering Management, IEEE Transactions on*, (4):166–169.

[28] Moslemi Naeni, L. and Salehipour, A. (2011). Evaluating fuzzy earned value indices and estimates by applying alpha cuts. *Expert systems with Applications*, 38(7):8193–8198.

[29] Partovi, F. Y. and Burton, J. (1993). Timing of monitoring and control of cpm projects. *Engineering Management, IEEE Transactions on*, 40(1):68–75.

[30] R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[31] Rand, G. K. (2000). Critical chain: the theory of constraints applied to project management. *International Journal of Project Management*, 18(3):173–177.

[32] Raz, T. and Erel, E. (2000). Optimal timing of project control points. *European Journal of Operational Research*, 127(2):252–261.

[33] Steyn, H. (2000). An investigation into the fundamentals of critical chain project scheduling. *International Journal of Project Management*, 19:363–369.

[34] Steyn, H. (2002). Project management applications of the theory of constraints beyond critical chain scheduling. *International Journal of Project Management*, 20:75–80.

[35] Trietsch, D. (2005). The effect of systemic errors on optimal project buffers. *International Journal of Project Management*, 23(4):267–274.

[36] Trietsch, D. and Baker, K. R. (2012). Pert 21: Fitting PERT/CPM for use in the 21st century. *International Journal of Project Management*, 30(4):490–502.

[37] Trietsch, D., Mazmanyan, L., Govergyan, L., and Baker, K. R. (2012). Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation. *European Journal of Operational Research*, 216:386–396.

[38] Tukel, O. I., Rom, W. O., and Eksioglu, S. D. (2006). An investigation of buffer sizing techniques in critical chain scheduling. *European Journal of Operational Research*, 172(2):401–416.

[39] Van de Vonder, S., Demeulemeester, E., Herroelen, W., and Leus, R. (2005). The use of buffers in project management: The trade-off between stability and makespan. *International Journal of Production Economics*, 97:227–240.

[40] Van Slyke, R. (1963). Monte Carlo methods and the PERT problem. *Operations Research*, 11:839–860.

[41] Vanhoucke, M. (2010a). *Measuring Time - Improving Project Performance using Earned Value Management*, volume 136 of *International Series in Operations Research and Management Science*. Springer.

[42] Vanhoucke, M. (2010b). Using activity sensitivity and network topology information to monitor project time performance. *Omega The International Journal of Management Science*, 38:359–370.

[43] Vanhoucke, M. (2011). On the dynamic use of project performance and schedule risk information during project tracking. *Omega The International Journal of Management Science*, 39:416–426.

[44] Vanhoucke, M. (2012). Measuring the efficiency of project control using fictitious and empirical project data. *International Journal of Project Management*, 30:252–263.

[45] Vanhoucke, M. (2014). *Integrated Project Management and Control: First come the theory, then the practice*. Management for Professionals. Springer.

[46] Vanhoucke, M., Coelho, J., Debels, D., Maenhout, B., and Tavares, L. (2008). An evaluation of the adequacy of project network generators with systematically sampled networks. *European Journal of Operational Research*, 187:511–524.

[47] Wauters, M. and Vanhoucke, M. (2014). Support vector machine regression for project control forecasting. *Automation in Construction*, 47:92–106.

[48] Williams, T. (1995). What are PERT estimates? *Journal of the Operational Research Society*, 46:1498–1504.

4

An empirical perspective on activity durations for project management simulation studies
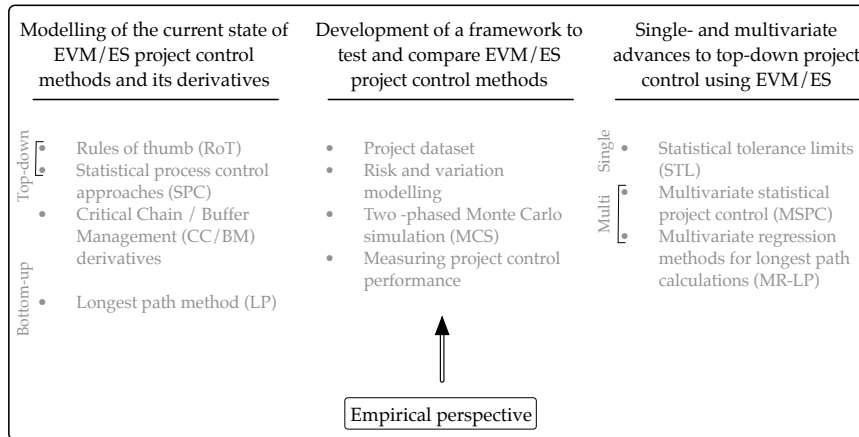
Figure 4.1: Schematic overview of the elements, of the larger body of research in this book, that are discussed in chapter 4.

Abstract

Simulation has played an important role in the project management studies of the last decades, but in order for them to produce practical results, a realistic distribution model for activity durations is indispensable. The construction industry often has the needed historical records of project executions, to serve as inputs to the distribution models, but a clearly outlined calibration procedure is not always readily available, nor are their results readily interpretable. In this study, we illustrate how data from the construction industry can be used to derive realistic input distributions. Therefore, we apply the model of [21] to the large empirical dataset of [1] and describe the results. From a discussion of these results, we present an empirical classification of project executions. We show three possible uses for the calibration procedure and the classification in project management simulation studies. These were validated using a case study in a construction company.

4.1   Introduction

The value of Monte Carlo simulations (MCS) for project management has long been established [10, 17, 19]. Owing to the advances in information technology of the last few decades, simulation studies now stand at the core of newly developed techniques in project management [6, 20, 25], and

serve to test them and quantify their performance in large computational experiments [4, 22].

In these simulations, the stochastic activity durations for the activity networks are drawn from a predefined distributional model. The parameters for the distributions that are proposed in literature however, are not always as clear and intuitive as one might hope. Moreover, they often lack a clearly outlined calibration method with which, in practice, input distributions can be generated from empirical records of project executions. A recent exception is the Parkinson distribution with a lognormal core [21], for which a calibration procedure was presented, alongside its theoretical description. In this paper, we expand on the work of [21] by applying it to the database of [1]. This database currently consists of 51 real-life projects, predominantly from the construction industry, which are freely available for researchers* in the ProTrack file format [22]. However, this paper aims not to draw general conclusions, but rather provides an illustration of how a calibration to a large empirical database can be performed and how its results can be interpreted. In order to generate practically applicable activity duration inputs for project management simulation studies, we believe that researchers and practitioners should always consider the use of their own historical data of project executions. Since, even when the dataset of [1] is expanded in the coming years, some project environments will still be too specific to be correctly referenced against the real-life project executions in this dataset.

The contribution of this paper is threefold. First and foremost, we present the first application of a theoretical distribution to a large database of empirical records in project management. From the thirteen projects that confirmed the theoretical distribution and which were retained for further analysis, eleven were performed in the construction industry and two were performed in IT and education industry respectively. Our study is therefore not restricted to the construction industry, but we pay special attention to the interpretation of the results from construction projects. In order to promote the reproducibility of this study and to encourage its use on practitioner databases, we have implemented the calibration procedure by [21] in the open source statistical programming language R[16]. Second, we present an empirical classification for records of project executions, which can assist in the interpretation of historical data and serve as input for project management simulation studies. Third, we show how the calibration procedure and the empirical classification can be usefully applied in the project management practice by validating them on a case study in a Belgian construction company. The outline of this paper is as follows. In the literature overview section, we will introduce the distributional model and calibration procedure by [21]. In the methodology section, we will dis-

---

*http://www.or-as.be/research/database

cuss the data cleaning that was performed on the online dataset of [1] and apply the calibration procedure. Using the outputs of this procedure, we will propose the empirical classification of project executions in the results section. In the discussion and conclusions section, we will validate the use of the calibration procedure and the classification method, using a case study on project management in a Belgian construction company. In addition, we will then summarize the work that was done for this paper and its value for future project management research. In the appendix A.1 of this paper,we present a working R template to reproduce the results from this paper.

4.2   Literature overview

The distributional model proposed by [21] is introduced in this section. First, we will discuss what level of detail is needed in the empirical data and how these data are interpreted. Second, we will illustrate how the calibration procedure by [21] was applied in our study.

4.2.1   Focus on the relative duration of activities

The desirable level of detail in empirical records for this study lies at the activity level of the work breakdown structure (WBS) of a project. Concretely, we aspire to have planning and progress data as complete as possible for all activities in a project. The justification for this is that we target the application of the calibration procedure to project management studies that depart from the schedule performance of a project. In these studies, the baseline schedule is considered as a point of reference throughout the execution of the project and the real schedule performance and the cost associated with it, are compared relative to the baseline estimates. In the MCS for such project management studies, the individual activities are given a duration that might deviate from the estimated duration that was used for the construction of the baseline schedule [4, 6, 23, 24]. The relation to real-life projects is evident, since only very seldom a project will go exactly as planned in terms of the durations of the individual activities and their associated costs. It is not within the scope of this paper to test and to compare all the different theoretical distributions for activity durations that have been proposed for use in project management studies in the literature. Among these are the generalized beta distribution [9], the lognormal distribution [15] and a mixture of a beta and a uniform distribution [8]. These distributions are valid in their own right for the generation of activity durations in project management simulations, since they all adequately model the real-world excess variability and over-occurrence of tail

area events [14]. In order not to risk presenting an oversimplified summary of these distributional models, we refer to the original papers for theoretical background and practical implementations. We will however describe the model proposed by [21] in more detail, since we have chosen to implement it in our study because of the structured manner in which the Parkinson effect and the influence of rounding errors are incorporated in the accompanying calibration method.

Consider a set of projects $\mathcal{P}$ in an empirical database. For a project $j \in \mathcal{P}$, that has a number of activities N_j, d_{ij} represents the real duration of activity $i = 1 \ldots N_j$ in project j. Correspondingly, for that activity, $\hat{d}_{ij}$ denotes the baseline estimate duration, that was used in the construction of the baseline schedule. We therefore require the empirical records to have data available such that the relative durations $d_{ij}/\hat{d}_{ij}$ for all activities in all projects can be calculated. If the proportion $d_{ij}/\hat{d}_{ij} > 1$, then activity i of project j has experienced a delay (tardy). The real duration of the activity was larger than its anticipated duration when the baseline schedule was constructed. If $d_{ij}/\hat{d}_{ij} < 1$, the activity was finished in less time than planned (early). [21] assume the proportion $d_{ij}/\hat{d}_{ij}$ to be lognormally distributed, even though this might not always be directly observable. In other words, the natural logarithm of $d_{ij}/\hat{d}_{ij}$ is assumed to be normally distributed. The lognormal distribution is reasoned to effectively incorporate both the additive and multiplicative effects that are omnipresent in a real-life project execution environment. This distributional model was validated by [21] for activity times on several independent datasets from project scheduling applications by incorporating two important practical considerations. These considerations explain why lognormality is not always apparent at first sight in real-life data. The first consideration is the *Parkinson effect* that is translated into the reported data as "hidden earliness". In projects, workers are generally not incentivised to report early activities. Rather, they will effectively use up all time allotted to a given task by starting late, by multi-tasking, by shifting resources or by simply reporting the task as on time [7]. Where in reality $d_{ij}/\hat{d}_{ij} < 1$ might have been possible, it is more likely that $d_{ij}/\hat{d}_{ij} = 1$ will be reported. The second consideration handles *rounding effects*. In practice, the reporting of activity performances is mostly done on a coarse time scale and reported times will be influenced by rounding errors. This problem can be recognised easily when a project with large differences in baseline estimate durations is considered. Typically, a single unit for reporting the time performance of activities will be chosen. For example, if a project has a large number of activities that are planned to take a couple of weeks and a small number of activities that take around 3-5 days, the unit for reporting activity performance is likely to be days. The rounding that occurs due to the reporting in days will surely have a larger influence on the analysis from the

**Histogram of the
relative empirical distribution**
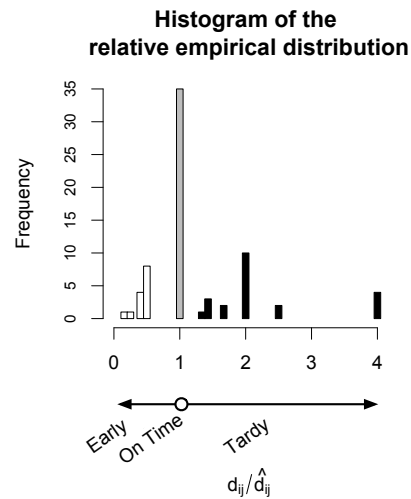
Figure 4.2: The relative empirical distribution for the illustrative record of an
executed project.

proportion $d_{ij}/\hat{d}_{ij}$ for the small activities than for the larger activities. Concretely, when the Parkinson effect and the rounding effects are considered, [21] conjectured that the relative empirical distributions $d_{ij}/\hat{d}_{ij}$ follow a Parkinson distribution with a lognormal core.

In addition to the validation of the Parkinson distribution with a lognormal core, [21] validated the assumption of linear association in project management. According to the authors, the lognormal distribution lends itself to be used when statistical dependence is modelled using linear association. We will not consider this further, as this is only relevant for when an actual simulation model is proposed. A recent application of linear association in project management simulation can be found in [4]. Here, we are primarily concerned with the calibration of the inputs to such simulation studies.

4.2.2   Calibration procedure proposed by Trietsch et al. (2012)[21]

In order to guide the reader through the calibration procedure proposed by [21], we introduce an illustrative record of an executed project. This is a record of a construction project for a commercial building, taken from the dataset of [1] (C2013-09: "Urban Development Project), that has a total value of around 1.5 € million and a planned duration of 291 working days. The scope of the project was development of an integrated building

complex, and the project was actually completed within 360 working days and experienced a budget overrun of around 13%. The project schedule consisted of 71 activities for which both baseline estimate durations and actual durations were reported. The relative empirical distribution $d_{ij}/\hat{d}_{ij}$ derived from the $i = 1 \ldots 71$ activities is depicted in the histogram in figure 4.2. In this figure, we depicted the early activities in white ($d_{ij}/\hat{d}_{ij} < 1$), the on-time activities in grey ($d_{ij}/\hat{d}_{ij} = 1$) and the tardy activities in black ($d_{ij}/\hat{d}_{ij} > 1$).

We will proceed with an application of the Parkinson distribution with a lognormal core to the presented project. We have structured the calibration procedure, proposed by [21], in the following way. A maximum of four hypothesis tests (P0, P1, P1.1, P1.2) can be conducted in this order to confirm the Parkinson distribution with a lognormal core. We will respectively discuss these tests in steps 1 to 4 of this section. There, we will also explicitly state the null hypothesis for these tests.

This procedure follows a sequential testing mechanism, starting with step 1. If in a certain step, the hypothesis test is not able to confirm the Parkinson distribution with a lognormal core, the procedure proceeds with the next step. Consequently, the test in step 4 will only be conducted if all prior tests fail to confirm the Parkinson distribution with a lognormal core.

Step 1 (P0)

A visual inspection of figure 4.2 reveals that the assumption of lognormality for the relative empirical distribution $d_{ij}/\hat{d}_{ij}$ seems reasonable. We now state the null hypothesis $H_{0,\text{P0}}$ for the test P0 and proceed with explaining how this hypothesis is tested according to the procedure proposed by [21].

$$H_{0,\text{P0}} \quad : \quad \ln(d_{ij}/\hat{d}_{ij}) \text{ follows a normal distribution}$$

In order to accept or reject $H_{0,\text{P0}}$, [21] suggested to compare the Pearson's linear correlation coefficient (R) to the tabulated values by [13]. The linear correlation coefficient can be used for testing the assumption of normality, by the use of normal probability plots. [13] quantified this use and from table 2 of their paper, a p-value can be obtained for this normality assumption test. The linear correlation coefficient R should be calculated from a linear regression of the empirical values of $\ln(d_{ij}/\hat{d}_{ij})$ on the Blom scores [2] for these values [21]. Blom proposed the use of $\Phi((r - 3/8)/(n + 1/4))$ to estimate the normal scores for a sample, where r represents the rank of an observation in a sample, n represents the sample size and $\Phi(p)$ is used to denote the pth quantile of the standard normal distribution.

Figure 4.3a shows the Blom scores for $\ln(d_{ij}/\hat{d}_{ij})$ plotted against the empirical values. The first thing that can be noticed from figure 4.3a is the large number of activities that have $\ln(d_{ij}/\hat{d}_{ij}) = 0$. This means that in this project a lot of activities were reported as exactly on time, since $\ln(d_{ij}/\hat{d}_{ij}) = 0 \implies d_{ij}/\hat{d}_{ij} = 1$. The linear regression line that is shown in figure 4.3a tries to explain the values from $\ln(d_{ij}/\hat{d}_{ij})$ using a linear fit on their Blom scores. According to [21], the intercept and the linear coefficient of this regression model respectively estimate the mean ($\hat{m}$) and the standard deviation ($\hat{s}$) of the natural logarithm of the relative empirical distribution $d_{ij}/\hat{d}_{ij}$. The R value, found in this regression, results in a p-value of 0.004, when it is compared to the tabulated values of [13]. Given a level of significance $\alpha = 5\%$, we have to conclude that the null hypothesis must be rejected (p-value $< \alpha$). The variable $\ln(d_{ij}/\hat{d}_{ij})$ can not be assumed to be normally distributed at this point in the analysis. Therefore, we proceed with step 2 of the procedure.

Step 2 (P1)

Given the large proportion p_p of activities that is reported as exactly on-time ($p_p = 35/71 = 49\%$, depicted in grey in figure 4.2), it is very likely that the Parkinson effect is present in these data. In order to prevent this effect from obscuring our interpretation of the data, the first step is to remove all activities that were reported as completed on time. [21] described this step as accounting for the (pure) Parkinson distribution. Figure 4.3b shows the plot of the empirical data and the Blom scores, with these on-time activities removed. Again, R, $\hat{m}$ and $\hat{s}$ are also shown. A new hypothesis ($H_{0,P1}$) can now be tested using R and the tabulated values in [13].

$H_{0,P1}$: $\ln(d_{ij}/\hat{d}_{ij})$ follows a normal distribution with a Parkinson effect after the removal of the on-time points

The resulting p-value of 0.006 again leads to the rejection of this new null hypothesis, using the significance value α of 5%.

In addition to trimming the sample of on-time points, [21] propose two additional treatments to test whether $d_{ij}/\hat{d}_{ij}$ indeed follows the Parkinson distribution with a lognormal core. For an extensive coverage and justification for these treatments, we refer to the original article. Here, we will only show how they affect our discussion of the illustrative empirical record. The first treatment is contained in the test that is described in step 3. If this test is still unable to accept the Parkinson distribution with a lognormal core for the project data, we proceed to the test in step 4, which implements the second treatment.

Step 3 (P1.1)

The first treatment that is implemented is the removal of a proportion p_p (the proportion of on-time points, depicted in grey in figure 4.2) of the strictly tardy points (black in figure 4.2). The justification is that if the Parkinson distribution applies, the resulting data would constitute a trimmed, but complete sample in which early and tardy points are approximately in the right proportion [21].

$H_{0,\text{P1.1}}$: $\ln(d_{ij}/\hat{d}_{ij})$ follows a normal distribution with a Parkinson effect
after the removal of on-time points
and after removing p_p strictly tardy points

We follow the example of [21] and work with two trimmed samples. From the 22 tardy activities (in black), we remove 11 points (49% (p_p) of 22). This results in two samples, each having 11 different strictly tardy points. Figure 4.3c shows the plots for these two samples where the empirical values are depicted against their Blom scores. This first treatment results in two p-values ((a) 0.004, (b) 0.004)) which both lead to the rejection of the null hypothesis $H_{0,\text{P1.1}}$ at the 5% significance level. Ultimately, we will therefore proceed with step 4.

Step 4 (P1.2)

The second treatment involves the removal of ties. In figure 4.3b, we notice strong grouping (tied Blom scores), even when a proportion of the strictly tardy points and the on-time points have been removed. These ties are due to the rounding that is caused by the coarse time scale in which activity times are reported. A possible remedy for this is the calculation of the average Blom score for all tied points in a group. Figure 4.3d shows the resulting plots of the empirical data and the averaged Blom scores with the corresponding regression model for the two samples with 11 tardy points removed and the ties resolved. We now have a better fit for the regression models, resulting in higher R values and consequently, higher p-values ((a) 0.959, (b) 0.986). These p-values mean that the new null hypothesis $H_{0,\text{P1.2}}$ can no longer be rejected at the 5% significance level. Consequently, we can now assume the variable $d_{ij}/\hat{d}_{ij}$ to be following a Parkinson distribution with a lognormal core, according to the theory of [21].

$H_{0,\text{P1.2}}$: $\ln(d_{ij}/\widehat{d_{ij}})$ follows a normal distribution with a Parkinson effect
after the removal of on-time points,
after removing p_p strictly tardy points
and resolving ties
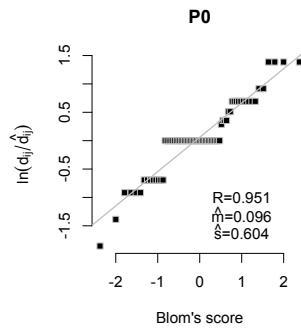
(a) Hypothesis test for P0 on the
relative empirical distribution
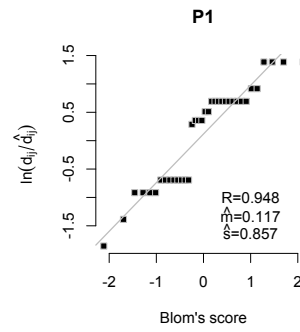
(b) Hypothesis test for P1 on the
relative empirical distribution
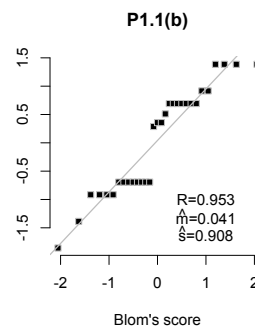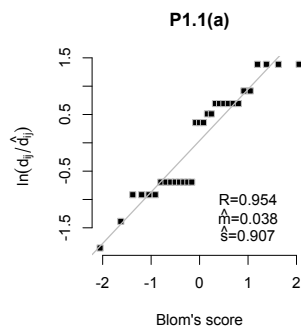
(c) Hypothesis test for P1.1 on the relative empirical distribution
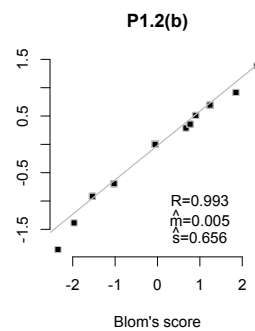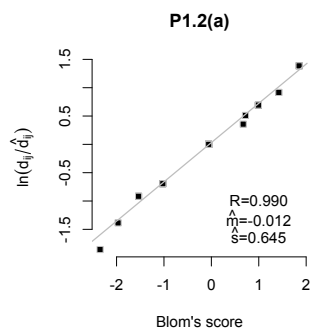
(d) Hypothesis test for P1.2 on the relative empirical distribution

Figure 4.3: Application of the four-step calibration on C2013-09 "Urban
Development Project"

This section gave an illustrative overview of the mechanism to test the Parkinson distribution with a lognormal core on real activity durations using the four-step procedure of [21]. For more detailed information on the mechanism and justification of the methodology, the reader is referred to the original paper. In the next section, the procedure of [21] is used on the project database of [1].

4.3 Methodology

In this section, we will discuss the analysis of the online database presented by Batselier and Vanhoucke (2014). First, we will explore which records from this database are suited for this study. Second, we will implement the Parkinson distribution with a lognormal core procedure on these data. Moreover, we will present the p-values and the estimated means and standard deviations for the natural logarithm of the relative empirical distributions, that result from this procedure.

4.3.1 Discussion of the empirical records in the database of Batselier and Vanhoucke (2014)

The empirical records discussed in this paper are selected from the online database of Batselier and Vanhoucke (2014), which currently consists of 51 real-life projects with actual progress data. 39 of these 51 projects are construction projects. This database is one of the largest and most diverse yet with respect to planned duration and budget at completion, in the project management literature and is updated and expanded continuously. Furthermore, the quality and authenticity of the data are guaranteed through the application of a database construction tool called project cards [1].

From this empirical database, we selected those projects that contain complete project progress information at the activity level, such that for all projects j in this set $\mathcal{P}$ the relative empirical distribution $d_{ij}/\hat{d}_{ij}, \forall i \in 1 \dots N_j$ could be calculated. Thus, from the total of 51 projects we retained 24 projects for further study.

Before proceeding with the application of the four-step procedure of [21], some more data cleaning had to be performed. First, the progress data at the work package level of the work breakdown structure (WBS) of the projects had to be removed. The work package level often provides adequate detail to monitor the progress of a project during its execution, but is of no use here. We are merely interested in the progress data at the activity level. Second, we had to manually remove some points from the calculated

values in $d_{ij}/\hat{d_{ij}}$, because they had no real connection to activity progress. Some very large and some very small values for $d_{ij}/\hat{d_{ij}}$ occurred because of errors that were made in the scheduling phase prior to the project executions. If an activity in the project was planned, but not executed, the ratio between the actual activity duration and its baseline estimate was very small and the data point needed to be removed. Alternatively, if during the project execution an additional activity was defined, which was not previously incorporated in the baseline schedule, the ratio for that activity could be very large. An example of such a situation is when additional work is created, when the scope of the project is redefined during the execution, or when cleaning-up activities are performed in the end of the project that were not accounted for in the scheduling phase. If the project manager did not bother to accurately adapt the baseline schedule to these situations, erroneous points in $d_{ij}/\hat{d_{ij}}$ occur, which consequently need to be removed from our analysis. Altogether, the manual removal of data points was quite futile. Only 7 out of a total of 1,881 activities for the 24 projects, had to be removed.

4.3.2 Application of the Parkinson distribution with a lognormal core

We applied the calibration procedure proposed by [21] to the 24 projects of the online dataset. For all 24 projects, we tested whether the relative empirical distribution $d_{ij}/\hat{d_{ij}}$ could be shown to be Parkinson distributed with a lognormal core. To this end, we applied the four-step procedure (P0, P1, P1.1 and P1.2) to the empirical records.

In table 4.1, we present the results for the hypothesis tests that allowed us to confirm the Parkinson distribution with a lognormal core for the projects in the empirical database. In between brackets, the p-values are shown. In addition, we present the estimated mean ($\hat{m}$) and standard deviation ($\hat{s}$) of the natural logarithm of the relative empirical distribution $d_{ij}/\hat{d_{ij}}$ for each project. Table 4.1 only shows 13 of the 24 projects. The remaining 11 projects yielded p-values lower than $\alpha = 5\%$ for all hypothesis tests ($H_{0,P0}$, $H_{0,P1}$, $H_{0,P1.1}$ and $H_{0,P1.2}$). For these projects, we were compelled to reject each of the null hypotheses at the 5% significance level and the assumption that the data are distributed according to the Parkinson distribution with a lognormal core could no be withheld. For the 13 projects that are shown in table 4.1 only the result of the hypothesis test is shown that allowed us to confirm the Parkinson distribution with a lognormal core for that project (p-value $> \alpha$). The reader should note that unlike in the previous section, we only show a single p value for tests $H_{0,P1.1}$ and $H_{0,P1.2}$ in table 4.1, although still the example of [21] was followed and two trimmed samples

Number	Project Name	Parkinson distribution with lognormal core $\hat{m}$ $\hat{s}$ (p-value)			
		P0	P1	P1.1	P1.2
C2011-07	Patient Transport System		0.47 0.26 (0.73)		
C2012-13	Pumping Station		0.24 0.62 (0.09)		
C2013-01	Wiedautokaai Fenders		0.45 0.70 (0.33)		
C2013-03	Brussels Finance Tower		-0.09 0.62 (0.45)		
C2013-07	Family Residence		-0.03 0.51 (0.59)		
C2013-08	Timber House		0.27 0.64 (0.15)		
C2013-09	Urban Development Project				0.05 0.73 (0.59)
C2013-11	Recreation Complex		0.20 0.72 (0.05)		
C2013-12	Young Cattle Barn	0.01 0.57 (0.18)			
C2013-13	Office Finishing Works (1)		-0.89 0.54 (0.59)		
C2013-15	Office Finishing Works (3)		-0.31 0.99 (0.57)		
C2013-17	Office Finishing Works (5)		-0.68 1.03 (0.42)		
C2014-03	Organizational Development				1.05 3.06 (0.46)

Table 4.1: Discussion of the activity times data of Batselier en Vanhoucke (2014) [1] using the Parkinson distribution with a lognormal core [21]

were used. However, if both p-values were larger than 5%, only the results for the sample with the highest p-value are shown. If one of the two p-values was smaller than 5%, the corresponding hypothesis was rejected. From the 13 projects that were retained for further analysis, only two were not performed in the construction industry. The "Patient Transport System" and "Organizational Development" projects were respectively performed in the IT and Education sector.

The first observation that can be made from table 4.1 is that only 1 of the 24 projects in the online dataset is directly lognormally distributed. The construction project described as "Young cattle barn" is the only one that has a p-value larger than $\alpha = 5\%$ for P0. For the remaining 12 projects, presented in table 4.1, the Parkinson distribution with a lognormal core is only confirmed for two projects by P1.2. For the other 10 projects, the distributional model was confirmed using P1. We have not recorded any occurrence of an empirical record for which P1.1 succeeded in confirming the Parkinson distribution with a lognormal core. As it appears from the online dataset, the first treatment (removing tardy points) is only effective to confirm the distributional model when it is combined with the second treatment (resolving ties).

In the next section we will discuss the $\hat{m}$ and $\hat{s}$ values of table 4.1 and present a classification of the data for further research.

4.4   Results

In this section, we will discuss the estimated values for the standard deviation $\hat{s}$ and the mean $\hat{m}$ of the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d_{ij}}$, as shown in table 4.1. First we will discuss the standard deviation $\hat{s}$. Second, the results for the mean $\hat{m}$ will be examined more closely. These standard deviations and means will be used in an empirical classification for project execution that will be presented at the end of this section.

4.4.1   The standard deviation of the natural logarithm of the relative empirical distributions

The standard deviations $\hat{s}$ of the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d_{ij}}$ are discussed in this section. We will start from a visual inspection and proceed with a more thorough statistical interpretation.

Visual inspection of $\hat{s}$

Figure 4.4c shows the estimated standard deviations for the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d}_{ij}$, that were previously presented in table 4.1. In order to incorporate the uncertainty that is associated with estimating the standard deviation, we added 95% confidence intervals to the plot. The confidence interval around $\hat{s}$ is calculated using the assumption that $(n-1)\hat{s}^2/s^2$ follows a χ^2 distribution with $n-1$ degrees of freedom, if the original population of data is normally distributed. s denotes the population standard deviation of the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d}_{ij}$, whereas $\hat{s}$ was the estimate produced from a sample, and n denotes this sample size. The confidence limits around s can then be calculated from the inequality $\chi^2_{1-\beta/2} \leq (n-1)\hat{s}^2/s^2 \leq \chi^2_{\beta/2}$, with a confidence level $1-\beta$. The confidence interval is denoted in grey, while the $\hat{s}$ value is plotted as a white circle. From a first glance of figure 4.4c, we notice the large estimated standard deviation of the natural logarithm of the relative empirical distribution of the "Organizational Development" project (C2014-03). This large standard deviation clearly is unique among the projects examined from the online dataset. For the other projects, a clear classification is not immediately apparent from a visual inspection.

Statistical interpretation of $\hat{s}$

We proceed with a more robust, statistical interpretation of the values for $\hat{s}$ shown in figure 4.4c. Therefore, we implemented Levene's test for the homogeneity of variances [12]. Levene's test can be used to verify the assumption that the variances (or standard deviations) are equal across different samples or groups. To that end, a statistic W is calculated that under the null hypothesis ($\hat{s}$ is equal for all projects in a group) is distributed according to an F distribution. For details on the calculation of W, the reader is referred to the original work.

The statistic W is used to test the assumption of homogeneity of variances for different groups of projects from table 4.1 and figure 4.4c. Clearly, the "Organizational Development" project (C2014-03) is unique as no other project can be found in the dataset for which the assumption of equal variance is not rejected by Levene's test at a 5% significance level. We will therefore create a group (HIGH $\hat{s}$) for this project alone. The question then remains whether the other projects can be assumed to have equal variances. Levene's test confirms this assumption, based on a found p-value of 0.39, at a 5% significance level ($W = 1.06$, $P[W > 1.06] = 0.39$ with $(d.f = 11; 223)$).

We have thereby classified the 13 projects into two groups. A group of projects with a LOW $\hat{s}$ and a group of one project with a HIGH $\hat{s}$. The LOW $\hat{s}$ group includes projects C2011-07, C2012-13, C2013-01, C2013-03, C2013-07, C2013-08, C2013-09, C2013-11, C2013-12, C2013-13, C2013-15 and C2013-17, while the HIGH $\hat{s}$ group includes only C2014-03.

Figure 4.4d reproduces figure 4.4c, with a color coding for the two groups. The HIGH $\hat{s}$ group is colored in black and the LOW $\hat{s}$ group is depicted in a light shade of grey. In addition, we added the pooled standard deviation $\hat{s}_{\mathrm{pooled}}$ for each group to represent the real standard deviation of that group. This pooled standard deviation is calculated for each group using:

$$\hat{s}_{\mathrm{pooled}} = \sqrt{\frac{\sum_{j=1}^{k}(n_j - 1)\hat{s}_j^2}{\sum_{j=1}^{k}(n_j - 1)\hat{s}_j}}$$

Where $\hat{s}_j$ is the estimated standard deviation of the natural logarithm of the relative empirical distribution for a project j in a group, and n_j is the sample size for that project. k is used to denote the number of projects in that group.

4.4.2   The mean of the natural logarithm of the relative empirical distributions

The mean $\hat{m}$ of the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d}_{ij}$ are closely examined in this section. We start again from a visual inspection and then add a statistical analysis.

Visual inspection of $\hat{m}$
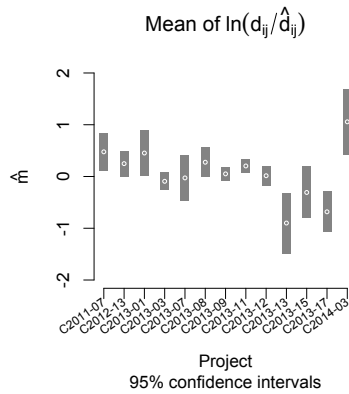
Mean of $\ln(d_{ij}/\hat{d}_{ij})$

$\hat{m}$

Project
95% confidence intervals

(a) $\hat{m}$ for the empirical project records
according to the Parkinson distribution
with a lognormal core.

Mean of $\ln(d_{ij}/\hat{d}_{ij})$

$\hat{m}$

High
Mid
Low

Project
95% confidence intervals

(b) $\hat{m}$ for the empirical project records
according to the Parkinson distribution
with a lognormal core, with the grouping
illustrated in color.

Standard deviation of $\ln(d_{ij}/\hat{d}_{ij})$

$\hat{s}$

Project
95% confidence intervals

(c) $\hat{s}$ for the empirical project records
according to the Parkinson distribution
with a lognormal core.

Standard deviation of $\ln(d_{ij}/\hat{d}_{ij})$

$\hat{s}$

High

Low
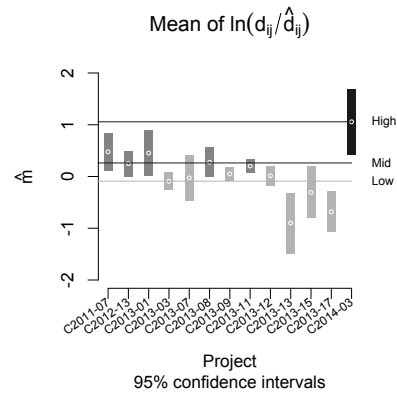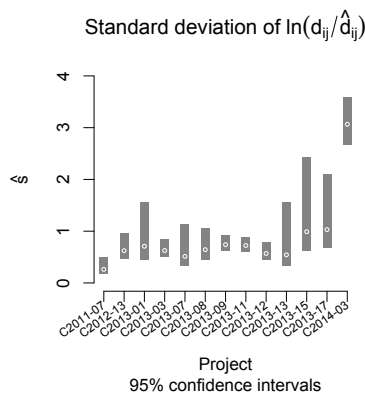
Project
95% confidence intervals
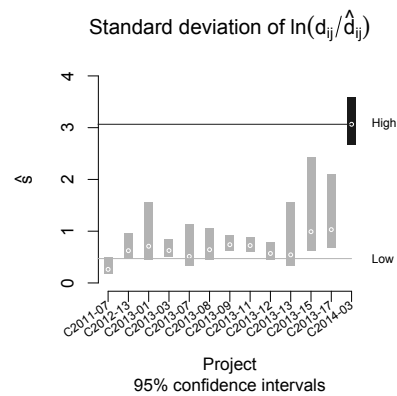
(d) $\hat{s}$ for the empirical project records
according to the Parkinson distribution
with a lognormal core, with the grouping
illustrated in color.

Figure 4.4: Discussion of the results obtained from the application of the four-step
calibration procedure on the dataset of empirical records of project executions

Figure 4.4a presents the estimated means $\hat{m}$ of the natural logarithm of the
relative empirical distributions $d_{ij}/\hat{d}_{ij}$, that were previously presented in
table 4.1. The data points are accompanied by 95% confidence limits that

were calculated from the inequality $t_{\beta/2,n-1} \leq (\hat{m} - m)/s \leq t_{1-\beta/2,n-1}$. Here, m denotes the population mean of the natural logarithm of the relative empirical distributions $d_{ij}/\hat{d_{ij}}$, whereas $\hat{m}$ was the estimate produced from a sample, and n denotes this sample's size. s is given by the $\hat{s}_{\text{pooled}}$ value of the group to which the project belongs, $t_{p,n-1}$ denotes the pth quantile of the t distribution with $n - 1$ degrees of freedom and $1 - \beta$ was chosen as 95%. From a visual inspection of figure 4.4a, we immediately notice the large mean for the "Organizational Development" project. This project was already identified as unique in terms of its standard deviation, and its mean does also reflect the unique character of this project. Moreover, we observe a large group of projects that have a logarithmic mean of around 0. Three distinct projects (C2013-13, C2013-15 and C2013-17) appear to have a somewhat smaller observed logarithmic mean. We aim to verify this observation using a statistical analysis in the next section.

Statistical interpretation of $\hat{m}$

Welsch's test [26] is well-known for its ability to test for a difference in the means across groups or samples, when the variance is not necessarily assumed to be equal. Under the assumption of equal mean (which is the standard null hypothesis for this test) Welsch's F statistic is distributed according to an F distribution. The conclusions for the analysis of the presented $\hat{m}$ values can be summarised along the following lines. First, Welsch's test confirms that not all means for all 13 projects of table 4.1 can be assumed to be equal as the null hypothesis is easily rejected at the 5% significance level ($F = 4.16$, $P[F > 4.16] = 1.9 * 10^{-4}$ with $(d.f = 12; 47.14)$). Moreover, the observations from the prior visual inspection could not be confirmed using Welsch's test. It is unlikely that the grouping mentioned earlier for $\hat{m}$ can be withheld. Although the three projects with the smaller $\hat{m}$ can be assumed to have an equal mean ($F = 0.74$, $P[F > 0.74] = 0.49$ with $(d.f = 2; 10.30)$), the hypothesis of equal means for the remaining projects is rejected ($F = 2.72$, $P[F > 2.72] = 0.015$ with $(d.f = 8; 44.62)$).

We therefore continue our discussion of $\hat{m}$ in a more systematic manner. We add the "Organizational Development" project to a HIGH $\hat{m}$ group and proceeded with organizing the remaining 12 projects into a MID $\hat{m}$ group and LOW $\hat{m}$ group using an enumeration algorithm. This algorithm enumerates all possible combinations of the remaining 12 projects and tests the assumption of equal mean for these combinations. Only six possible combinations were found where the assumption of equal mean is confirmed within the groups and rejected between the groups. We then chose proceed with that combination that had the highest combined p-value for the Welsch's tests for the two groups. The MID $\hat{m}$ group then consists of projects C2011-07, C2012-13, C2013-01, C2013-08 and C2013-11, while the

LOW $\hat{m}$ group consists of C2013-03, C2013-07, C2013-09, C2013-12, C2013-13, C2013-15 and C2013-17.

Figure 4.4b shows the grouping and the corresponding pooled mean estimates $\hat{m}_{\text{pooled}}$. These where calculated for each group as:

$$\hat{m}_{\text{pooled}} = \frac{\sum_{j=1}^{k} n_j \hat{m}_j}{\sum_{j=1}^{k} n_j}$$

Where $\hat{m}_j$ is the estimated mean of the natural logarithm of the relative empirical distribution for a project j in a group, and n_j is the sample size for that project. k is again used to denote the number of projects in that group. These $\hat{m}_{\text{pooled}}$ estimate the real logarithmic means of the groups of project executions, even though this mean is not observed directly from the data. In figure 4.4b, the HIGH $\hat{m}$ group is colored black, the MID $\hat{m}$ group has a grey color and the LOW $\hat{m}$ group was given a lighter shade of grey. Table 4.2 shows the results for the Welsch's test for the constructed groups.

	Welsch's $\hat{m}$ test	
	F	p.value $(d.f.1; d.f.2)$
LOW	2.42	0.06 (6; 21.62)
MID	1.17	0.34 (4; 24.9)
HIGH	* -	- (0; -)

*: No value could be found since the group consists out of a single project.

Table 4.2: Summary of Welsch's tests for $\hat{m}$

4.4.3 Empirical classification for project executions

It has been established [29] that project management simulation studies need to incorporate empirical data in order to produce practical results. However, it is not a straightforward task to incorporate empirical data into simulation models of project executions. In order to facilitate this task, we propose a classification for project executions that results from the grouping of the $\hat{m}$ and $\hat{s}$ values in the previous section. Next, we will propose two ways to classify new (real or simulated) project executions. Finally, we

will discuss the proposed classification from a construction industry point of view.

Given the grouping of the empirical records of [1] for both the $\hat{m}$ and $\hat{s}$ values, we propose the classification of table 4.3. The six classes of project executions in table 4.3 all combine a value from the $\hat{m}_{\mathrm{pooled}}$ estimates with a value from the $\hat{s}_{\mathrm{pooled}}$ estimates. A project execution, whether it is real or simulated, can then be referenced against these six classes. This can be done in two ways, as we will describe in the following paragraphs.

The first way to classify a new project execution would be to apply the procedure of Triesch et al. (2012). If the Parkinson distribution with a log-normal core is confirmed for that project execution, we can continue with the estimates $\hat{m}$ and $\hat{s}$. Levene's and Welsch's tests can then be deployed to check whether the natural logarithm of the relative distribution of the new project execution has respectively an equal standard deviation or an equal mean with any of the groups from the analysis in this paper. This would identify the project execution as belonging to a single class of table 4.3. However, this would mean that the empirical records, that were needed to obtain the classification in table 4.3, need to be consulted for each new project execution.

Therefore, we propose an alternative procedure that relies only on the estimates for $\hat{s}_{\mathrm{pooled}}$ and $\hat{m}_{\mathrm{pooled}}$ of table 4.3. Again, for each new project execution, the Parkinson distribution with a lognormal core would need to be confirmed, and the $\hat{m}$ and $\hat{s}$ estimates calculated. We then suggest to calculate a confidence interval (e.g. the 95% confidence interval) for both the $\hat{s}$ and $\hat{m}$ value. Then, for each cell in table 4.3, one should compare the $\hat{m}_{\mathrm{pooled}}$ and $\hat{s}_{\mathrm{pooled}}$ values for the corresponding groups to the appropriate confidence interval. If the $\hat{m}_{\mathrm{pooled}}$ for a certain group (LOW $\hat{m}$, MID $\hat{m}$ or HIGH $\hat{m}$) is contained within the confidence interval around $\hat{m}$ and if the $\hat{s}_{\mathrm{pooled}}$ value of a certain group (LOW $\hat{s}$ or HIGH $\hat{s}$) is contained within the confidence interval around $\hat{s}$, then we can assume that the project execution belongs to the class of table 4.3 that is characterised by these $\hat{m}_{\mathrm{pooled}}$ and $\hat{s}_{\mathrm{pooled}}$ values. We will illustrate this with examples in the discussion and conclusions section.

We will now discuss the use of the classification of table 4.3 for construction projects. As noted earlier, the database of [1] predominantly contains records of construction projects. However, we included some other sectors (IT and education) to produce the classification in this section. Most notably, the "Organizational Development" project (education) has influenced the proposed classification greatly. The empirical evidence shows that, for construction projects, only the two classes in the top left corner of table 4.3 should be considered. Since, in the dataset of [1], the construction projects were found to belong to the LOW $\hat{m}$ and MID $\hat{m}$ groups with respect to the mean of the natural logarithm of the relative empir-

ical distributions, and to the LOW $\hat{s}$ group with respect to the standard deviation of the natural logarithm of the relative empirical distributions. For these two classes, we calculated the coefficient of variation (c.v.) with c.v.$^{\dagger} = \sqrt{e^{\hat{s}^2_{\text{pooled}}} - 1}$. The value of 0.5 for the LOW $\hat{s}$ group confirms the findings of [21] for the construction industry, as it is between their values of 0.3 and 0.8, found for respectively low and mid variance construction projects.

We will discuss how the empirical classification of table 4.3 can be put to work in future simulations studies for project management in the next section.

	$\hat{s}_{\text{pooled}}$	
$\hat{m}_{\text{pooled}}$	0.47	3.06
-0.09	LOW $\hat{m}$/LOW $\hat{s}$	LOW $\hat{m}$/HIGH $\hat{s}$
0.26	MID $\hat{m}$/LOW $\hat{s}$	MID $\hat{m}$/HIGH $\hat{s}$
1.05	HIGH $\hat{m}$/LOW $\hat{s}$	HIGH $\hat{m}$/HIGH $\hat{s}$

Table 4.3: Empirical classification of project executions

4.5 Discussion and conclusions

In this section we will first provide a discussion to the empirical classification that is proposed in this paper. The discussion constitutes three possible uses for the empirical classification in the project management practice. We validated these three possible uses in a case study, performed in a construction company. Second, we will present overall conclusions for the study that is presented in this paper.

4.5.1 Discussion

We will validate the use of the empirical classification, presented in table 4.3, using a case study of the project management practice in a construction company. We will first discuss three projects that were performed in a construction company. We will then show three possible uses for the classification in table 4.3. In combination with the empirical data from the

---

†The formula for the c.v. of a lognormally distributed variable X follows from: $\mathrm{E}[X] = e^{m+s^2/2}$ and $\mathrm{sd}[X] = e^{m+s^2/2}\sqrt{e^{s^2}-1} = \mathrm{E}[X]\sqrt{e^{s^2}-1} \implies \sqrt{e^{s^2}-1} = \mathrm{sd}[X]/\mathrm{E}[X] = $ c.v. The reader should note that the c.v. is independent of the expected value of a lognormally distributed variable and therefore, $\hat{m}_{\text{pooled}}$ is not incorporated in the formula.

construction company, we show how table 4.3 can lead to the implementation of a SPC project control tool, the planning of management reserve and execution of a schedule risk analysis.

Analysis of construction projects

We obtained data from three residential building projects that were performed by a Belgian construction company in the beginning of 2014. An associated researcher worked as a project management consultant for this construction company in the first six months of 2014, providing both planning and control services. Due to confidentiality, we abstracted only the necessary details from these projects and made them available online in R data format[‡]. These new projects, performed by a single construction company, are currently being processed to be included in the database of [1], and can be retrieved using the names C2014-05, C2014-06 and C2014-07. These projects constitute a homogeneous set in terms of their scope (structural construction work), duration (30 - 70 days) and budget (140,000 € - 210,000 €). A qualitative analysis indicated that from the three projects, only C2014-06 was perceived as successful by the project management team that was responsible for the execution of the projects. A quantitative analysis supported this finding. For project C2014-06, a budget overrun and schedule delay of around 10% were observed, while the two other projects experienced a budget overrun of around 17%-19% and were finished with a 24% schedule delay.

Table 4.4 shows the result of the four-step procedure of [21] applied to projects C2014-05, C2014-06 and C2014-07. The estimates $\hat{m}$ and $\hat{s}$ for these projects are also depicted in figure 4.5a, along with their 95% confidence limits. Figure 4.5(a) allows us to identify the projects as belonging to either of the empirical classes that were presented in this study. From the $\hat{m}$ and $\hat{s}$ grid in figure 4.5a, we were able to classify projects C2014-05 and C2014-07 as MID $\hat{m}$/LOW $\hat{s}$ and project C2014-06 as LOW $\hat{m}$/LOW $\hat{s}$. This detailed inspection of the activity durations for these projects explains the budget overruns and schedule delays that were observed by the project management team. Moreover, the empirical classification effectively allows the quantification of the subjectively perceived "successfulness" of project C2014-06, by the project management team. In the construction company that is the subject of our case study, a project belonging to the LOW $\hat{m}$/LOW $\hat{s}$ class is perceived as successful. Consequently, this class can be used as a reference for future project executions. We will now show how this class can be used in three project simulation applications, which incorporate the projects of the validation set. In addition, the reader should

---

[‡]"Validation Dataset" at the `statistical project control` page on `http://www.projectmanagement.ugent.be`

Project			Parkinson distribution with lognormal core			
			$\hat{m}$ $\hat{s}$			
			(*p*-value)			
Number	Name	P0	P1	P1.1	P1.2	
C2014-05	*Structural construction works 1*		0.37 0.30			
			(0.09)			
C2014-06	*Structural construction works 2*		0.05 0.33			
			(0.61)			
C2014-07	*Structural construction works 3*		0.34 0.31			
			(0.21)			

Table 4.4: Discussion of the activity times data of the validation dataset from the case study in a Belgian construction company

note that the use of the two top left classes of table 4.3 for construction projects is validated by these independent data.

A statistical process control (SPC) application for project management

The implementation of a SPC procedure for project control was proposed for the construction company, based on the preceding classification of the projects in the case study. For such a SPC procedure for project control, a major prerequisite is the ability of the process (in this case the projects that are executed) to "speak for itself" [18] and the control approach should be able to accurately reflect the "voice of the process" [28]. Effectively, this means that empirical data from the construction company needed to be incorporated in the SPC procedure. Figure 4.5b shows the SPC procedure proposed by [11]. A recent study [3] shows that this approach outperforms the other SPC procedures that were proposed in the literature for project control. For details on the calculation of the control limits (and consequently, the zones A, B and C) and the simulation from which these can be calculated, the reader is referred to the original article and the comparative study. Here, it suffices that the simulations incorporate the LOW $\hat{m}$/LOW $\hat{s}$ class into the definition of the "in control" reference. Moreover, the observations for project C2014-05 were added to the SPC chart (diamonds). If this had been done dynamically during the execution of the project, the SPC chart would have signalled an "out of control" situation when the project was around 50% completed, on account of more than 4 consecutive observations falling within the C zone [11, 27]. Project C2014-05 would have been identified as unlikely to be belonging to the LOW $\hat{m}$/LOW $\hat{s}$, and the project management team would have been given an early warning signal on which they could have made a decision to steer the project back towards this class. In conclusion, figure 4.5b shows how the empirical classification of table 4.3 allows the implementation of SPC charts for project control.

Validation projects

**Leu and Lin (2008)
SPC application**
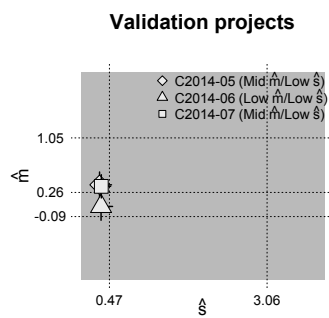
(a) $\hat{m}$ and $\hat{s}$ for the validation dataset
(C2014-05, C2014-06, C2014-07)
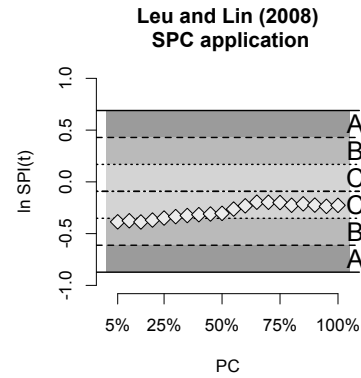
(b) A SPC application for project control,
depicting project C2014-05

**Management reserve
planning application**

**Schedule risk analysis
application**
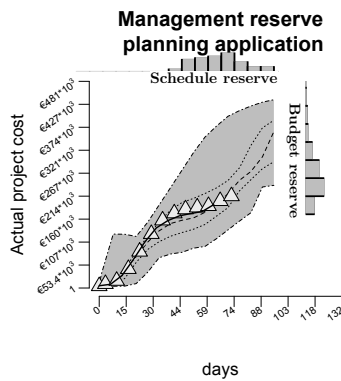
(c) A management reserve planning
application for project C2014-06
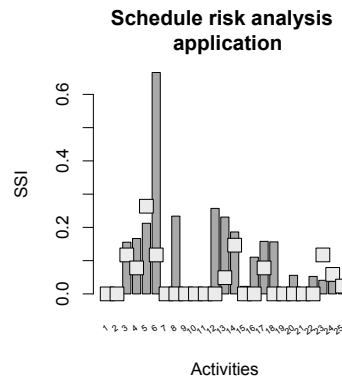
(d) A schedule risk analysis application
for project C2014-07

Figure 4.5: Validation of the empirical classification for project executions on a
case study in a Belgian construction company

Management reserve planning application for project management

In the construction company that is the subject of our case study, the practice of running MCS in order to estimate possible budget overruns and schedule delays was not previously adopted. The planning of a management reserve was therefore done consistently based on personal experience and rules-of-thumb. The main reason provided for this was that it requires a detailed input analysis, for which the know-how was not present within

the construction company. After the identification of the successful histor-ical project (C2014-06) as belonging to the Low $\hat{m}$/Low class, we showed them how this class could serve as an input to future simulation studies. Figure 4.5c shows a grey area of actual cost curves for fictitious executions of project C2014-06 that would belong to the Low $\hat{m}$/Low class. The sim-ulation that was used to produce these fictitious project executions was de-scribed in [3]. The probability of these actual cost curves is indicated by the dotted lines that are superimposed onto the grey area. From top to bottom, these respectively represent the 100th, the 75th, the 50th, the 25th and the 0th quantile. The full black line represents the baseline schedule for project C2014-06 and the real execution is depicted using white triangles. Even though the simulation was run after the completion of the project, it could have been done as easily during its planning and negotiation phase. This would have given the construction company insights into the risks that they are exposed to when performing the project. A "schedule reserve", could be taken into consideration when engagements are made with re-spect to the actual duration of the project with the project owner or the other possible stakeholders. In addition, a "budget reserve" could be taken into account when a price is set for the execution of the project in a contrac-tual agreement. The probabilistic outputs of the simulation for the actual cost and the total project duration were added to figure 4.5c as histograms opposing respectively the y-axis and the x-axis. These histograms, and the information that can be retrieved from them, show the value of our empir-ical classification for Monte Carlo simulations in the project planning and negotiation phase.

Schedule risk analysis application

A third possible use for the proposed empirical classification is in the appli-cation of a schedule risk analysis (SRA) for project management. SRA are not conducted regularly for projects in the construction company, due to the difficulties that are associated with deriving a probabilistic input for the MCS. However, given the identification of the successful historical project as belonging to the Low $\hat{m}$/Low class, the required input analysis is dras-tically reduced. Figure 4.5d shows the SRA that was conducted post-hoc for project C2014-07. Here, the schedule sensitivity index (SSI) is shown as grey bars for all activities. Although our approach is not restricted to the SSI, we limit the illustration to this single indicator, since it was found to deliver the best possible results in a project control application [23]. For more details regarding the characteristics and the calculations required in a SRA, we refer to more extensive studies on the subject [5, 22]. Moreover, we also show the observed SSI for the real execution of project C2014-07, using white squares in figure 4.5d. Since the real execution of C2014-07 was found to belong to the Mid $\hat{m}$/Low class (see figure 4.5a), substantial

differences in the SSI for certain activities can be observed. This indicates a possibly larger variance for activity durations with respect to the LOW $\hat{m}$/LOW class, which confirms the findings of the classification in figure 4.5d. Moreover, figure 4.5d illustrates that our classification allows to test the robustness of a SRA, which shows its value for the development and testing of project control approaches which are based on SRA.

4.5.2 Conclusion

In this paper, we have analysed the empirical data for activity distributions of an existing dataset using the distributional model of [21]. The purpose of this application was to show the relevance of such an analysis in future project management studies in which data of new projects will be used. This new data, coming from historical observations or simulation studies, can then be compared and validated against the classification, shown in this paper. The potential value of our classification has been demonstrated on a case study in a construction company.

The analysis that was conducted in this paper was made possible by the extensive dataset of project executions provided by [1]. To this dataset, we applied the distributional model and calibration procedure that were proposed by [21]. This model assumes the relative empirical distribution $d_{ij}/\hat{d_{ij}}$ to be lognormally distributed, when the Parkinson effect and the occurrence of rounding errors are correctly accounted for, known as the Parkinson distribution model with a lognormal core. The large majority of the historical records in the database of [1], that were suited for this analysis, are from the construction industry. For 54% of these records, we confirmed the Parkinson distribution model with a lognormal core.

During our analysis, we obtained estimates for the standard deviation $\hat{s}$ and the mean $\hat{m}$ of the natural logarithm of the relative empirical distributions ($d_{ij}/\hat{d_{ij}}$). Using these estimates, we found respectively two and three groups of project executions for $\hat{s}$ and $\hat{m}$. The validity of these groups was confirmed using a statistical analysis with Levene's test for the homogeneity of variances and Welsch's test for equal means. The two groups that were found for $\hat{s}$ have values 0.47 and 3.06 for the pooled standard deviation of the natural logarithm of the relative empirical distributions. The three groups for $\hat{m}$ have values -0.09, 0.26 and 1.05 for the pooled mean of the natural logarithm of the relative empirical distributions. For construction projects, we confirmed the findings of [21], with respect to the observed coefficients of variation of 0.5 for the LOW $\hat{s}$ group of project executions, and validated the use of the LOW $\hat{m}$/LOW and MID $\hat{m}$/LOW on independent data.

The grouping of the historical records from the dataset of Batselier and

Vanhoucke (2014) into two and three groups, for respectively $\hat{s}$ and $\hat{m}$, allowed us to propose an empirical classification for project executions, using 6 classes. Subsequently, we showed how this classification can be useful in the project management practice using a case study in a construction company. The empirical classification can be deployed to develop SPC charts for project control, for planning applications of the management reserve and for SRA studies. It is also believed that our classification method and the suggested applications can lead to more generalisable results in future research in the project management domain.

References

[1] Batselier, J. and Vanhoucke, M. (2014). "Construction and evaluation framework for a real-life project database." *International Journal of Project Management*.

[2] Blom, G. (1958). "Statistical estimates and transformed beta-variables.

[3] Colin, J. and Vanhoucke, M. (2014a). 'Developing a framework for statistical process control approaches in project management" *International Journal of Project Management*.

[4] Colin, J. and Vanhoucke, M. (2014b). "Setting tolerance limits for statistical project control using earned value management." *Omega The International Journal of Management Science*, 49, 107–122.

[5] Elmaghraby, S. (2000). "On criticality and sensitivity in activity networks." *European Journal of Operational Research*, 127, 220–238.

[6] Elshaer, R. (2013). "Impact of sensitivity information on the prediction of project's duration using earned schedule method." *International Journal of Project Management*, 31, 579–588.

[7] Gutierrez, G. and Kouvelis, P. (1991). "Parkinson's law and its implications for project management." *Management Science*, 37, 990–1001.

[8] Hahn, E. (2008). "Mixture densities for project management activity times: A robust approach to PERT." *European Journal of Operational Research*, 188, 450–459.

[9] Kuhl, M. E., Lada, E. K., Steiger, N. M., Wagner, M. A., and Wilson, J. R. (2007). "Introduction to modeling and generating probabilistic input processes for simulation." *Proceedings of the 2007 Winter Simulation Conference*, S. Henderson, B. Biller, M. Hsieh, J. Shortle, J. Tew, and R. Barton, eds., New Jersey: Institute of Electrical and Electronics Engineers, 63–76.

[10] Kwak, Y. H. and Ingall, L. (2007). "Exploring monte carlo simulation applications for project management." *Risk Management*, 9(1), 44–57.

[11] Leu, S. S. and Lin, Y. C. (2008). "Project performance evaluation based on statistical process control techniques." *Journal of Construction Engineering and Management*, 134, 813–819.

[12] Levene, H. (1960). "Robust tests for quality of variances. in ?contribution to probability and statistics: essays in honour of harold hotelling?.(ed. i olkin) pp. 278–292." *Stanford University Press: Redwood City, CA.) Libosvarsky, VA, Saeed, DW, and Nemcova, M.(1985). Fecundity of roach in a newly built reservoir. Folia Zoologica*, 34, 357–372.

[13] Looney, S. W. and Gulledge Jr, T. R. (1985). "Use of the correlation coefficient with normal probability plots." *The American Statistician*, 39(1), 75–79.

[14] McCullagh, P. and Nelder, J. (1989). *An outline of generalized linear models*. Springer.

[15] Mohan, S., Gopalakrishnan, M., Balasubramanian, H., and Chandrashekar, A. (2007). "A lognormal approximation of activity duration in pert using two time estimates." *Journal of the Operational Research Society*, 58(6), 827–831.

[16] R Core Team (2013). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, <http://www.R-project.org>.

[17] Ragsdale, C. (1989). "The current state of network simulation in project management theory and practice." *Omega The International Journal of Management Science*, 17(1), 21–25.

[18] ReVelle, J. B. (2004). *Quality Essentials: A reference guide from A to Z*. ASQ Quality Press.

[19] Schonberger, R. (1981). "Why projects are "always" late: A rationale based on manual simulation of a PERT/CPM network." *Interfaces*, 11, 65–70.

[20] Trietsch, D. and Baker, K. R. (2012). "Pert 21: Fitting PERT/CPM for use in the 21st century." *International Journal of Project Management*, 30(4), 490–502.

[21] Trietsch, D., Mazmanyan, L., Govergyan, L., and Baker, K. R. (2012). "Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation." *European Journal of Operational Research*, 216, 386–396.

[22] Vanhoucke, M. (2010a). *Measuring Time - Improving Project Performance using Earned Value Management*, Vol. 136 of *International Series in Operations Research and Management Science*. Springer.

[23] Vanhoucke, M. (2010b). "Using activity sensitivity and network topology information to monitor project time performance." *Omega The International Journal of Management Science*, 38, 359–370.

[24] Vanhoucke, M. (2011). "On the dynamic use of project performance and schedule risk information during project tracking." *Omega The International Journal of Management Science*, 39, 416–426.

[25] Wauters, M. and Vanhoucke, M. (2014). "Support vector machine regression for project control forecasting." *Automation in Construction*, 47, 92–106.

[26] Welch, B. L. (1947). "The generalization ofstudent's' problem when several different population variances are involved." *Biometrika*, 28–35.

[27] Western Electric Company and Small, B. B. (1958). *Statistical Quality Control Handbook*. AT&T.

[28] Wheeler, D. J. (1995). *Advanced topics in statistical process control*, Vol. 470. SPC press Knoxville, TN.

[29] Williams, T. (1999). "Towards realism in network simulation." *Omega The International Journal of Management Science*, 27, 305–314.

5

Developing a framework for statistical process control approaches in project management

Modelling of the current state of EVM/ES project control methods and its derivatives	Development of a framework to test and compare EVM/ES project control methods	Single- and multivariate advances to top-down project control using EVM/ES

Top-down
- Rules of thumb (RoT)
- **Statistical process control approaches (SPC)**
- Critical Chain / Buffer Management (CC/BM) derivatives

Bottom-up
- Longest path method (LP)

- Project dataset
- Risk and variation modelling
- Two-phased Monte Carlo simulation (MCS)
- Measuring project control performance

Empirical perspective

Single
- Statistical tolerance limits (STL)

Multi
- Multivariate statistical project control (MSPC)
- Multivariate regression methods for longest path calculations (MR-LP)

Figure 5.1: Schematic overview of the elements, of the larger body of research in this book, that are discussed in chapter 5.

abstract

Different statistical process control (SPC) approaches have been proposed over the years for project management using earned value management/earned schedule (EVM/ES). A detailed examination of these approaches has led us to express a need for a unified framework in which to test and compare them. The main drivers for this need were the lack of a formal definition for a state of control, the unavailability of a benchmark dataset, the absence of measures to quantify the SPC performance and the lack of consensus on how to overcome and test the normality assumption. In this paper, we present such a framework that combines a classification from empirical data, a well known project dataset, a sound simulation model and two quantitative measures for project control efficiency. Four SPC approaches from prior literature have been implemented and an exhaustive experiment was set up to compare and to discuss their value for the project management practice.

5.1   Introduction

Research on the construction of the project baseline schedule has been an integral part of the project management literature, ever since the first occurrence of project planning approaches such as the critical path method (CPM, [23]), the program evaluation and review technique (PERT, [29]) and

more advanced algorithms to construct a schedule under the limited availability of resources [17]. During the execution of a project, the baseline schedule is reduced to a single point of reference for the actual progress that has been made. According to [38], a project control system is required to produce a reliable indication of the direction of change when actual performance is compared with preliminary planning variables. Consequently, an efficient project control system should generate a warning signal when the gap between planning and performance becomes unacceptably large, in order to allow the project manager to take corrective actions.

Earned value management/earned schedule (EVM/ES) is such a system that generates performance metrics from which, after interpretation, signals can be acquired and corrective actions can be planned. This interpretation however, is impeded by the unintuitive dynamics that some of these performance metrics suffer from. Therefore, the use of EVM/ES is often characterized in practice by decision making from practical experience, rules-of-thumb and anecdotal evidence [9]. In order to overcome this problem and to assist project managers in controlling a project, statistical process control/statistical quality control (SPC/SQC, [37]) techniques have been implemented to determine action limits on the EVM/ES performance metrics. In this paper, we will discuss the approaches published by [4], [48], [25] and [1]. We will discuss the approaches of these aforementioned authors in greater detail in section 4.2. These authors propose distinct solutions for the problems associated with the implementation of SPC/SQC charts and often apply them to different performance metrics from the EVM/ES system. Moreover, all of these project control systems have been validated on different datasets from different case studies of project executions. In order to compare their performance, we present a unified framework in which we will conduct an exhaustive simulation experiment.

The outline of this paper is described along the following lines. Section 5.2 reviews the SPC/SQC methods that were proposed for schedule control using EVM/ES. Section 5.3 presents the project simulation framework that is used to quantify the performance of the SPC/SQC methods in extensive simulation experiments. We will show the results of the experiments in section 5.4 and the conclusions of this work are presented in section 5.5.

5.2   Literature overview

5.2.1   Earned value management/earned schedule

Since its development in the 1960's as a unified methodology for cost and schedule control for US. Department of Defence projects, EVM has received a lot of attention in the academic literature [2, 7, 15, 20, 24, 32]. A noteworthy extension to the EVM methodology, to improve its schedule control capabilities, was proposed by [28] in the form of earned schedule (ES). Their combined use, abbreviated as EVM/ES, has been integrated with risk management in the large simulation experiment by [46].

In EVM/ES, a project performance baseline is calculated from initial estimates for the activity costs and durations. The monetary value of the work that is supposed to be done at a given time is the planned value (PV). During project progress, the budgeted cost of the work that is actually performed (earned value, EV) is compared with the PV. From this comparison the schedule variance and the schedule performance index can be calculated:

$$\begin{aligned} \text{SV} &= \text{EV} - \text{PV} \quad \text{Schedule variance} \\ \text{SPI} &= \text{EV/PV} \quad\;\; \text{Schedule peformance index} \end{aligned}$$

The schedule performance metrics are based on the monetary values EV and PV. Consequently, SV is expressed in monetary units. To overcome this, [28] defined the ES, which is expressed in time units (days, weeks, months, ...) and can therefore be directly compared to the time that has passed since the beginning of the project (actual time, AT). This leads to two additional performance metrics, the schedule variance using ES (SV(t)) and the schedule performance index using ES (SPI(t)):

$$\begin{aligned} \text{SV(t)} &= \text{ES} - \text{AT} \quad \text{Schedule variance using ES} \\ \text{SPI(t)} &= \text{ES/AT} \quad\;\, \text{Schedule peformance index using ES} \end{aligned}$$

The reader is referred to the original work by [28] and the detailed discussion by [46] for a background on the calculations of the EVM/ES system.

5.2.2   Statistical process control/statistical quality control

The field of SPC encapsulates all methods which are applied to monitor and to control processes in services and manufacturing. SQC focusses on the outputs of the process rather than the process itself [37]. Despite this distinction, the two terms have been used interchangeably in the context of statistical control for project management. The schedule control metrics

from EVM/ES can be used both to monitor the instantaneous health of the project as to forecast final project duration and therefore, both SPC and SQC are applicable. For the remainder of this paper, we refrain from the use of the term SQC and proceed with SPC.

SPC was pioneered by W. Shewhart at Bell laboratories in the early 1920's. He developed control charts to identify significant variation in the process before it could lead to sub-standard process outputs. At any give time, a process is said to be potentially under the influence of two classes of variation [40]. "Common cause" or "non assignable" sources of variation act constantly on the process and produce a stable and repeatable distribution of outputs over time. "Special cause" or "assignable" sources affect only some of the process outputs and should be removed. In SPC, control charts always attempt to differentiate between the two classes of variation in order to eliminate the latter from the process.

A SPC system is required to use variables that allow "the process to speak for itself." Control limits are therefore derived from the recorded variables of the ongoing process when special cause sources of variation have been removed. Over the years, a large number of different control charts have been developed to cope with different characteristics and assumptions for the process variables (CUSUM [34], EWMA [22], . . .). One major assumption for most of these is that the recorded samples need to be normally distributed variables. Although this assumption is not explicitly stated for all control charts, such as the individual moving range (I-mR) chart proposed by [40], their use on non-normally distributed data is strongly discouraged [31].

5.2.3   SPC for project schedule control

SPC charts have been applied to schedule control in four research studies by [4], [48], [25] and [1]. These SPC control chart implementations provide distinct solutions to a common problem, which is how to set action limits for schedule control during project progress, based on a "state of statistical control" reference. Table 5.1, and the succeeding paragraphs, present the different approaches of the aforementioned authors with respect to the choice of control charts, how normality is tested, how is dealt with normality, what reference is used to define the state of statistical control and how the approach is validated.

Control Chart
Both Leu and Lin (2008) and Aliverdi et al. (2013) deploy Shewhart's individuals (I) and moving range charts (mR). This combination of two charts

		SPC for schedule control		
	[4]	[48]	[25]	[1]
Control chart	CUSUM	Shewhart	I-mR	I-mR
	Custom rules	-	Western Electric rules	Nelson's rules
Normality test	-	SK	KS	χ^2/KS/AD
Normality	-	+	$\ln$ and $\ln^{-1}$	Box-Cox/Johnson
Reference	20 similar projects	historical	44-53 similar projects	historical
Validation data	6 projects (CAPP)	1 project (-)	73 projects (-)	1 project (-)

Table 5.1: Discussion of the characteristics of SPC procedures for project control

(the first plots the moving range and the second monitors the standard deviation of the individual observations) is optimized to monitor variable data from a process for which it is impractical to use subgroup averages. Since a project is characteristically a one-time endeavour, it is distinctly different from a process for which at a given time an average can be drawn for a group of products or services flowing through the process. Therefore, the individuals and moving range chart is best suited to the characteristics of a project. Leu and Lin (2008) implement the additional run rules from the Western Electric company [10] and Aliverdi et al. (2013) those from Nelson [33]. [48] deploy a "Shewhart" chart for monitoring the individual observations. For this chart, the observations are plotted against three sigma limits, for which the standard deviation is estimated using historical data. The moving range was not investigated. [4] implement a CUSUM chart for project control, with additional custom rules. The CUSUM chart is conjectured to incorporate the increasing nature of period-to-period measurements of project performance. As an additional adaptation of traditional CUSUM charts, they normalize the different time lengths of all projects in their dataset to incorporate the finite time-span of projects.

Normality and normality tests
Given the predicted unreliable behaviour of SPC charts for non-normally distributed variables, Wang et al. (2006), Leu and Lin (2008) and Aliverdi et al. (2013) implement hypothesis tests to validate the assumption of normality of the observed project data. This is done by a range of goodness-of-fit tests under the null hypothesis of normally distributed data. The skewness-kurtosis test (SK, [11]), the Kolmogorov-Smirnov test (KS, [30]) and Pearson's chi-squared test (χ^2, [35]) have been successfully applied in these studies. If the null hypothesis is rejected for a given level of significance, Leu and Lin (2008) and Aliverdi et al. (2013) propose a transformation of the raw project data. Leu and Lin (2008) follow [27]'s suggestion to use the logarithm (ln) or the logarithm of the inverse ($\ln^{-1}$) to normalize their data, while Aliverdi et al. (2013) use the Minitab implementations of the Box-Cox and Johnson transformation. Bauch and Chung (2001) do not test this assumption, implicitly conjecturing their approach to be robust for all types of distributions of the observed project data. They have however, proposed a custom transformation of the project data in order to incorporate the finite time-span of projects, as discussed in the previous paragraph. Wang et al. (2006) confirmed the normality assumption for their data, hence the "+" in table 5.1. Consequently, they implicitly assumed all future data to be normally distributed and did not propose any transformation.

State of statistical control reference

The four research articles on SPC for project control all use a different mechanism for selecting the project data from which the control limits are derived. Two classes can be observed. Either the control limits are calculated from data observed for *similar* projects that have been successfully completed or *historical* data is recorded in the early stages of the project execution phase. In the former case, the data from similar projects are assumed to be representative. Bauch and Chung (2001) note that the projects that are used should be very similar in terms of scope, duration, and environment, although some adjustments to the individual projects may be made, for size and length, by normalizing processes. In the latter case, data are recorded in the early stages of the on-going project to determine a statistical control reference. Aliverdi et al. (2013) provide a 30 month sample of project progress data from which the control limits are calculated. These control limits are then used to monitor the performance of 10 additional months. Wang et al. (2006) illustrate their approach on 18 SPI-values from which only a post-hoc analysis of the project execution is performed.

Validation data

The proposed SPC charts for project control have all been validated on different datasets. Bauch and Chung (2001) discuss the performance of their approach on 6 projects from Russel and Lawrence's Continuous Assessment of Project Performance (CAPP, [39]) tool, while the other authors do not offer a reference for their data, hence the "(-)" in table 5.1. Wang et al. (2006) and Leu and Lin (2008) add detail to their control chart validation in the form of a cause-and-effect discussion.

Summary observations from prior literature on SPC for project control

From the discussion in this section of the prior literature on SPC for project control, summarized in table 5.1, we observed that:

- There is no explicit definition for a state of statistical control in project management;

- There is no unified project benchmark on which the SPC methods have been validated;

- No quantitative measures have been used to compare performances between methods;

- There is no consensus on normality or transformations for normality on project EVM/ES data.

These observations allow us to express a need for a unified framework in which SPC approaches for project control can be tested. In section 5.3
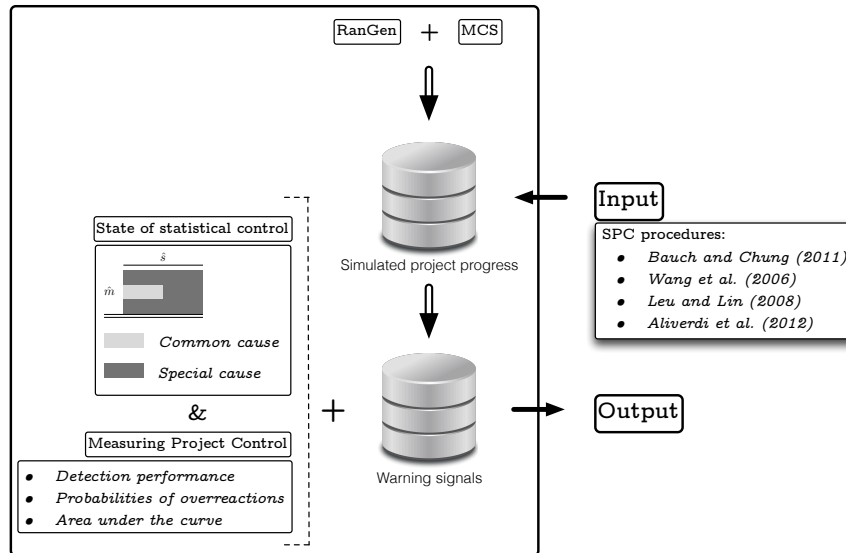
Figure 5.2: Schematic representation of the proposed framework for statistical project control

we propose such a framework based on the following four characteristics.

- A state of statistical control for projects will be defined by letting the "process speak for itself" using an empirical database of project executions;

- A project dataset will be implemented on which all methods can be tested;

- A recently published Monte Carlo simulation model will be applied;

- Two quantitative metrics will be proposed to compare SPC methods for project control.

Using this framework, we will test and compare the proposed SPC approaches with respect to their ability to distinguish between common cause and special cause sources of variation and the the normality assumption for all proposed transformations.Results for these tests along with a discussion of their limitations will be presented in section 5.4

5.3 Methodology

In this section, we present the framework in order to compare the SPC procedures, from prior literature for project control. Figure 5.2 shows a schematic representation of this framework. In this section, we will focus on the actual framework (outlined using a thick black line) on the left of the schematic overview. The inputs (the SPC approached form prior literature) and outputs (the results of our experiments) will be described in section 5.4.

At the top, figure 5.2 shows the backbone technologies that have been applied in order to build this framework. More specifically, we combined a dataset of project networks (from RanGen, [12]) with a Monte Carlo simulation (MCS) model. These are described in sections 5.3.1 and 5.3.2 respectively. The application of the MCS to the dataset of project networks results in a database of simulated project progress situations. This set contains records of fictitiously executed projects, onto which the inputs (SPC approaches) can be applied. Consequently, this produces a database of warning signals for the fictitious project executions that can be applied for schedule control. In order to quantify the performance of these warning signals, we need to asses whether they correctly indicate that special cause sources of variation are present in the project execution. Therefore, we apply a definition of a state of statistical control (section 5.3.3) and measure the performance of the warning signals, using a combination of two metrics for the effectiveness of project control efforts (section 5.3.4). Ultimately, this results in the output that will be described in section 5.4.

5.3.1 Project benchmark dataset for project control

We implemented the set of 900 project networks that was previously presented by [45]. This dataset was used in previous simulation studies on project control systems [9, 13, 45, 46]. The 900 projects were generated using the RanGen algorithm [12] in accordance with the serial/parallel (SP) indicator, which measures how close a given project network is to a completely serial or parallel network. These projects were generated with such diversity with respect to the network topology that averaged values over all projects are conjectured to be general enough, so that they can be extrapolated for all possible network structures.

All projects contained 30 activities for which the baseline estimate duration was uniformly sampled between 8 days and 56 days. In addition, fixed and variable costs are assigned to the activities, which were drawn from uniform distributions between €0 and €500 and €700 and €1500 respectively. The number of activities and the ranges for the baseline estimate

durations and the assigned costs conform with more than 80% of the empirical database of activity information [3]. We therefore consider them to be representative for realistic small to medium scaled projects.

5.3.2 Simulating project progress situations

In order to produce fictitious project progress situations, we applied the Monte Carlo simulation model that was proposed by [9]. This simulation employs a linear association variable [42] to model dependencies between activities and a uniform distribution [16] to add variation.

The simulation model was implemented in P2 Engine [47]. A Monte Carlo draw from the linear association variable and one from a uniform distribution yield a real duration for each activity for the fictitious executions of a project. These real durations define the fictitious project execution. For each fictitious project execution, the real schedule could then differ from the baseline schedule, which was previously planned according to the earliest possible start of the activities. From these two schedules the EV and the PV were calculated. For each activity, the EV and the PV followed a linear accrue, starting from zero at its start up to its budget at completion (BAC) when it was finished. Consequently, at the aggregate project level the SV, SPI, SV(t) and SPI(t) could be calculated.

5.3.3 Defining a state of statistical control for project management

In order to quantify the performance of the warning signals that were produced by the SPC approaches (the inputs to our framework in figure 5.2), we needed to assess whether they correctly indicate departures from a state of statistical control in the project execution. In other words, a SPC approach for project control should produce a warning signals if special cause sources of variation are present in addition to the omnipresent common cause sources.

This state of statistical control should accurately represent the 'voice of the process" [49]. The transfer of this concept to the problem of project schedule control inspired us to apply the classification of empirical project records that was proposed by [8]. Before going into the details of this classification and how it has come to be, we will motivate why it is a suitable basis for our definition of a state of statistical control.

Since the classification was produced from empirical records of project executions, it is able to reflect "the voice of the process". Therefore, we first

	$\hat{s}$	
$\hat{m}$	0.47	3.06
-0.09	LOW $\hat{m}$/LOW $\hat{s}$	LOW $\hat{m}$/HIGH $\hat{s}$
0.26	MID $\hat{m}$/LOW $\hat{s}$	MID $\hat{m}$/HIGH $\hat{s}$
1.05	HIGH $\hat{m}$/LOW $\hat{s}$	HIGH $\hat{m}$/HIGH $\hat{s}$

Table 5.2: Empirical classification of project executions

considered the activity level progress of project as "the voice". The activity level progress is expressed in the distribution of the relative (real over planned) duration of the activities in the empirical records, from which the classification by [8] was produced. Second, we considered the execution of the project's schedule as "the process". This process can only be observed indirectly trough the EVM/ES system. The reader should recall that [28] regarded activity level control as a cumbersome and often disruptive task for the project manager and reasoned that the control should be done at a higher level of the work breakdown structure (WBS) of the project. This control process can therefore benefit from a SPC approach in order to distinguish situations where special cause sources of variation are present from the "state of statistical control" situations, where only common cause sources of variation are present.

Table 5.2 presents the empirical classification of [8], that we have applied in order to define a state of statistical control for project management. This classification was developed by a statistical analysis of 51 real projects from the database of [3]. To start, the $\hat{m}$ and $\hat{s}$ values for all projects were calculated. These represent estimates for two parameters that characterize the distribution of the relative durations of the activities. $\hat{m}$ and $\hat{s}$ represent respectively the mean and standard deviation of the natural logarithm of this distribution. According to [43], empirical records of relative activity durations follow a Parkinson distribution with a lognormal core and therefore, the two aforementioned parameters can be used to fully characterize this distribution. Using these $\hat{m}$ and $\hat{s}$ values, [8] showed that the project executions, from the dataset of [3] can be classified into six classes. Since, with respect to $\hat{s}$, these projects were shown to belong to either a LOW or $\hat{s}$ group. With respect to $\hat{m}$, each project in the dataset could be shown to belong to either a LOW, MID or HIGH group. The combinations of these three groups for $\hat{m}$ and the two groups for $\hat{s}$ result in the six classes that are shown in table 5.2. The values for $\hat{m}$ and $\hat{s}$, that can be found in the margins of table 5.2, represent the real underlying parameter values, which should be observable in practice.

We will now illustrate how the classification in table 5.2 leads to the definition of a state of statistical control that is applied in our framework. First,

let us assume a hypothetical project environment for which all historical projects can be shown to belong to the MID group with respect to $\hat{m}$ and to the LOW group with respect to $\hat{s}$. These historical projects can then be said to belong to the MID $\hat{m}$/LOW $\hat{s}$ class of table 5.2. This class is depicted in grey in table 5.2. Consequently, we can now define the state of statistical control for this project environment as determined by this class of table 5.2. Moreover, we then conjecture that projects from the MID $\hat{m}$/LOW $\hat{s}$ class are only influenced by the common cause sources of variation that are constantly active in that same project environment. If a new project is executed in this project environment, it is fair to believe that in accordance with historical data, the project is likely to be categorizable into the same class. However, since each project is a one-time endeavour, executed in a stochastic setting, there is a probability that other (assignable cause) sources of variation might show up during the execution. SPC charts should therefore be employed to generate early warning signals if the variation in the project is driven by special cause sources of variation, in order to allow the project manager to plan corrective actions to remove these sources.

5.3.4   Measuring project control effectiveness/efficiency

In order to validate and to compare the warning signals that are produced by our inputs (the SPC approaches), we needed to assess their performance with respect to our definition of a state of statistical control. We therefore suggest the use of the *detection performance* and the *probability of overreactions* from [9] and again consider the state of statistical control as characterised by the MID $\hat{m}$/LOW $\hat{s}$ class of table 5.2. If a new project execution falls within this class, it is assumed to be influenced only by common cause sources of variation. If the project execution falls within a different class of table 5.2, or no class at all, special cause sources of variation are assumed to be in existence within this project. Special cause sources of variation should be signalled to the project manager by the control system. Following such a signal, the project manager might drill down the WBS of the project to get a more detailed look and plan actions to remove the special cause sources of variation from the project.

The detection performance measures how likely a fictitious project execution with special cause sources of variation generates a warning. Using 1,000 of these project executions in our simulation study, we measured the proportion that generates this warning. The higher the detection performance, the more effective the project control system is.

The probability of overreactions measures how often the project manager is warned by the control system even though only common cause sources of variation are present in the project. Following such a warning signal, the project manager might act to invest time and resources in getting a more

detailed look of the project's state of control. If such efforts are in vain, we consider them overreactions. The probability of overreactions is calculated from 1,000 fictitious project executions which contain only common cause sources of variation.

The detection performance and the probability of overreactions are closely linked. We therefore, propose the *area under the curve* (AUC) as a measure of efficiency. This curve is often named the Operating Curve (OC) and is widely applied in classification testing using machine learning [18]. Here, it is obtained from different detection performance and probability of overreactions couples and the area underneath it is calculated. A purely arbitrary classification method will have an AUC of 0.5 since its detection performance will always equal its probability of overreactions over a large set of project executions. The higher the AUC, the more efficient the control system operates.

The reader should note that the detection performance and probability of overreactions which are applied here are conceptually very close to respectively the true positive rate and false positive rate in binary classification testing. Moreover, the concepts are also closely linked to typical Type II and Type I errors associated with SPC procedures and their corresponding average run lengths (ARL). We have however, restricted our analysis and reporting to the calculation of AUC through the detection performance and probability of overreactions in order to correspond with the recently published study on statistical project control by [9].

5.4 Computational experiment, results and discussion

In this section, we describe the experiments that were conducted using the framework, depicted in figure 5.2. Section 5.4.1 presents some additional details on the setting that were used in the simulation model and the inputs to our framework. Subsequently, the outputs are discussed in section 5.4.2. Finally, section 5.4.3 provides a discussion.

5.4.1 Computational experiment

Here, we provide details on the computational experiment. First, some general details on the implementation of the simulation model and the conducted analyses will be presented. Second, the SPC procedures from prior literature [1, 4, 25, 48] that were applied to the simulated data of project executions, will be reviewed.

Details on the simulation experiment and the succeeding analysis

Our simulation is based on the model of [9], as we have discussed earlier. It is run in P2 Engine and produces a minimum of 1,000 fictitious project executions for all 6 classes defined by the $\hat{m}_{\mathrm{pooled}}$ and $\hat{s}_{\mathrm{pooled}}$ levels of table 5.2. For all 900 projects in the dataset we have generated 6,000 fictitious project executions with complete day-to-day data of project progress and EVM/ES observations. With an average of 300 days per project execution and 4 calculated EVM/ES metrics (SV, SPI, SV(t), SPI(t)), this resulted in 6.48 billion data points. To speed up the process of generating these data, the work was carried out on the STEVIN Supercomputer Infrastructure at Ghent University, funded by Ghent University, the Flemish Supercomputer Center (VSC), the Hercules Foundation and the Flemish Government - department EWI. The SPC procedures from prior literature, the normality tests, the proposed transformation and the interpretation of the warning signals were implemented in the statistical programming language R [36]and were run using an average of 90 nodes from the STEVIN Supercomputer Infrastructure.

Implementation of the SPC procedures from prior literature

The SPC procedures that were previously applied for project control are discussed in this section. In figure 5.3, we present an illustrative situation to show how the SPC procedures can be quantified using the detection performance, the probability of overreactions and the area under the curve. The reader should note that figure 5.3 is only included for illustrative purposes and that the results we show in section 5.4.2 are produced from a much larger number of such situations.

In addition to the separate discussion for all the SPC procedures in the following paragraphs, we discuss one common characteristic first. We denote the moment from which the SPC charts are dynamically applied to control the project progress as $\mathrm{PC}_{\mathrm{dynamic}}$. Here, PC is used to denote the percentage completion of a project during its execution (PC=EV/BAC). The moment $\mathrm{PC}_{\mathrm{dynamic}}$ was chosen differently in all of the published SPC procedures for project control. The procedures of Bauch and Chung (2001) and Leu and Lin (2008) deploy a reference of "similar" projects that were completed prior to the calculation of the control limits and therefore, $\mathrm{PC}_{\mathrm{dynamic}}$ could technically be chosen as 0% for a new project execution. Wang et al. (2006) and Aliverdi et al. (2013), however propose to calculate the control limits based on "historical" observations from the project that is being executed. The observations prior to $\mathrm{PC}_{\mathrm{dynamic}}$ are used to calculate the control limits and consequently, from the point $\mathrm{PC}_{\mathrm{dynamic}}$ on, these control limits are used dynamically to control the project progress. In their respec-

tive papers, Wang et al. (2006) propose $PC_{dynamic} = 100\%$, since they only present a post-hoc analysis of a project execution and Aliverdi et al. (2013) apply their SPC chart on 10 months of data, after collecting observations for 30 months ($PC_{dynamic} = 75\%$). In order to present results that are as general as possible, we will test all SPC procedures using equal values for $PC_{dynamic}$. $PC_{dynamic}$ will be chosen as a common multiple of 5% (e.g. 10%, 15%, 20%, ...). Moreover, we standardise the number of reporting periods after $PC_{dynamic}$ such that exactly (20-$\lceil PC_{dynamic}/5\% \rceil$) observations are made for each SPC procedure during the dynamic control process. Here, $\lceil x \rceil$ is used to denote the smallest following integer of x.

[4]

The CUSUM charts proposed by Bauch and Chung (2001) have been implemented to be used in an EVM/ES system. Although the authors present their approach whilst using a measure for "the actual owner expenditure", we conjectured that its use would be appropriate for SV, SPI, SV(t) and SPI(t) as well. According to the authors, the data should first undergo a cumulative normalization process after which the data are normalized with respect to time. The latter is done in our analysis by the generation of observations for EVM/ES at fixed percentage complete (PC = EV/BAC) points in the project. For the application of the Bauch and Chung (2001) procedure, the fictitious project execution lengths were therefore normalized using 20 PC points (PC = 5%, 10%, ..., 100%).

The control limits for the CUSUM chart were calculated for these PC points from observations in earlier "similar" projects. Prior to the calculation of the detection performance and the probability of overreactions, we will use 100 fictitious executions of the same project where only common cause sources of variation are present in order to build this historical reference of similar projects. Six control limits were then calculated to define upper and lower boundaries for the A, B and C zones. With these zones the 6 statistical project control tool (SPCT) rules of Bauch and Chung (2001) can then be applied.

Figure 5.3(a) displays the detection performance and probability of overreactions for an illustrative project simulation. The labels on these detection performance-probability of overreactions couples should be read as following: "y" includes SPCT rules 1 to y in the project control approach. For example, "3" resulted in a detection performance of about 42% for a probability of overreactions of about 21%. Moreover, it is shown for this illustrative simulation, that the inclusion of rules 4 up to 6 results in the same performance as if only the first three rules were used, and hence only a maximum of 42% can be obtained for the detection performance. The estimated area under the curve can then be calculated using a Riemann integration of the thick curve shown in figure 5.3(a). The thin curve is added

to all figures 5.3(a-d) for illustration purposes and represents a classification process based on a completely randomized process such as a coin-toss (AUC = 0.5).

[48]

Wang et al. (2006) proposed a Shewhart chart for statistical project control, where the control limits are calculated from early "historical" observations in the project. We chose the observations up to $PC_{dynamic}$ completion of the project as the historical reference of the project's performance. 20-$\lceil PC_{dynamic}/5\% \rceil$ observations, equally spaced along the time axis between $PC_{dynamic}$ and the project's end, are then used in the dynamic control process in order to calculate the detection performance and the probability of overreactions.

Since Wang et al. (2006) suggested the use of only one chart (with no additional rules), figure 5.3(b) has only one labelled detection performance and probability of overreactions couple. This results in the depicted 35% and 10% values for the illustrative situation in figure 5.3. The AUC can nevertheless be estimated using the additional theoretical couples 0-0 and 1-1 that exist for any classification method.

[25]

Leu and Lin (2008) implemented the I-mR chart with additional Western Electric rules for project control. As a reference base for the state of project control, they suggested the use of EVM/ES data of "similar" projects. Much like in our implementation of the Bauch and Chung (2001) approach, we calculated the boundaries for the A, B and C zones from 100 fictitious executions of the same project where only common cause sources of variation are present. Following the approach of the original paper, the data were not normalized to produce observations at fixed PC points, but rather, 20-$\lceil PC_{dynamic}/5\% \rceil$ observations, equally spaced along the time axis between $PC_{dynamic}$ and the project's end, are used to calculate the detection performance and the probability of overreactions.

Figure 5.3(c) depicts the detection performance-probability of overreactions couples for the combined use of the four Western Electric rules on an illustrative project simulation. The reader should note that we included the inspection of the moving range chart with the first Western Electric rule applied to the individuals chart (point "1"). Consequently, all of the depicted detection performance-probability of overreactions couples include the inspection of the moving range chart.

The reader should note that, for the illustrative situation in figure 5.3, the Leu and Lin (2008) approach results in the largest overall detection performance (for point "4"). Even though this large detection performance
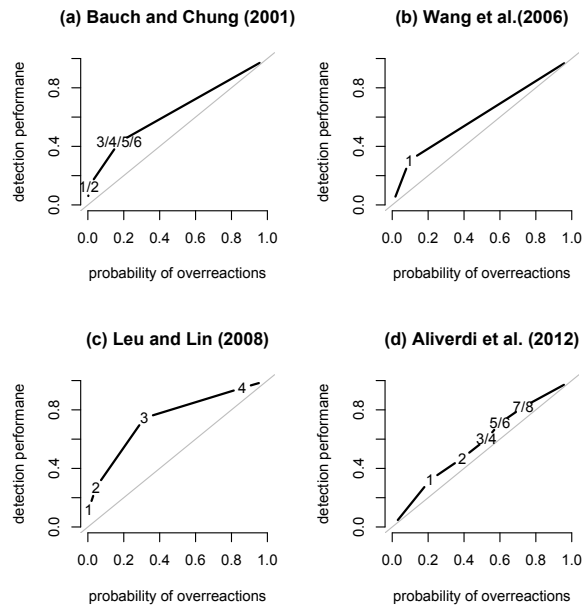
Figure 5.3: Illustration of the calculated probability of overreactions and the detection performance in an Operating Curve for the SPC methods defined by [4] (a), [48] (b), [25] (c) and [1] (d).

is coupled to a large probability of overreactions, the thick curve in figure 5.3(c) results in the highest recorded area under the curve for all approaches depicted in figure 5.3.

[1]

Aliverdi et al. (2013) suggest the use of the I-mR chart combined with Nelson's rules. The boundaries for the A, B and C zones were calculated from the "historical" reference of the EVM/ES observations made up to the point $PC_{dynamic}$. Again, 20-$\lceil PC_{dynamic}/5\% \rceil$ observations between the $PC_{dynamic}$ point and the project's end were used to calculate the detection performance and the probability of overreactions.

Figure 5.3(d) illustrates the performance of the eight control rules defined by Nelson. The inspection of the moving range chart was done simultaneously with the application of the first rule to the individuals chart. This means that, like in the Leu and Lin (2008) approach, all the depicted detection performance-probability of overreactions couples include the inspection of the moving range chart.

5.4.2 Results

The previous section illustrated how the three measures, detection performance, probability of overreactions and area under the curve, are calculated for each SPC approach on our large set of data, during our simulation experiments. The results of our simulation experiments are presented in this section. In order to generate results that can be compared to each other for all SPC procedures, $PC_{dynamic}$ is first kept fixed at 75% (i.e. the value derived from the study of Aliverdi et al. (2102)). With this setting, the assumption of normality is checked for all data in section 5.4.2, and the measured area under the curve for the four SPC procedures is presented in section 5.4.2. Finally, in section 5.4.2, the setting $PC_{dynamic}$ is varied in order to test the effect of the point in the project from which the execution is dynamically monitored.

Normality tests

In our framework, proposed to test the efficiency of the SPC approaches, we also test for the assumption of normality of the project progress data that are used in the control charts.

Rejection rates per normality test
We applied three hypothesis tests (KS, χ^2, SK) to check the normality of all of the data generated in our framework. The reader should note that prior to these tests, the data were normalized using the transformations that were proposed in the discussed papers on SPC for project control.

Table 5.3 shows the mean and the standard deviation of the rejection rates for all hypothesis tests. A data point was rejected in our experiments if its p-value was lower than 5%. In other words, the null hypothesis (normality) was rejected for these points at a 5% significance level. Table 5.3 clearly shows that a comparison of the SPC approaches based on different tests is unlikely to be fair. The normality assumption could not be withheld for a single data point when the skewness-kurtosis test was used. Although this test was successfully used by Wang et al. (2006) on their data, it can serve no purpose for our experiment. Therefore, table 5.3 primarily shows that the results of a normality test should be interpreted with care. We will proceed our analysis of the SPC approaches using Pearson's χ^2 goodness-of-fit test for all normality checks, as this gives us the least rejections (55%).

Rejection rates per transformation method
Table 5.4 presents the rejection rates for Pearson's χ^2 test for the transfor-

| | **Hypothesis test** | | |
	Kolmogorov-Smirnov	Pearson's χ^2	Skewness-Kurtosis
Mean	77%	55%	100%
Standard deviation	1.1%	5.2%	0.0%

Table 5.3: Rejection rates per hypothesis test

mations proposed in literature. Bauch and Chung (2001) proposed a normalization for both the time and the observed cumulative values. Wang et al. (2006) did not propose a transformation and applied the normality test directly to historical samples. Leu and Lin (2008) applied a normality test to a reference of similar project executions, where the data is used either directly (x), after taking the logarithm ($\ln x$) or after taking the logarithm of its inverse ($\ln x^{-1}$). Although [26, 27] suggested the use of the logarithm only for the SPI(t), we also implemented its use for the SPI, the SV and the SV(t). Given that the variance-based measures (SV and SV(t)) can take on negative values, the absolute value for these observations are used in the transformation. The Box-Cox transformation and Johnson transformation, proposed by Aliverdi et al. (2013), were performed simultaneously in their research. The authors proposed an optimization procedure to find the best transformation. We mimicked this optimization method in our tests, using the parameters provided in the original article and earlier research on the Johnson system [41].

Table 5.4 shows that the Bauch and Chung (2001) and Wang et al. (2006) approaches result in the rejection of the normality assumption for nearly all samples (99% and 97%) generated in the simulation experiments. Clearly, the generated EVM/ES data are not suited to be used in the SPC procedures of Bauch and Chung (2001) and Wang et al. (2006). Moreover, table 5.4 shows that the Aliverdi et al. (2013) approach results in significantly less rejections of the null hypothesis than the other transformations (31%). The transformations involved in the Aliverdi et al. (2013) approach are much more complex then those that are proposed by the other authors and therefore, the normality assumption is less often rejected. A noteworthy exception is the application of the logarithmic and reverse logarithmic transformation for SPI(t), proposed by Leu and Lin (2008). Here, the rejection rates (30% and 32%) are very much comparable to those observed for the Aliverdi et al. (2013) approach. Overall, there seems to be little difference in the rejection rates for the SPI, the SV and the SV(t). The SPI(t) however, shows an average rejection rate that is clearly lower.

Area under the curve

The efficiency of a project control approach is expressed through its area under the curve. We will first review the AUC values for all SPC approaches. Second, we will explore the specific efficiencies observed for the different simulated classes of project executions.

Project control efficiency per SPC approach
Figure 5.4 presents the measured AUC values for all SPC approaches. A

Performance measure	Bauch and Chung (2001)	Wang et al. (2006) x	$\ln x$	$\ln x^{-1}$	Aliverdi et al. (2013)	mean (performance measure)
SPI	100%	98%	82%	97%	92%	34%	84%
SV	100%	97%	94%	97%	97%	23%	85%
SPI(t)	97%	96%	78%	30%	32%	38%	62%
SV(t)	100%	97%	100%	98%	98%	31%	87%
mean (SPC approach)	99%	97%	88%	80%	80%	31%	

Table 5.4: Rejection rates per transformation method using Pearson's χ^2 test
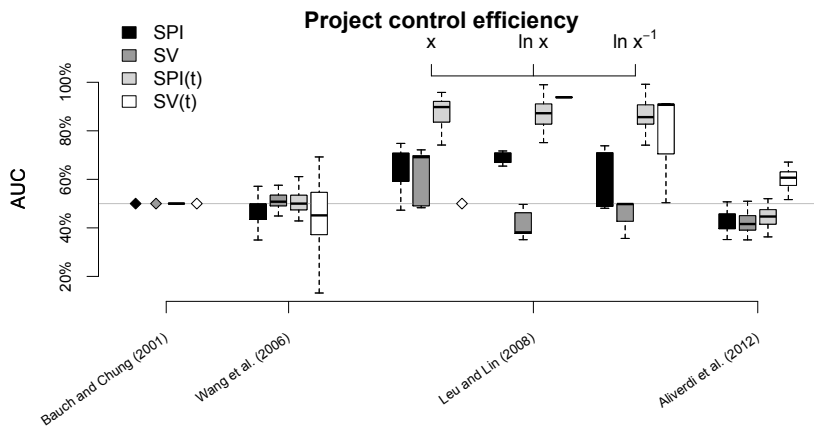
Figure 5.4: Measured AUC values for the SPC approaches from prior literature on project control

thin grey line is again added to represent a completely randomized classification process (AUC=0.5). Coloured diamonds were added to figure 5.4 to represent missing values, i.e. those situations for which the normality assumption was rejected in 100% of the cases. Figure 5.4 clearly shows how the Leu and Lin (2008) approach outperforms all others. Leu and Lin (2008) clearly presented a more efficient project control system, which can be seen from the higher AUC values when compared to the other SPC approaches. The latter do not outperform a randomized classification process in many of the simulated project executions. In particular the SPI(t) provides the best option for a project manager to differentiate between common cause and special cause sources of variation. The average AUC for the SPI(t) is the highest when the observations are used directly (x) on the I-mR control charts. However, given the normality rejection rates of table 5.4, the application of the logarithm (or its inverse) to the SPI(t) values in the Leu and Lin (2008) approach seems to present the most favourable option to be used in SPC charts for project control.

Project control efficiency for different special cause sources of variation

Table 5.5 shows the measured AUC values for the four SPC approaches for different special cause sources of variation. The table consists of 4 smaller tables that are designed after table 5.2, and shows the AUC for the EVM/ES schedule performance metric that performs best for each SPC approach. The experiments were performed in a fashion similar to our illustrative discussion of the state of statistical control in section 5.3.2. The state of

Bauch and Chung (2001)		
	$\hat{s}$	
$\hat{m}$	LOW	HIGH
LOW	50%	50%
MID		50%
HIGH	50%	50%

Wang et al. (2006)		
	$\hat{s}$	
$\hat{m}$	LOW	HIGH
LOW	54%	52%
MID		65%
HIGH	51%	73%

Leu and Lin (2008)		
	$\hat{s}$	
$\hat{m}$	LOW	HIGH
LOW	78%	97%
MID		98%
HIGH	96%	99%

Aliverdi et al. (2013)		
	$\hat{s}$	
$\hat{m}$	LOW	HIGH
LOW	63%	63%
MID		61%
HIGH	60%	64%

Table 5.5: AUC values for different sources of variation

statistical control is characterized by the MID $\hat{m}$/LOW $\hat{s}$ class. For project executions in this class it is conjectured that only common cause sources of variation are present. The project executions that contains only common cause sources of variation contribute to the calculation of the probability of overreactions. The other $\hat{m}$-$\hat{s}$ combinations are said to contain special cause sources of variation with respect to the state of project control defined by the class in grey. Project executions that fall in a different $\hat{m}$/$\hat{s}$ class contribute to the detection performance and consequently, the area under the curve for that cell. As a result, no AUC value can be calculated for the mid MID $\hat{m}$/LOW $\hat{s}$ class (in grey) since there can only be overreactions for project executions in this class.

Table 5.5 confirms the findings of figure 5.4. The Leu and Lin (2008) approaches outperforms all of the other SPC approaches. In addition, table 5.5 also provides insights in the project control efficiency if the project execution contains different special cause sources of variation. If the activity durations of the project are characterized by LOW $\hat{m}$ and LOW $\hat{s}$, the project control process is not efficient (low AUC values). In general, we can expect the project control process to become more efficient, for increasing levels of $\hat{m}$ and $\hat{s}$. The AUC values increase for increasing $\hat{m}$ and $\hat{s}$ levels.

Dynamic project control

Figure 5.5 presents the effect of the $PC_{dynamic}$ point on the AUC of the SPC approaches. The Bauch and Chung (2001) and Wang et al. (2006) approaches show little effect of the $PC_{dynamic}$ and their AUC values remain
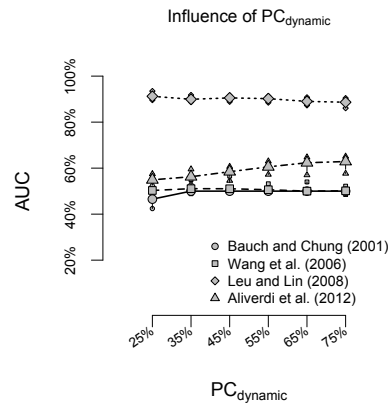
Figure 5.5: Influence of the point $PC_{dynamic}$ from which the project is dynamically controlled using the SPC approaches

low for all the tested $PC_{dynamic}$ settings. For the other two SPC approaches, we observe two distinct phenomena.

An increase of the AUC values can be observed for the Aliverdi et al. (2013) approach, for increasing values of $PC_{dynamic}$. This means that the project control efficiency improves when more observations are included in the historical record. The control limits, derived from this historical record, are then more useful to distinguish between common cause and special cause sources of variation. For the Leu and Lin (2008) approach, an opposite effect can be observed. The AUC values decrease slightly for increasing $PC_{dynamic}$. The reader should note that Leu and Lin (2008) deploy a reference of similar projects, that have all been finished, to calculate the control limits. Consequently, the control limits are not affected by the value for $PC_{dynamic}$. Only the number of observations that are used during the dynamic control process (20-$\lceil PC_{dynamic}/5\% \rceil$) is influenced by $PC_{dynamic}$. In our tests, the number of observations decrease for increasing $PC_{dynamic}$, which affects the AUC negatively.

5.4.3 Discussion

The results presented in this paper should be interpreted with care as they are the results of the presented framework. Our framework is characterised by two key elements, namely an exclusive focus on the schedule performance of the project and a top-down project control approach. Below, we shortly elaborate on these two elements.

The first restriction of our framework is caused by our exclusive interest in the schedule performance of the project. Although numerous researchers have expressed their interest in controlling and estimating the total cost of a project [1, 5, 6, 20, 21, 32, 48], we have kept a restricted focus on controlling the schedule.

The second restriction of our framework is a direct consequence of our definition of a state of schedule control in section 5.3.3. We conjectured that the real schedule performance is shown at the activity level of the project in the relative durations of the individual activities. Since period-to-period monitoring at the activity level is believed to be a disruptive task, we try to approximate this schedule control process by making use of EVM/ES measures that are calculated at the project level. The use of an EVM/ES system in combination with control limits calculated for its performance measures is thereby reduced to a classification process. We have calculated and presented all finding in this paper based on that paradigm. This approach to EVM/ES control was previously discussed as top-down project control [9, 44, 46].

In addition, it should be mentioned that a detailed comparison of the top-down approach used in this paper and alternative project control approaches available in the literature is not within the scope of this paper. A comparison with CPM, PERT and its novel extension to PERT21 [42], a bottom-up approach using schedule risk analysis [14] or the critical chain methodology [19] is left as promising future research avenues.

Finally, the reader should note that the presentation of our research is restricted in order to show the results that were obtained using our framework, rather than to illustrate the mechanism of the individual elements. This is motivated by the availability of such illustrations in other research articles. An illustration of the mechanisms of the proposed simulation model to generate project progress situations and the calculation of the detection performance, the probability of overreactions and the AUC can be found in [9]. In addition, [8] provide an illustration of how the definition of a state of schedule control can be applied to a new record of a project that is executed in real-life and how an SPC approach might be implemented. Moreover, we refer to the cited references for the different SPC approaches that are discussed in this paper, for illustrations on their details. Finally, section 5.4.1 and in particular figure 5.3, provide the reader with an illustration of how the SPC approaches translate to the measured detection performance and probability of overreactions and how the AUC can be calculated.

5.5 Conclusion

This research article has been set up to test and compare the SPC approaches that were previously published in literature. From a detailed discussion of these approaches we recognized the apparent need for a unified framework in which these SPC techniques could be applied to project control. The drivers for this need have been identified in this paper as the lack of a formal definition for a state of control, the unavailability of a benchmark dataset, the absence of measures to quantify the SPC performance and the lack of consensus on how to overcome and test the normality assumption.

The framework, that we propose in this paper, addresses these issues in the following ways. The classification that was developed from the empirical records of [3] allows us to define a state of control. Since this classification was produced from the distribution of relative activity durations in real-life projects, we conjecture that it accurately expresses the "voice of the process" and that it is suitable to define a state of control for project management. The concern regarding the unavailability of a project benchmark dataset has been alleviated by our inclusion of the RanGen dataset and the proposed use of a validated project progress simulation model. Using the resulting dataset of simulated project progress situations, we were able to compare the transformations and statistical tests, that were proposed in literature, with respect to the normality of the input values for SPC approaches. Moreover, given the proposal of two quantitative measures for project control, we were able to measure and compare the performance of four SPC approaches from literature.

We have performed a comprehensive experiment using simulated project progress situations in the proposed framework. The results for all four SPC approaches from prior literature have been presented. The SPC approaches have been compared based on their performance with respect to the normality assumption and the project control efficiency.

This study should be useful to both academics and practitioners in the field of project management. The former have been provided with a framework in which they can test newly developed control procedures and which they can extent to meet additional requirements. The latter are presented with a clear overview of the research on SPC approaches for project management using EVM/ES. In addition to a discussion of the major contributions of these SPC approaches, a comprehensive comparison of their performance is given. The results show how the [25] approach is able to effectively distinguish between common cause and special cause sources of variation. The [25] approach derives its control limits from a reference set of similar projects. The SPC control charts, using these control limits, are applicable in practice and can therefore provide a valuable alternative to the decision

making based on intuition and experience that characterizes the field of EVM/ES practice today.

References

References

[1] Aliverdi, R., Moslemi Naeni, L., Salehipour, A., 2013. Monitoring project duration and cost in a construction project by applying statistical quality control charts. International Journal of Project Management 31 (3), 411–423.

[2] Anbari, F., 2003. Earned value project management method and extensions. Project Management Journal 34(4), 12–23.

[3] Batselier, J., Vanhoucke, M., 2014. Construction and evaluation framework for a real-life project database. International Journal of Project Management (0), –.
URL http://www.sciencedirect.com/science/article/pii/S0263786314001410

[4] Bauch, G. T., Chung, C. A., 2001. A statistical project control tool for engineering managers. Project Management Journal 32, 37–44.

[5] Bee Hua, G., 2008. The state of applications of quantitative analysis techniques to construction economics and management (1983 to 2006). Construction Management and Economics 26 (5), 485–497.

[6] Cheng, M.-Y., Peng, H.-S., Wu, Y.-W., Chen, T.-L., 2010. Estimate at completion for construction projects using evolutionary support vector machine inference model. Automation in Construction 19 (5), 619–629.

[7] Cioffi, D., 2006. Completing projects according to plans: An earned-value improvement index. Journal of the Operational Research Society 57, 290–295.

[8] Colin, J., Vanhoucke, M., 2014. An empirical perspective on activity durations for project management simulation studies. Journal of Construction Engineering and Management, –.

[9] Colin, J., Vanhoucke, M., 2014. Setting tolerance limits for statistical project control using earned value management. Omega The International Journal of Management Science 49, 107–122.

[10] Company, W. E., Small, B. B., 1958. Statistical Quality Control Handbook. AT&T.

[11] D'Agostino, R., Pearson, E. S., 1973. Tests for departure from normality. empirical results for the distributions of b2 and? b1. Biometrika 60 (3), 613–622.

[12] Demeulemeester, E., Vanhoucke, M., Herroelen, W., 2003. Rangen: A random network generator for activity-on-the-node networks. Journal of Scheduling 6, 17–38.

[13] Elshaer, R., 2013. Impact of sensitivity information on the prediction of project's duration using earned schedule method. International Journal of Project Management 31, 579–588.

[14] Fazar, W., 1959. Program evaluation and review technique. The American Statistician 13, 10–10.

[15] Fleming, Q., Koppelman, J., 2010. Earned value project management. 3rd Edition, 3rd Edition. Project Management Institute, Newton Square, Pennsylvania.

[16] Hahn, E., 2008. Mixture densities for project management activity times: A robust approach to PERT. European Journal of Operational Research 188, 450–459.

[17] Hartmann, S., Briskorn, D., 2010. A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research 207, 1–15.

[18] Hastie, T., Tibshirani, R., Friedman, J., Hastie, T., Friedman, J., Tibshirani, R., 2009. The elements of statistical learning. Vol. 2. Springer.

[19] Herroelen, W., Leus, R., 2001. On the merits and pitfalls of critical chain scheduling. Journal of Operations Management 19, 559–577.

[20] Hunter, H., Fitzgerald, R., Barlow, D., 2014. Improved cost monitoring and control through the earned value management system. Acta Astronautica 93, 497–500.

[21] Ji, S.-H., Park, M., Lee, H.-S., 2010. Data preprocessing–based parametric cost model for building projects: Case studies of korean construction projects. Journal of Construction Engineering and Management 136 (8), 844–853.

[22] Jones, L. A., 2002. The statistical design of ewma control charts with estimated parameters. Journal of Quality Technology 34 (3), 277–288.

[23] Kelley, J., 1961. Critical path planning and scheduling: Mathematical basis. Operations Research 9, 296–320.

[24] Khamooshi, H., 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.

[25] Leu, S. S., Lin, Y. C., 2008. Project performance evaluation based on statistical process control techniques. Journal of Construction Engineering and Management 134, 813–819.

[26] Lipke, W., 2002. A study of the normality of earned value management indicators. The Measurable News 4, 1,6,7,12–14,16.

[27] Lipke, W., Vaughn, J., 2000. Statistical process control meets earned value. CrossTalk: The Journal of Defense Software Engineering June, 16–20,28–29.

[28] Lipke, W., Zwikael, O., Henderson, K., Anbari, F., 2009. Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes. International Journal of Project Management 27, 400–407.

[29] Malcolm, D., Roseboom, J., Clark, C., Fazar, W., 1959. Application of a technique for a research and development program evaluation. Operations Research, 646–669.

[30] Massey Jr, F. J., 1951. The kolmogorov-smirnov test for goodness of fit. Journal of the American statistical Association 46 (253), 68–78.

[31] Montgomery, D. C., 2012. Statistical Quality Control: A Modern Introduction, 7th Edition International Student Version. Wiley.

[32] Narbaev, T., De Marco, A., 2014. An earned schedule-based regression model to improve cost estimate at completion. International Journal of Project Management 32 (6), 1007–1018.

[33] Nelson, L. S., 1985. Interpreting shewhart x control charts. Journal of Quality Technology 17 (2), 114–116.

[34] Page, E., 1954. Continuous inspection schemes. Biometrika, 100–115.

[35] Plackett, R. L., 1983. Karl pearson and the chi-squared test. International Statistical Review/Revue Internationale de Statistique, 59–72.

[36] R Core Team, 2013. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org

[37] ReVelle, J. B., 2004. Quality Essentials: A reference guide from A to Z. ASQ Quality Press.

[38] Rozenes, S., Vitner, G., Spraggett, S., 2006. Project control: literature review. Project Management Journal 37, 5–14.

[39] Russell, J. S., Jaselskis, E. J., Lawrence, S. P., 1997. Continuous assessment of project performance. Journal of construction engineering and management 123 (1), 64–71.

[40] Shewhart, W. A., 1931. Economic control of quality of manufactured product. Vol. 509. ASQ Quality Press.

[41] Slifker, J. F., Shapiro, S. S., 1980. The johnson system: selection and parameter estimation. Technometrics 22 (2), 239–246.

[42] Trietsch, D., Baker, K. R., 2012. Pert 21: Fitting PERT/CPM for use in the 21st century. International Journal of Project Management 30 (4), 490–502.

[43] Trietsch, D., Mazmanyan, L., Govergyan, L., Baker, K. R., 2012. Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation. European Journal of Operational Research, 386–396.

[44] Vanhoucke, M., 2010. Measuring Time - Improving Project Performance using Earned Value Management. Vol. 136 of International Series in Operations Research and Management Science. Springer.

[45] Vanhoucke, M., 2010. Using activity sensitivity and network topology information to monitor project time performance. Omega The International Journal of Management Science 38, 359–370.

[46] Vanhoucke, M., 2011. On the dynamic use of project performance and schedule risk information during project tracking. Omega The International Journal of Management Science 39, 416–426.

[47] Vanhoucke, M., 2014. Integrated Project Management and Control: First come the theory, then the practice. Management for Professionals. Springer.

[48] Wang, Q., Jiang, N., Gou, L., Che, M., Zhang, R., 2006. Practical experiences of cost/schedule measure through earned value management and statistical process control. Lecture Notes in Computer Science 3966, 348–354.

[49] Wheeler, D. J., 1995. Advanced topics in statistical process control. Vol. 470. SPC press Knoxville, TN.

6

A multivariate approach for top-down project control using earned value management
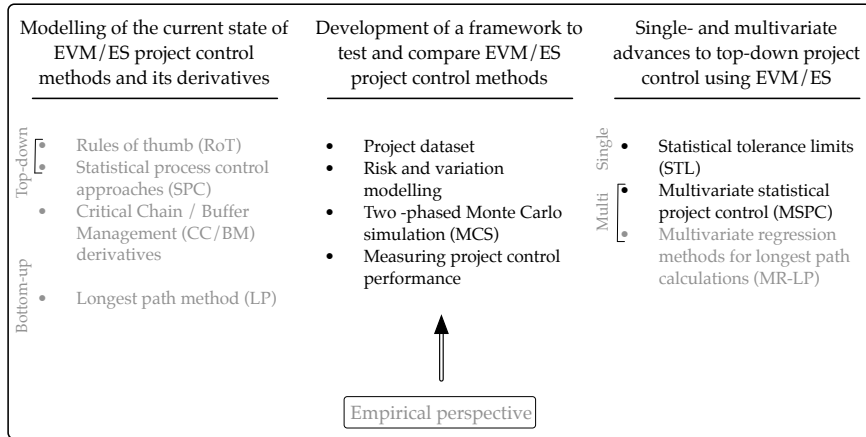
Figure 6.1: Schematic overview of the elements, of the larger body of research in this book, that are discussed in chapter 6.

Abstract

The application of a top-down project schedule control process requires an advanced decision support system (DSS), since aggregated performance measures to control a project's makespan are likely to obscure details on the activity level schedule progress. Earned value management/earned schedule (EVM/ES) is typically applied in a DSS for top-down project schedule control. However, traditional models do not correctly account for the multivariate nature of the EVM/ES measurement system. We therefore propose a multivariate model for EVM/ES, which implements a principal component analysis (PCA) on a simulated schedule control reference. During project progress, the real EVM/ES observations can then be projected onto these principal components. This allows for two new multivariate schedule control metrics (T^2 and SPE) to be calculated, which can be dynamically monitored on project control charts. Using a computational experiment, we show that these multivariate schedule control metrics lead to enhanced decision making in top-down schedule control, when compared to traditional univariate EVM/ES models.

6.1   Introduction

The focus of this research is on the development of a decision support system (DSS) for project schedule control. Although cost and scope were also

identified as important dimensions of project management success [1], we will only focus on the control of the schedule performance of a project. If the project manager has a detailed overview of the progress of all the individual activities, at any given time in the project, a decision support framework would be a relatively straightforward one. However, it has been argued in literature that the activity level monitoring of all activities would be a cumbersome and often disruptive task for the project manager [2, 3]. We therefore focus on a DSS that is implemented on a higher level of the work breakdown structure (WBS) and which can be situated within the risk monitoring and control phase of a classical project risk management (PRM, [4]) process. If the project is monitored at these high levels of the WBS, the need for an advanced DSS becomes apparent. In this case, only aggregated measures of the project schedule performance are monitored regularly and details on the underlying activity level performance might be obscured. The DSS that we present here, aims to help the project manager in making the decision whether he/she should invest time and effort to obtain a more detailed (activity level) view of the project schedule performance, in order to take further actions to bring the project back on track. This drill-down of the WBS in order to obtain a more detailed activity look is known as top-down control [3, 5]. In our research, we focus on the development of such a DSS and we will not incorporate possible actions that can be taken to bring the project back on track. Earned value management/Earned schedule (EVM/ES) is often implemented in a DSS for top-down project schedule control, and we will discuss its use in this paper due to its popularity in practice. The subject of EVM/ES has received a lot of attention from academics in recent years, as can be seen from the numerous publications on case studies [6, 7], consolidated research experiments [3, 5] and possible extensions [2, 8–10].

We will briefly introduce the EVM/ES methodology for project schedule control using the three fundamental components of any DSS architecture, i.e. the database, the model and the user interface [11].

Database The data that are used in an EVM/ES system are aggregated performance metrics calculated at the project level of the WBS from the earned value (EV) and the planned value (PV). These metrics compare the preliminary planning variable (PV) with the actual performance (EV) in order to asses the schedule progress of the project and to indicate the direction of changes with respect to the baseline schedule. The EV is the budgeted value of the work that is performed in the project, whereas the PV is the budgeted value of the work that was scheduled to be performed [8]. The schedule variance (SV=EV-PV) and schedule performance index (SPI=EV/PV) can be directly calculated from the EV and PV, which are expressed in monetary units. The earned schedule (ES) translates the EV

into time units and expresses the time since the beginning of the project
at which the current EV should have been accrued, according to the base-
line schedule. When compared to the actual time (AT) since the begin-
ning of the project, the ES gives rise to the schedule variance using earned
schedule (SV(t)) and the schedule performance index using earned sched-
ule (SPI(t)).

Model   The model to interpret the schedule control metrics, provided by
the EVM/ES system, has been subject to some debate in the literature.
Straightforward control limits for these metrics where proposed [12] as
well as implementations of statical process control charts [13–15] and sta-
tistical tolerance limits [5]. However, these methods do not account for
the project-specific dynamics of the EVM/ES metrics [5] or fail to accu-
rately incorporate the multivariate nature of the EVM/ES system. As a
consequence, the use of EVM/ES in practice is typically characterised by
decision making based on experience, arbitrary rules-of-thumb and anec-
dotal evidence. In order to aid the project manager in his/her decision
making process, the EVM/ES system is often implemented in software.
Commercial products have emerged on the market (for example, Decision
Edge [16] to go along with Microsoft Project, or the stand-alone applica-
tions ProTrack [17] and P2 Engine [18]) to present the project manager
with the schedule performance metrics throughout the execution of the
project.

User interface   The user interface is mostly comprised of a set of charts
that depict the time-series of the different schedule performance metrics.
The interpretation of these charts can be impeded by the shear overload of
observations presented to the project manager, the redundancy in these ob-
servations or the noise that is inherent to the EVM/ES system. Ideally, the
model that is applied to the EVM/ES data should handle these character-
istics, in order for the user interface to be effective in driving the decision
process.

In this paper, we present a multivariate model to be applied in a DSS us-
ing EVM/ES for top-down project schedule control, that correctly accounts
for the characteristics that were briefly introduced above. We will elabo-
rate on these characteristics in section 6.2. We account for these charac-
teristics by analysing the correlation structure of EVM/ES measurements
over the course of the execution of the project in a simulated schedule con-
trol reference. From this correlation structure, we calculate latent variables
that reflect the underlying dynamics of the EVM/ES measurement system.
These latent variables are identified as principal components, calculated by
a principal component analysis (PCA). We will introduce PCA in section
6.3. The principal components represent the basis for a new coordinate

system onto which the EVM/ES observations during the project execution can be projected. Using this projection, we will show in section 6.4 how two new performance metrics can be calculated, which accurately incorporate the multivariate nature of the EVM/ES system. These new multivariate control metrics can be presented on project control charts in order to steer the decision process in top-down project schedule control. We will demonstrate the enhanced decision making that results from these two new metrics using a large computational experiment, outlined in section 6.5. Section 6.6 provides the results and overall conclusions are drawn in section 6.7. Four appendices to this paper are available online. They can be freely accessed on the `statistical project control` research page on `www.projectmanagement.ugent.be`. In appendix A.2, a more detailed overview is given for the mathematics involved in calculating a PCA. Appendix A.3 contains a geometrical interpretation for a PCA. In appendix A.4, we introduces the theoretical distributions that are used in the literature on batch process control to derive control limits for the T^2 and the SPE metrics. Finally, in appendix A.5 a numerical example for a project schedule control process using the multivariate T^2 and SPE metrics is presented.

6.2   Multivariate nature of schedule control

In this section, we first present a formal characterisation of the top-down project schedule control process. This characterisation will be used in this paper to quantify the performance of the decision process that follows from a DSS using EVM/ES for top-down project schedule control. Second, we will discuss the multivariate nature of the EVM/ES system and we will show which aspects need to be considered in order to improve the top-down schedule control process. Finally, we will briefly introduce the body of literature that has been developed to handle these aspects in a batch process control context.

In order to present a formal characterisation of the top-down schedule control process, let us consider a vector x of EVM/ES measurements along the lifetime of the project as an observation for a multivariate random variable X. This random variable X represents the schedule performance measurements of a project as observed at the top WBS level (SV, SPI, SV(t), SPI(t)). In EVM/ES, X is a function of the underlying activity level performance, which can be expressed as the multivariate random variable D, containing the real durations for all activities in the project. From a computational perspective, it is not opportune to derive a closed-form analytical expression for the function $X = f(D)$ for projects with a large number of activities. Rather, we wish to deduce the activity level performance d from the vector

of measurements x during the execution of the project. In this research, we do not intend to calculate the activity level schedule performance explicitly, but we would like to infer *a state of schedule control*, i.e. whether the activity durations conform a pre-defined state of control. This implicit inference process will be used to quantify the performance of the decision making that follow from our multivariate model for a DSS using EVM/ES in section 6.5. The end user of our DSS (i.e. the project manager) will be presented with control charts and corresponding control limits. If the control chart produces a signal (a control limit that is exceeded), the project manager is provided with an indication that the underlying activity durations do not conform with the pre-defined state of control. The project manager will then likely invest time and resources to drill down the WBS of the project to find the activities that cause this departure from the pre-defined state of control. If, at the activity level of the project, the departure from the pre-defined state of control can not be confirmed, the efforts spent by the project manager were in vain. The signal, produced by the DSS is then said to be false. If however, the departure can be confirmed at the activity level, the signal can be said to be true. In section 6.5, we will provide more details on how these true and false signals are used to quantify the performance of the decision making process.

With respect to the multivariate nature of EVM/ES observations, and improved DSS using EVM/ES should consider the following aspects:

- **Data overload:** The project manager using an EVM/ES system is provided with an abundance of data over the lifetime of a project. The choice of which of the performance metrics (SV, SV(t), SPI or SPI(t)) is likely to result in an as good as possible inference of activity level schedule performance, is a question that is often answered based on subjective arguments.

- **Redundancy:** Even in the case that the vector x is composed out of observations for a single EVM/ES performance metric along the lifetime of the project, x can still be subject to redundancy. Redundancy, or collinearity, is a problem that is found when different observed variables are influenced by a common factor. Redundancy should be addressed when inference is made from a multivariate variable [19]. If we suppose that SPI(t) is measured along the lifetime of the project, it is likely that some of the observations for SPI(t) are influenced by the same couple of activities, hence collinearity will exist between these variables.

- **Noise:** Multivariate systems can be subject to noise. It has been shown that SPI behaves unreliable towards the end of late projects, since for any execution of a project that experiences delays, SPI will equal 1 at the end of the project [2]. The use of SPI for duration forecasts in the project is therefore not recommended. One could ar-

gue that SPI is erroneous, and should therefore not be considered for schedule control. However, in early stages of the project, the information contained in SPI can be valuable for schedule control. In late projects, even the rate at which SPI converges towards 1 and the timing of the point at which the increase towards 1 takes place can provide the project manager with effective information, with respect to the underlying activity level performance. What is regarded by one as noise in the system, could be usefully employed by another. We therefore reason that a complex multivariate system such as EVM/ES could benefit from an automatic mechanism to identify and handle noise.

The aspects discussed here are common to many multivariate measurement systems. They have previously been addressed for the control of batch processes by MacGregor [20], to whom we refer for an illustration on how a principal component analysis overcomes these problems. Multivariate projection methods have been used to overcome the problems associated with the multivariate nature of the measured quality of a system since the first application of Hotelling's [21] multivariate t-test measure, in combination with PCA. Ever since, strong interest has been shown for this field of study, which resulted in related multivariate projection methods. The reader is referred to Bersimis et al. (2006) [22] for a recent overview and a comparison of these techniques. For the scope of this paper we will refrain from these more recent procedures. We believe that for this introduction on how a multivariate projection method can be used in a DSS using EVM/ES for top-down project schedule control, PCA is an appropriate choice. PCA is algorithmically easier, and provides some insight in how a new basis for comparison of project schedule performance is chosen, based on a reference set of project executions.

6.3 Principal component analysis of EVM/ES schedule performance metrics

In this section, we will first introduce the basics of a principal component analysis and how it can be performed. Second, we will discuss how the schedule control reference matrix should be composed.

6.3.1 Calculating principal components

In the very early years of the 20th century PCA was developed by Pearson [23]. Ever since, PCA has been a popular procedure to reduce the dimensionality of a variable space. A full coverage on the basics of linear algebra,

fundamental to PCA, lies outside the scope of this paper and the reader is referred to the recent book of Jolliffe [24].

Instead, we will give a brief overview of the matrix calculus that is required for a PCA.

PCA assumes that the true rank of a matrix of observations X is less than the number of observations which are made. Consequently, it conjectures that the observations can be projected onto a new set of coordinate axes, thereby removing redundancy and noise from the system. The PCA decomposition method first calculates the principal components of the observation space, i.e. the directions that will make up the new coordinate axes. These principal components are often also named the latent variables, since they represent the underlying (unobservable) factors really influencing the system dynamics. Consider a $(n \times P)$ matrix X that is a collection of n measurements for a P-variate randomly distributed variable $\mathbf{x}$. The first principal component of $\mathbf{x}$ is defined as the vector of coefficients $\mathbf{p}_1$ for which the linear combination $t_1 = \mathbf{x}\mathbf{p}_1$ captures as much as possible of the variance contained in X, subject to $|\mathbf{p}_1| = 1$. The second principal component is then the vector of coefficients $\mathbf{p}_2$ for which the linear combination $t_2 = \mathbf{x}\mathbf{p}_2$ contains as much of the variance from X that is not captured within t_1. Additional principal components up to P are similarly defined.

In practice, a PCA is always performed using the computationally efficient singular value decomposition (SVD, [25]) of X ($X = \text{ULA}^T$). A PCA decomposition of the matrix X can be written as:

$$X = \text{TP}^T \tag{6.1}$$

Correspondingly, the standard deviation of the ith principal component:

$$s_{\mathbf{t_i}} = \sqrt{\frac{l_i^2}{n-1}} \tag{6.2}$$

can be obtained from the singular values on the diagonal of L, since these are equal to the square roots of the eigenvalues ($\sqrt{l_i^2}, \forall i \in \{1, \ldots, P\}$) of $X^T X$.

The score-loading nomenclature is very common in PCA literature and is therefore adopted here. The matrix of loadings $\text{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \ldots & \mathbf{p}_P \end{bmatrix}$ can be seen as a $(1 \times P)$ collection of $(P \times 1)$ vectors of coefficients for the linear combination $t_i = \mathbf{x}\mathbf{p}_i$ that defines the ith principal component. A related term "matrix of rotations" expresses the geometrical interpretation of the loadings, as they represent a new coordinate space onto which $\mathbf{x}$ is projected. The $(n \times P)$ matrix of scores T can then be interpreted as the values in the new coordinate space for the collection of n measurements of $\mathbf{x}$.

With respect to the ability of a PCA to remove redundancy and noise from the system, we need to discuss the relative importance of the different principal components, as expressed by their standard deviations. We will now explore this further, while discussing whether or not all principal components should be retained in the analysis.

If all of the P principal components are retained, for a further analysis of the data, an observation for the P-variate vector of observations $\mathbf{x}$ can be reconstructed from its $(1 \times P)$ vector of scores $\mathbf{t}$.

$$\mathbf{x} = \mathbf{t}\mathrm{P}^T = \sum_{i=1}^{P} t_i \mathbf{p}_i \qquad (6.3)$$

However, since PCA is used to find a solution to the problems associated with the multivariate nature of project data, it will try to reduce the dimensionality of the problem in a structured manner, without losing valuable information. We assume that an integer $0 < k \leq P$ exists such that the last $P - k$ principal components do not represent valuable information for our system. By definition, each principal component will only explain a very small part of the original variation contained in X. For now we state that if only k principal components are retained, the original observation for the P-variate vector of observations $\mathbf{x}$ can be estimated as $\hat{\mathbf{x}}$ from its $(1 \times k)$ vector of scores $\mathbf{t}$.

$$\hat{\mathbf{x}} = \mathbf{t}\mathrm{P}^T = \sum_{i=1}^{k} t_i \mathbf{p}_i \qquad (6.4)$$

For the collection of n observations for $\mathbf{x}$ in X, the matrix form is:

$$X = \sum_{i=1}^{k} \mathrm{T}_k \mathrm{P}_k^T + \mathrm{E}_k \qquad (6.5)$$

where T_k is the $(n \times k)$ matrix of scores, P_k is the $(P \times k)$ matrix of loadings when only k principal components are retained and E_k is the $(n \times P)$ error matrix. E_k can be seen as the collection of error vectors $\mathbf{e}$, each corresponding to the vector $\mathbf{x}$ when only k principal components are retained:

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}} = \sum_{i=k+1}^{P} t_i \mathbf{p}_i \qquad (6.6)$$

6.3.2   PCA model for a DSS using EVM/ES for project schedule control

When PCA is implemented on a simulated schedule control reference, it is essential that this matrix X is in an appropriate format. This aspect has

been dealt with in numerous publications for the continuous monitoring of batch processes. An EVM/ES schedule control matrix can be considered to resemble the process matrix of a batch process that can be unfolded into a flat structure according to MacGregor and Kourti [26, 27]. Table 6.1 introduces the symbols and variables that are used in this section.

During the execution of a project, a functional EVM/ES system will require periodic measurements of the performance metrics displayed in table 6.1. In practice this might be done using software [16–18] and as specified within a contractual agreement or scheduled at distinct time interval along the project lifetime. The schedule control reference, from which principal components will be calculated, are fictitious project executions that are produced by a Monte Carlo simulation on activity durations. Different fictitious project executions can lead to different numbers of performance variables recorded for each execution. To have an equal amount of observations for each execution, equally spaced over the lifespan of a project, it is not appropriate to use a time index increasing from the start to the end of the project. Instead, our simulation model will use the project percentage complete (PC_t) as a monotonically rising, scaled time-indicator ranging from 0% to 100%. Without loss of generality, we will assume that after each ΔPC percentage of the work performed, the project schedule performance metrics are recorded.

We present our methodology in a general form, and proceed with a vector of observations $\mathbf{x}_{\kappa,1:J}$ with length J reported at each review period $\kappa \in \{1, \ldots, K\}$, when $PC = \kappa \Delta PC$ percentage of the work in the project is completed. In the experimental results sections of this paper, $\mathbf{x}_{\kappa,j}$ will be equal to $\left[SV, SPI, SV(t), SPI(t)\right]$ for $j \in \{1, \ldots, 4\}$ as the periodically reviewed EVM/ES schedule control metrics. The outcome of a Monte Carlo simulation on the duration of the activities results in a set of fictitious project executions, for which the EVM/ES observations are structured as the three dimensional matrix presented at the top of figure 6.2. For each periodic review period $\kappa \in \{1, \ldots, K\}$, J EVM/ES schedule control metrics are observed for n Monte Carlo runs. In order to use equation 6.1 to perform a PCA decomposition, the three dimensional matrix consisting of K times $n \times J$ matrices has to be unfolded into a big $n \times KJ$ (with $P = KJ$) two dimensional matrix as presented in figure 6.2.

Since we will combine SV and SV(t), which are expressed in monetary and time units respectively, with dimensionless indices SPI and SPI(t), we need to scale and center each column of the matrix X before the PCA is performed. The SV and SV(t) values will be much larger than the SPI and SPI(t) values. This could result in relatively higher weights assigned to the SV and SV(t) observations in the loading vectors found by PCA, which might then lead to wrong interpretations. We will adopt the normalisation per column which is most common in literature, but other weighted PCA

Earned Value/Earned Schedule

Project EVM/ES key metrics

BAC	Budget at completion for the project
PV_t	Planned value for the project at period t
EV_t	Earned value for the project at period t
ES_t	Earned schedule for the project at period t
PC_t	The percentage complete for a project at period t: $PC_t = EV_t/BAC$

Project EVM/ES performance metrics

$SV_t = EV_t - PV_t$	Schedule variance for the project at period t
$SPI_t = EV_t/PV_t$	Schedule performance index for the project at period t
$SV(t)_t = ES_t - t$	Schedule variance using earned schedule for the project at period t
$SPI(t)_t = ES_t/t$	Schedule performance index using earned schedule for the project at period t

with:

$t = 1, \ldots, T$	Current time period (otherwise denoted as AT)
T	Total duration of the project

Schedule control observations

$\boldsymbol{x}_{\kappa, 1:J}$	Vector of observations at review period κ, including J performance metrics
κ	The index for the review period ($\kappa \in 1 : K$)
K	The total number of review periods for which each project execution in the reference data has observations: $K = 100/\Delta PC$
ΔPC	The percentage complete increment between two review periods
J	The number of performance metrics included in the schedule analysis.
$P = J * K$	The total number of observations for each project execution.

Table 6.1: Symbols and abbreviations used in PCA for EVM/ES schedule control
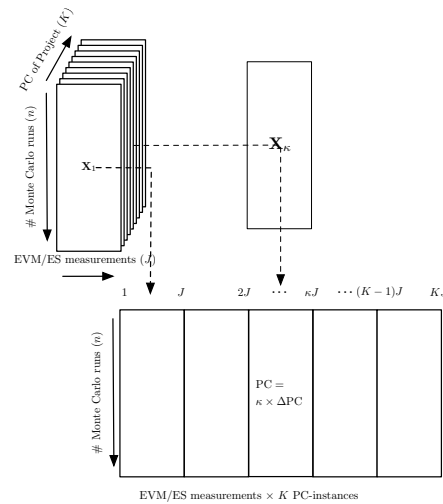
Figure 6.2: Unfolding of the three dimensional project data matrix ([28])

examples can also be found [22].

6.4 Multivariate schedule control in a DSS using EVM/ES

In this section, we will apply the PCA decomposition discussed in section 6.3 to produce two multivariate schedule control metrics (Hotelling's T^2 and SPE, section 6.4.1). In order to benefit from these metrics during the execution of a project , so that they can lead to enhanced decision making, we need to be able to apply them dynamically on project progress data. Section 6.4.2 introduces how missing observations are filled in to overcome the problems associated with the dynamic use of the T^2 and SPE metrics. These two newly proposed schedule metrics should improve the top-down schedule control process by generating correct warning signals. A signal is said to be produced when a tolerance limit for either T^2 or SPE is exceeded. Section 6.4.3 introduces how tolerance limits can be calculated prior to the execution of a project.

6.4.1   Two new schedule control metrics

Let us assume that we have a simulated schedule control reference in the appropriate matrix format X. This matrix contains observations for all J

EVM/ES schedule performance metrics, for all K review periods and for all n fictitious project executions produced by a Monte Carlo simulation. The PCA decomposition of X results in a matrix of loadings P and a matrix of scores T. Let us assume that the project is now executed in real life, which results in a vector of observations $\boldsymbol{x}_{new}$. This P-variate vector of observations will now be referenced against the PCA model, producing two schedule control metrics (T^2, section 6.4.1 and SPE, section 6.4.1), in order to enhance the decision making process in top-down schedule control.

Hotelling's T^2

Hotteling [21] proposed the T^2 measure as a multivariate extension of the t-statistic, frequently used in statistical hypothesis testing. If the population covariance Σ for the P-variate variable $\mathbf{x}$ is not known, it can be estimated from its sample covariance matrix S using n samples,

$$S = \frac{1}{n-1} \sum_{i=1}^{n} (\mathbf{x}_i - \bar{\mathbf{x}})^T (\mathbf{x}_i - \bar{\mathbf{x}}) \tag{6.7}$$

where $\bar{\mathbf{x}}$ is used to represent the P-variate mean of $\mathbf{x}$. Hotellings T^2 statistic can then be used to measure the weighted multivariate distance of $\boldsymbol{x}_{new}$ from this mean $\bar{\mathbf{x}}$:

$$T^2 = (\mathbf{x}_{new} - \bar{\mathbf{x}}) S^{-1} (\mathbf{x}_{new} - \bar{\mathbf{x}})^T \tag{6.8}$$

It should be noted that, in order to calculate T^2 directly from $\mathbf{x}_{new}$, the inverse of the covariance matrix S needs to be calculated. As discussed earlier, collinearity and noise in the multivariate variable severely impede the accurate calculation of this inverse. Calculating T^2 from the scores however, resulting from a prior PCA decomposition, is much more computationally stable. Principal components are mutually independent and thus, the covariance matrix is reduced to a diagonal matrix, for which calculating the inverse is trivial. The scores for the new vector of observations $\mathbf{t}_{new}$ can be found by projecting the P-variate EVM/ES observation vector $\mathbf{x}_{new}$ onto the k-dimensional principle component space.

$$\mathbf{t}_{new} = \mathbf{x}_{new} P \tag{6.9}$$

Hotellings T^2 can then be calculated as:

$$T_k^2 = \sum_{i=1}^{k} \left(\frac{(\mathbf{t}_{new})_i}{s_{\mathbf{t}_i}} \right)^2 \tag{6.10}$$

with $(\mathbf{t}_{new})_i$ the ith element of the vector of scores $\mathbf{t}_{new}$ and $s_{\mathbf{t}_i}^2$ the estimated variance of the ith principal component.

From equation 6.10, it is apparent that the last $P - k$ principal components should not be retained in the calculation. These explain very little of the variation in X and generally represent random noise or errors introduced by the measurement system. Calculating T^2 on the scores results in the summation of terms $((\mathbf{t}_{new})_i / s_{\mathbf{t}_i})^2$. Due to the small variances $s_{\mathbf{t}_i}^2$ explained by the last principal components, the slightest deviation (for principal components that have almost no effect on X) would lead to very large weighted distances. Therefore, retaining only the first k principal components is of importance to ensure that only the principal component that have the greatest influence on the EVM/ES schedule control vector $\mathbf{x}$ are expressed in this distance.

The squared prediction error SPE

Hotelling's T^2 statistic can be used to monitor the weighted distance of the vector of schedule performance observations as projected onto a reference defined by the principal components. However, this metric will only detect whether or not the observed variation is greater than what was contained within the reference data matrix X. When this variation on the activity level is significantly different than the reference variation, the basis formed by the principal components might no longer be representative. In order to monitor whether the applied principal component analysis transformation is still representative for the new vector of observations $\boldsymbol{x}_{new}$, the squared prediction error (SPE) should be calculated. SPE represents the squared perpendicular distance of the new P-variate observation from the projection space defined by the principal components and can be calculated as:

$$SPE = \mathbf{e}\mathbf{e}^T = \sum_{i=1}^{P} (\mathbf{x}_{new,i} - \hat{\mathbf{x}}_{new,i})^2 \tag{6.11}$$

where $\mathbf{e}$ is defined for $\mathbf{x}_{new}$ using equation 6.6 if k principal component are retained.

6.4.2   Dynamic use of the T^2 and SPE schedule control metrics

We want to apply the proposed T^2 and SPE schedule measures dynamically during the project execution phase. The new vector of observations $\mathbf{x}_{new}$ needs to be projected onto the principal component space produced

by the PCA decomposition of the schedule control reference X. Moreover, in order to calculate the scores $\mathbf{t}_{new}$ with equation 6.9, the $(1 \times P)$ vector $\mathbf{x}_{new}$ needs to be complete. However, at a review period $\kappa \in \{1, \ldots, K\}$ of the project in progress, only κJ of the P observations will be available. To come up with an estimate for the scores $\mathbf{t}_{new}$, we will need to deal with the problem of missing observations [29]. For notation purposes, we divide the vector of observations $\mathbf{x}$ into a $(1 \times \kappa J)$ vector $\mathbf{x}^*$ for which values are already recorded and a $(1 \times (K - \kappa)J)$ vector $\mathbf{x}^{\#}$ of missing measurements $(\mathbf{x} = \begin{bmatrix} \mathbf{x}^* & \mathbf{x}^{\#} \end{bmatrix})$.

The problem of producing an estimate for $\mathbf{t}_{new}$ when there are only observations available for $\mathbf{x}^*$ is described in the literature as dealing with missing data for on-line process monitoring. Different procedures have been proposed to deal with missing data in the literature and a comprehensive comparison is given by Nelson [30]. Conditional Mean Replacement (CMR) has been found to perform well by this study and is implemented here to produce an estimate $\hat{\boldsymbol{\tau}}$ for $\mathbf{t}_{new}$ during the dynamic project control process.

In CMR, the expected values for the conditional multivariate distribution are used to estimate $\mathbf{x}^{\#}$, given the present data and the most accurate estimate for the mean $\bar{\mathbf{x}}$ and covariance matrix S $(\hat{\mathbf{x}}^{\#} = \mathbb{E}(\mathbf{x}^{\#} | \mathbf{x}^*, \bar{\mathbf{x}}, \mathrm{S}))$. CMR allows us to estimate the score $\hat{\boldsymbol{\tau}}$ as:

$$\hat{\boldsymbol{\tau}} = \mathbf{x}^* \mathrm{S}_{22}^{-1} \mathrm{S}_{21} \mathrm{P}^{\#} + \mathbf{x}^* \mathrm{P}^* \tag{6.12}$$

with the covariance matrix S rewritten as:

$$\mathrm{S} = \begin{bmatrix} \mathrm{P}^{\#} \Theta \mathrm{P}^{\#T} & \mathrm{P}^{\#} \Theta \mathrm{P}^{*T} \\ \mathrm{P}^* \Theta \mathrm{P}^{\#T} & \mathrm{P}^* \Theta \mathrm{P}^{*T} \end{bmatrix} = \begin{bmatrix} \mathrm{S}_{11} & \mathrm{S}_{12} \\ \mathrm{S}_{21} & \mathrm{S}_{22} \end{bmatrix} \tag{6.13}$$

where $\Theta = \mathrm{T}^T \mathrm{T} / (n-1)$ represents a diagonal matrix with the variances explained by each principal component on its diagonal and the matrix of loadings is restructured as $\mathrm{P} = \begin{bmatrix} \mathrm{P}^{\#} \\ \mathrm{P}^* \end{bmatrix}$.

6.4.3 Tolerance limits for the T^2 and SPE schedule control metrics

During the execution of a project, the T^2 and SPE metrics, calculated at the top WBS level, should be interpreted to infer a state of activity level schedule control. The most straightforward interpretation follows when a warning signals arises at the top WBS level. If either the T^2 or the SPE metric exceeds a tolerance limit, a warning signal is said to be produced. This warning signal should then indicate that the activity level performance does not longer conform a pre-defined state of schedule control.

For independence with respect to distributional assumptions, we propose the use of the empirical cumulative distribution function (ecdf) for T^2 and SPE. In practice, this requires the calculation of the T^2 and SPE schedule control metric at each review period ($\forall 1 \leq \kappa \leq K$) and for each fictitious project execution in our schedule control reference X. Obviously, the estimate $\widehat{\tau}$, produced using CMR, now replaces the score vector $\mathbf{t}_{new}$ in calculating T^2 and SPE for all $\kappa < K$. The tolerance limits at a review period κ for T^2 and SPE can then be calculated in accordance with a tolerance level α as the αth sample quantile. We refer to Hyndman and Fan [31] for more detail on the calculations of the sample quantiles.

6.5   Experimental test design

In this section the design of the experiments to characterise the performance of the T^2 and the SPE metrics for project control is outlined. Section 6.5.1 introduces the data generation process, to create a project benchmark set. In section 6.5.2 the Monte Carlo simulations are described in detail. Section 6.5.3 provides the measures to quantify the performance of the decision process using the T^2 and the SPE schedule control metrics.

6.5.1   Data generation

We test the performance of the T^2 and SPE schedule control metrics on a set of 900 projects generated by the project network generator RanGen [32], recommended by Herroelen [33] to protect the test ensemble from a possible bias in network structure. Projects from this set were extensively used in previous research on project control [3, 5, 10, 34].

The baseline durations are randomly assigned to the 30 activities in the projects. They are sampled from a uniform distribution between 8 and 56 days. The baseline duration (estimate) for an activity i will be denoted as $\hat{d}_i$. The fixed cost for each activity is sampled uniformly between $0\,€$ and $500\,€$ and the variable cost is sampled uniformly between $700\,€$ and $1500\,€$. The schedule, referred to as the baseline during the execution of the project, is the earliest start schedule obtained from a single forward pass of the critical path method.

6.5.2   Monte Carlo simulations

In order to quantify the use of the two newly proposed schedule control metrics, we ran an extensive set of two-phased Monte Carlo simulation experiments. Section 6.5.2 describes the dynamic project progress model with which EVM/ES measurements are produced for fictitious project executions. Section 6.5.2 recounts how Monte Carlo simulations serve a double purpose in this paper and section 6.5.2 describes the combined use of variation and risk input modelling for activity durations.

EVM/ES model

Fictitious project executions are simulated using P2 Engine [18] to generate EVM/ES data at $K = 19$ distinct PC intervals, with $\Delta PC = 5\%$, from 5% to 95%. At any review period $\kappa \in 1, .., K$ we chose $\mathbf{x}_{\kappa,1:J} = \begin{bmatrix} SV, SPI, SV(t), SPI(t) \end{bmatrix}$ for $j \in \{1, \ldots, 4\}$ to be the periodically reviewed EVM/ES schedule control metrics. A total of $P = 19 \times 4 = 76$ original variables is recorded for the matrix X.

Our model assumes that the earned value (EV) for a single activity follows a linear accrue, starting from its actual start up to its budget at completion (BAC) when it is finished [3]. Planned value (PV) follows this same linear accrue from the planned start up to the planned finish time of the activity. EV and PV are calculated in P2 Engine at the project level and are compared to produce the SV, SPI, SV(t) and SPI(t) schedule performance metrics at each review period κ.

The calculations for the PCA decomposition and the CMR procedure to produce the T^2 and SPE schedule control metrics, along with the analysis of the results presented in section 6.6, were implemented in the statistical programming language R [35].

Two-phased experiment

Monte Carlo simulations serve a dual purpose in our experiments. In the *first phase* the outcome of a large simulation (10,000 runs) is used to build a schedule control reference set for each of the 900 projects, which is then used to perform the PCA decomposition on for our enhanced DSS using EVM/ES. The preferred state of schedule control will then determine which fictitious project executions end up in the reference data matrix X. We will provide additional detail on how we define this state of schedule control in section 6.5.3.

In the *second phase* of the Monte Carlo experiment, project progress situations are simulated (now with 1,000 runs per project), where the activity level performance might not conform with the pre-defined state of schedule control. This set is used to quantify the performance of the decision process that follows from our DSS using EVM/ES for project schedule control. The warning signals that are generated by the T^2 and SPE control charts should accurately indicate whether or not the pre-defined state of control can be confirmed at the activity level. We will provide details on how the performance of the decision process for top-down schedule control is quantified in section 6.5.3

Activity duration input modelling

The Monte Carlo simulations in this paper produce fictitious executions to generate project progress data. To that purpose, we need an appropriate model to accurately represent the uncertainty experienced at the activity level of the project. We opted for a combined input modelling where both risk and variation, often considered as separate sources of uncertainty in project management literature [36], are represented using separate probability distributions. We have chosen to implement probability functions from the family of generalised beta distributions, which have long been used in project management [3, 34, 37] due to their ability to accurately mimic the behaviour of random input processes driving the system [38], and their association to PERT-style three point estimates [39]. The probability density function for a generalised beta random variable $\mathbf{D}$ can be stated as:

$$f_{\mathbf{D}}(d|a,b,\theta_1,\theta_2) = \begin{cases} \frac{\Gamma(\theta_1+\theta_2)}{\Gamma(\theta_1)\Gamma(\theta_2)} \frac{(d-a)^{\theta_1-1}(b-d)^{\theta_2-1}}{(b-d)^{\theta_1+\theta_2-1}} & \text{if } a \leq d \leq b \\ 0 & \text{if } d < a \vee b < d \end{cases}$$
(6.14)

where $\Gamma()$ denotes the gamma function and θ_1 and θ_2 represent shape parameters. We propose the use of a parameter vector $\boldsymbol{\omega} = (a,b,m,\mu)$ with estimates for the minimum (a), the maximum (b), the mode (m) and the mean (μ) of the distribution, expressed as fractions of the baseline estimate duration $\hat{d}_i$ for activity $i \in \mathcal{N}$. In doing so, we can sample for all activities i, using only a limited number of distributions, represented by their parameter vector $\boldsymbol{\omega}$ (for which settings are presented in table 6.2). From $\boldsymbol{\omega}$ the shape parameters can be found using:

$$\begin{cases} \theta_1(\boldsymbol{\omega}) &= -\frac{(b+a-2m)(a-\mu)}{(m-\mu)(a-b)} \\ \theta_2(\boldsymbol{\omega}) &= \frac{(b+a-2m)(b-\mu)}{(m-\mu)(a-b)} \end{cases}$$
(6.15)

In each Monte Carlo simulation, *risk* is first modelled using the concept of linear association [5, 40] and a parameter vector $\boldsymbol{\omega}_R$ is assigned. Due to

the absence of a unified methodology to test the impact of riskful events or dependencies between activities in the project management literature, Trietsch et al. [40] suggest the use of a positive random variable B to act as a bias term in simulations of project executions. B is most easily perceived as a consistent over- or underestimation of activity durations, or a project-wide effect of an uncertain event. In our experiments, we will assign a parameter vector $\boldsymbol{\omega}_R$, to represent this risk factor. Consequently, for each fictitious project execution in the simulation, a bias term is sampled from the generalised beta distribution with parameter vector $\boldsymbol{\omega}_R$. Subsequently, for all activities in the project, variation is added using a parameter vector $\boldsymbol{\omega}_V$, which can be chosen independently from $\boldsymbol{\omega}_R$. The duration d_i of an activity i in a fictitious project execution can then ultimately be considered as a sample from a generalised beta distribution (with parameter vector $\boldsymbol{\omega}_V$) multiplied with a biased baseline estimate $B\hat{d_i}$.

In the experiments, for which results will be shown in section 6.6, $\boldsymbol{\omega}_R$ and $\boldsymbol{\omega}_V$ are chosen separately from the set of parameter vectors presented in table 6.2. In the *general performance experiment*, the parameter vectors are chosen such that they represent realistic activity duration distributions for activities that either finish early ($\boldsymbol{\omega}_1$), on time ($\boldsymbol{\omega}_2$) or late ($\boldsymbol{\omega}_3$), on average. From the distribution for which activities end on time, we derived two distinct sets of parameter vectors ($\boldsymbol{\omega}_{2\mu_1}$ to $\boldsymbol{\omega}_{2\mu_5}$ and $\boldsymbol{\omega}_{2s_1}$ to $\boldsymbol{\omega}_{2s_6}$) to model changes in respectively the mean and the standard deviation. These can be used in the *sensitivity experiment* to test the robustness to over- or underestimation, when assumptions are made with respect to distributional characteristics of the activity durations.

6.5.3 Quantifying the performance of the decision making in a DSS using EVM/ES

In order to show the enhanced decision making that follows from our multivariate schedule control metrics in a DSS using EVM/ES for top-down schedule control, we will compare it to the schedule control process using statistical tolerance limits. This univariate procedure was presented by Colin and Vanhoucke (2014) [5] and uses statistical tolerance limits for SV, SPI, SV(t) and SPI(t) to steer the decision making in top-down schedule control using EVM/ES. The schedule control metrics were applied either directly (X chart) or the difference between two consecutive measurements was monitored (R chart). The X and R charts were shown to outperform the decision making based on rules-of-thumb that is omnipresent in EVM/ES schedule control in practice.

Moreover, we were able to obtain a fair comparison in our experiments between the newly proposed multivariate schedule control metrics and the

General performance experiment			Sensitivity experiment					
				Mean			Standard deviation	
ω	(a, b, m, μ)	σ	ω	(a, b, m, μ)	σ	ω	(a, b, m, μ)	σ
ω_1	$(0.1, 1.2, 0.7, 0.6)$	0.38	$\omega_{2\mu_1}$	$(0.2, 4, 0.51, 0.7)$	0.3	ω_{2s_1}	$(0.2, 4, 0.40, 1)$	0.50
			$\omega_{2\mu_2}$	$(0.2, 4, 0.90, 1.0)$	0.3	ω_{2s_2}	$(0.2, 4, 0.70, 1)$	0.42
ω_2	$(0.2, 4, 0.9, 1)$	0.30	$\omega_{2\mu_3}$	$(0.2, 4, 1.22, 1.3)$	0.3	ω_{2s_3}	$(0.2, 4, 0.85, 1)$	0.35
			$\omega_{2\mu_4}$	$(0.2, 4, 1.57, 1.6)$	0.3	ω_{2s_4}	$(0.2, 4, 0.92, 1)$	0.28
			$\omega_{2\mu_5}$	$(0.2, 4, 1.89, 1.9)$	0.3	ω_{2s_5}	$(0.2, 4, 0.96, 1)$	0.20
ω_3	$(0.9, 4, 1.3, 1.4)$	0.38				ω_{2s_6}	$(0.2, 4, 0.98, 1)$	0.15

Table 6.2: Parameter vectors for the generalised beta distribution

Figure 6.3: Activity level test for schedule control: Kolmogorov-Smirnov

univariate schedule control process using EVM/ES. Since, the same fictitious project executions were used to calculate tolerance limits for both the univariate and multivariate EVM/ES schedule control metrics. Subsequently, we compared the warning signals generated by all performance metrics with respect to the underlying activity level of the project for additional fictitious project executions in the second phase, in order to quantify the performance of the decision making process.

In order to demonstrate the enhanced decision process that follows from the newly proposed multivariate control metrics, we first needed to define a *state of schedule control*, as established at the activity level of the project. The performance of the decision process could then be quantified by whether or not this state of control can be correctly inferred from the observations made at the highest WBS level. We measured this performance using two descriptive quantities, the *detection performance* and the *probability of overreactions*. These can be combined into a single measure, known in the classification techniques literature as the *area under the curve*.

Defining a state of schedule control   The state of schedule control is primarily defined by the fictitious project executions that are included in the schedule control reference matrix X. This reference is then used to compare with future executions and to calculate tolerance limits. We will illustrate along the following lines how a set of fictitious project executions, generated in the first phase of the simulation model, can be chosen to constitute the project schedule control reference.

In the first phase of the Monte Carlo simulation, specific values are chosen for the parameter vectors ω_R and ω_V, which results in a probability

Area under the curve
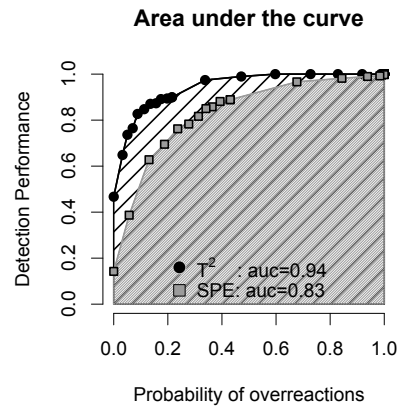
Figure 6.4: Illustration of the duality of the detection performance and the probability of overreactions. The area under the curve allows to combine both measures into one descriptive value

distribution for the simulated real activity durations of a project. Figure 6.3 shows the approximated density of the scaled activity durations $(d_i/\hat{d}_i)$ from a sample of 10,000 fictitious project executions, where $\omega_R = \omega_2$ and $\omega_V = \omega_2$. The histograms in lightgrey and darkgrey in figure 6.3 represent two fictitious project executions, from this sample of 10,000. Even though these fictitious executions where simulated from the same distribution, they are very dissimilar in terms of their appearance on the activity duration scale.

Figure 6.3 illustrates how it can be difficult to define a state of schedule control due to random variation. We will therefore employ a metric to describe how close a fictitious project execution lies to the empirical density, approximated from all 10,000 samples. We test for each execution whether it is likely to have been sampled from the empirical distribution function, which is based on ω_R and ω_V. We do this using the two-sample Kolmogorov-Smirnov (K-S) statistic, a well-established non-parametric statistic to test the equality of two distributions [41]. Corresponding to a significance level of $\eta = 0.001$, a critical value (K-S$_\eta = 0.36$) for the K-S statistic can be found [42]. Thus, using the K-S statistic and its critical value, we will only include those fictitious project executions, in our schedule control reference, for which the sample of activity durations is likely to have been drawn from the distribution defined by ω_R and ω_V (K-S < K-S$_\eta$).

Detection performance   In the second phase of the simulation experiment, we test whether the signals generated at the highest WBS level correspond to the underlying activity level schedule performance. In other words, for a project execution that does not conform with the defined state of schedule control (K-S > K-S$_\eta$), the project manager should be given a warning during the lifetime of the project, using the control charts that act as the user interface of the DSS. The *detection performance* is thereby calculated for all runs in the second phase that do not comply to the state of schedule control, as the ratio of the number of those runs that generate a signal to the total number of those runs.

Probability of overreactions   In conjunction with the detection performance, a dual measure needs to be formulated. The *probability of overreactions* measures whether the control charts provide a false warning signal to the project manager, who is the end user of the DSS. If a warning signal is produced at the highest WBS level, we do not want the project manager's effort spent in drilling down the WBS to be in vain. The probability of overreactions can be calculated from all runs in the second phase of the simulation experiment for which the state of schedule control can still be assumed to be representative (K-S < K-S$_\eta$).

Area under the curve   Figure 6.4 shows an illustration of the probability of overreactions and the detection performance for the multivariate schedule control metrics proposed in this paper. The probability of overreactions is denoted along the x-axis and the detection performance along the y-axis. An ideal control metric should have an as high as possible detection performance, whilst keeping the probability of overreactions close to 0. The points presented on the graph represent different probability of overreaction/detection performance couples for varying values of the level α. This α represents the αth quantile used for the statistical tolerance limits. In order to capture the dynamics of both the probability of overreactions and the detection performance into a single descriptive measure, we propose the use of the *area under the curve*, which should be maximised for as good as possible decision making from the DSS using EVM/ES for top-down schedule control. This measure is widely used in classification testing and machine learning [43] and can be obtained by Riemann integration of the curves produced by the probability of overreaction/detection performance couples, as displayed in figure 6.4.

General performance
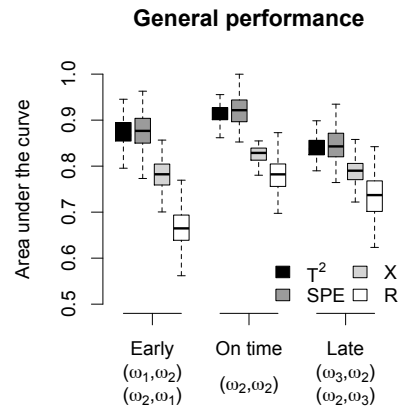
Figure 6.5: General performance comparison of the models for a DSS using
EVM/ES for top-down schedule control

6.6   Results and Discussion

In order to demonstrate the improved decision making that follows from
our multivariate model for a DSS using EVM/ES, we conducted the fol-
lowing experiments. Using the parameter vectors introduced in table 6.2
of section 6.5.2, we conducted two large simulation experiments. Using
the parameter vectors outlined under *general performance experiment*, we
will discuss the overall improvement of multivariate schedule control over
the univariate use of EVM/ES metrics in section 6.6.1. While using these
results, we explore how many principal components should be retained in
the analysis for schedule control. The *sensitivity experiment* was conducted
using the parameter vectors for the generalised beta family of distributions
where either the mean ($\omega_{2\mu_1}$ to $\omega_{2\mu_5}$) or the standard deviation is varied
(ω_{2s_1} to ω_{2s_6}). In section 6.6.2, we present the effect of an under- or over-
estimation of the mean or standard deviation on the performance of the
decision making process. In section 6.6.3, we will add discussion to the
results that are presented here.

6.6.1   General performance experiment

In order to compare the general performance of the multivariate and uni-
variate schedule control metrics described in this paper, a simulation ex-
periment was carried out with scenarios where activities are either early,
on time or late, on average. Figure 6.5 displays the assignments made to

Number of principal components   **EVM/ES schedule control metric**
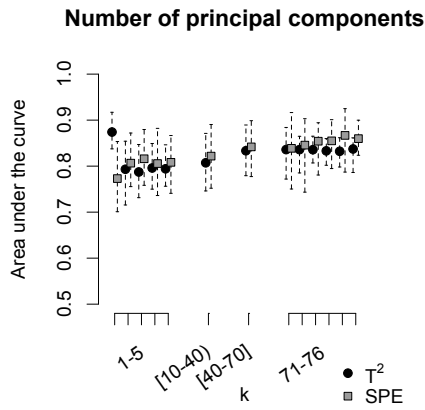
Figure 6.6: Optimal number of principal components for the multivariate schedule control metrics
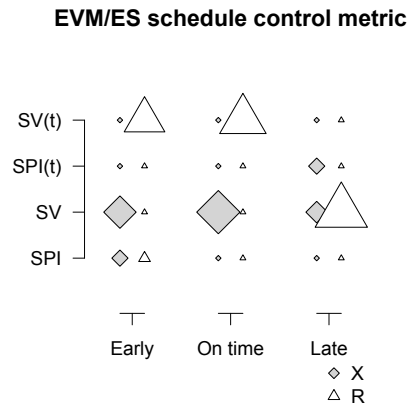
Figure 6.7: Optimal schedule control metrics

the couple (ω_R,ω_V) to obtain these scenarios, along its x-axis.

The boxplots of figure 6.5 present the recorded area under the curve for the multivariate T^2 and SPE metrics, and the univariate schedule control metrics X and R from Colin and Vanhoucke (2014) [5]. Overall, figure 6.5 shows that the multivariate metrics outperform the traditional univariate use of EVM/ES. The difference between the T^2 metric and the squared prediction error SPE is small, with the latter outperforming the former slightly. The difference between the multivariate and the univariate control metrics becomes smaller in the situations where activities are on time or late on average, but the multivariate approaches still show to be significantly better. In conclusion, our multivariate approach is likely to lead to enhanced decision making in a DSS using EVM/ES for top-down schedule control.

Number of principal components to retain   The results of figure 6.5 are restructured to analyse the optimal number of principal components that needs to be retained for building an as accurate as possible PCA model. Figure 6.6 shows the recorded area under the curve for the T^2 and SPE metrics in function of the number of principal components retained in the model k. It shows that, in general, the T^2 metric benefits from a very low number of principal components (preferably only 1), while the SPE metric performs better when more principal components are included in the PCA model.
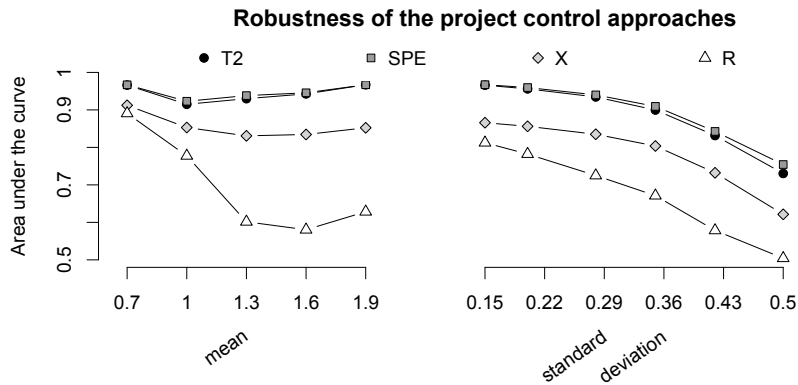
Figure 6.8: Influence of a change in the mean and the standard deviation on the project schedule control performance

Analogously, we investigated which EVM/ES control metric (SV, SPI, SV(t), SPI(t)) should be used in the schedule control procedures X and R in order to deliver the best performance. We present these findings in figure 6.7, where a birds-eye perspective is given on six histograms in which the relative size of the points represents how often a certain metric is found to be performing best. A single EVM/ES metric that outperforms all other could not be found. Therefore a combined use of all EVM/ES schedule control metrics is advised, where none is neglected during the execution of the project. This can been seen as a confirmation of the problems discussed in section 6.2, which inspired our multivariate model for a DSS using EVM/ES for top-down project schedule control.

6.6.2   Sensitivity experiment

The statistical tolerance limits for the schedule control metrics discussed in this paper are generated from a simulated set of fictitious project executions. These are generated using a Monte Carlo simulation, for which probability distributions serve as input to model the real activity durations. In a realistic project environment, it is not always possible to produce probability distributions that accurately reflect the stochastic nature of the input processes. Consequently, an under or over-estimation of one of the distributional characteristics can not always be avoided.

We model this situation using the parameter vectors $\omega_{2\mu_1}$ to $\omega_{2\mu_5}$ to test a deviation in the mean and ω_{2s_1} to ω_{2s_6} to test a deviation of the standard

deviation. In the sensitivity experiment, the statistical tolerance limits are produced from a state of schedule control defined by $\omega_R = \omega_V = \omega_2$. Additional fictitious project executions are generated with alternately one of the parameter vectors $\omega_{2\mu_1} \ldots \omega_{2\mu_5}$ or $\omega_{2s_1} \ldots \omega_{2s_6}$ assigned to either ω_R or ω_V, in order to test the effect of a change in the mean or standard deviation.

Figure 6.8 shows the calculated area under the curve for the univariate and multivariate procedures that are used in a DSS for top-down schedule control, as described in this paper. On the left x-axis, the average activity duration for the additional runs, from which the detection performance and probability of overreactions are calculated, is presented. The standard deviation is presented on the right x-axis. The pre-defined state of schedule control ($\omega_R = \omega_V = \omega_2$) has mean 1 and standard deviation 0.3.

From figure 6.8, we can conclude that the performance of the decision making process that follows from the R schedule control procedure decreases drastically for projects for which the average activity duration increases. The decision making processes based on T^2, the SPE and the X schedule control procedure seem to be much more robust to under- or overestimates of the mean activity duration.

With respect to the sensitivity to changes in the standard deviation of the underlying activity durations, all of the investigated schedule control procedures exhibit the same behaviour. When the standard deviation is larger than what was assumed for calculating the schedule control reference, the performance of the decision making process decreases significantly, as shown on the right of figure 6.8.

6.6.3   Discussion

In this section, we will add some discussion to the results that are presented in this section. Although we have shown how our multivariate model can lead to enhanced decision making for a top-down schedule control process, some aspects still need to be addressed.

The newly proposed schedule metrics T^2 and SPE outperform the current best-known choices for schedule control using EVM/ES with respect to the decision making by the project manager of whether or not he/she should invest time and effort in drilling down the project schedule. Despite its improved performance compared to the traditional univariate approach, the area under the curve is still mostly lower than 1, which means that some portion of the activity level performance is still obscured by the aggregated EVM/ES observations at the high WBS level of the project. This issue is intrinsically connected to a top-down schedule control process, but our re-

search shows that considerable improvements can be made with the application of a multivariate model to the EVM/ES measurement system.

The practicality of the DSS for the end user is also significantly improved. Whereas all schedule control metrics (SV, SPI, SV(t), SPI(t)) should be monitored in a univariate model to ensure the best possible decision making, we show that the use of a single multivariate metric (T^2 or SPE) can lead to considerable improvements. The data overload from which the univariate EVM/ES models suffer is thereby reduced. When only a single multivariate metric should be monitored by the end user of the DSS, this is likely to improve the accuracy and the timeliness of the decision making. In addition, a single multivariate metric is also useful from a reporting and communications point of view. Moreover, due to the application of a PCA, redundancies and noise are removed from the inference process of underlying activity level schedule performance.

6.7   Conclusion

In this paper, we present a multivariate model for a DSS using EVM/ES for top-down project schedule control. The need for a multivariate approach is dictated by the very nature of the EVM/ES measurement system. Observations for the EVM/ES variables are characterised by a data-overload, noise and redundancy. Based on the principle of batch process control from the chemometrics literature, we have initiated a multivariate model to be used in a DSS using EVM/ES for top-down project schedule control. In the resulting DSS, all the available EVM/ES observations are used in a structured manner, through analysis of their correlation structure. Two multivariate schedule control metrics (T^2 and SPE) are proposed, that can be presented to the end user of the DSS on control charts.

We tested our multivariate model on a large simulation experiment that is in line with previous project control research. The performance of the T^2 and the SPE schedule control metrics was quantified and compared to the univariate use of EVM/ES observations in a DSS for top-down schedule control. The T^2 and the SPE metrics outperform the univariate EVM/ES approaches in terms of their ability to infer underlying activity level schedule performance more accurately, which ultimately leads to enhanced decision making. Moreover, the use of the multivariate control metrics leaves a project manager with only a single metric to monitor over the lifetime of a project. Traditional use of the EVM/ES metrics would require monitoring all the EVM/ES schedule control metrics, since their performance might vary from situation to situation. The SPE metric slightly outperforms the T^2 metric, but both these multivariate schedule control metrics are likely to improve not only the accuracy and the timeliness of the inference process of the underlying state of control at the activity level, but are also useful

from a reporting and communications point of view.

Further work on the topic of multivariate metrics for top-down schedule control will expand on the results presented here. Other projection methods or transformations might improve the inference process even more, which will consequently lead to more improvements in the decision making process. PCA should currently be considered as the current best option for a model to handle EVM/ES measurements in a DSS for top-down project schedule control.

References

[1] PMBOK, A Guide to the Project Management Body of Knowledge, Third Edition, Newtown Square, Pa.: Project Management Institute, Inc., 2004.

[2] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[3] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[4] C. Fang, F. Marle, A simulation-based risk network model for decision support in project risk management, Decision Support Systems 52 (3) (2012) 635 – 644. doi:http://dx.doi.org/10.1016/j.dss.2011.10.021.
URL http://www.sciencedirect.com/science/article/pii/S0167923611002016

[5] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega The International Journal of Management Science 49 (2014) 107–122.

[6] H. Hunter, R. Fitzgerald, D. Barlow, Improved cost monitoring and control through the earned value management system, Acta Astronautica 93 (0) (2014) 497 – 500. doi:http://dx.doi.org/10.1016/j.actaastro.2012.09.010.
URL http://www.sciencedirect.com/science/article/pii/S0094576512003670

[7] H. L. Chen, Improving forecasting accuracy of project earned value metrics: Linear modeling approach, Journal of Management in Engineering 30 (2) (2014) 135–145.

[8] M. Plaza, O. Turetken, A model-based DSS for integrating the impact of learning in project control, Decision Support System 47 (2009) 488–499.

[9] H. Khamooshi, H. Golafshani, EDM: Earned duration management, a new approach to schedule performance management and measurement, International Journal of Project Management 32 (6) (2014) 1019–1041.

[10] R. Elshaer, Impact of sensitivity information on the prediction of project's duration using earned schedule method, International Journal of Project Management 31 (2013) 579–588.

[11] G. M. Marakas, Decision support systems in the 21st century, Vol. 2, Pearson Education, Prentice Hall, New Jersey, 2003.

[12] F. Anbari, Earned value project management method and extensions., Project Management Journal 34(4) (2003) 12–23.

[13] Q. Wang, N. Jiang, L. Gou, M. Che, R. Zhang, Practical experiences of cost/schedule measure through earned value management and statistical process control, Lecture Notes in Computer Science 3966 (2006) 348–354.

[14] S. S. Leu, Y. C. Lin, Project performance evaluation based on statistical process control techniques, Journal of Construction Engineering and Management 134 (2008) 813–819.

[15] R. Aliverdi, L. Moslemi Naeni, A. Salehipour, Monitoring project duration and cost in a construction project by applying statistical quality control charts, International Journal of Project Management 31 (3) (2013) 411–423.

[16] DecisionEdge, `http://www.decisionedge.com/` (2014).

[17] M. Vanhoucke, Dynamic Scheduling: Integrating Schedule Risk Analysis with Earned Value Management, The Measurable News 2 (2012) 11–13.

[18] P2 Engine, Visit the P2 Engine website at www.p2engine.com (Jan. 2013).

[19] G. W. Stewart, et al., Collinearity and least squares regression, Statistical Science 2 (1) (1987) 68–84.

[20] J. F. MacGregor, Using on-line process data to improve quality: Challenges for statisticians*, International Statistical Review 65 (3) (1997) 309–323.

[21] H. Hotelling, A generalized t test and measure of multivariate dispersion, Proc. Second Berkeley Symp. on Math. Statist. and Prob. (1951) 23–41.

[22] S. Bersimis, S. Psarakis, J. Panaretos, Multivariate statistical process control charts: an overview, Quality and Reliability Engineering International 23 (5) (2006) 517–543.

[23] K. Pearson, Liii. on lines and planes of closest fit to systems of points in space, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2 (11) (1901) 559–572.

[24] I. Jolliffe, Principal component analysis, Wiley Online Library, 2005.

[25] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammerling, A. McKenney, et al., LAPACK Users' guide, Vol. 9, Siam, 1999.

[26] J. MacGregor, T. Kourti, Statistical process control of multivariate processes, Control Engineering Practice 3 (3) (1995) 403–414.

[27] T. Kourti, J. F. MacGregor, Process analysis, monitoring and diagnosis, using multivariate projection methods, Chemometrics and intelligent laboratory systems 28 (1) (1995) 3–21.

[28] D. S. Lee, J. M. Park, P. A. Vanrolleghem, Adaptive multiscale principal component analysis for on-line monitoring of a sequencing batch reactor, Journal of Biotechnology 116 (2) (2005) 195.

[29] P. Nomikos, J. F. MacGregor, Multivariate SPC charts for monitoring batch processes, Technometrics 37 (1) (1995) 41–59.

[30] P. R. Nelson, J. F. MacGregor, P. A. Taylor, The impact of missing measurements on pca and pls prediction and monitoring applications, Chemometrics and intelligent laboratory systems 80 (1) (2006) 1–12.

[31] R. J. Hyndman, Y. Fan, Sample quantiles in statistical packages, The American Statistician 50 (4) (1996) 361–365.

[32] E. Demeulemeester, M. Vanhoucke, W. Herroelen, Rangen: A random network generator for activity-on-the-node networks, Journal of Scheduling 6 (2003) 17–38.

[33] W. Herroelen, Project scheduling — theory and practice, Production and Operations Management 14 (4) (2005) 413–432.

[34] M. Wauters, M. Vanhoucke, Support vector machine regression for project control forecasting, Automation in Construction 47 (2014) 92–106.

[35] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2013). URL `http://www.R-project.org`

[36] C. Loch, A. De Meyer, M. Pich, Managing the unknown: A new approach to managing high uncertainty and risk in project, John Wiley and Sons, Inc., New Jersey., 2006.

[37] S. AbouRizk, D. Halpin, J. Wilson, Fitting beta distributions based on sample data, Journal of Construction Engineering and Management 120 (1994) 288–305.

[38] M. E. Kuhl, E. K. Lada, N. M. Steiger, M. A. Wagner, J. R. Wilson, Introduction to modeling and generating probabilistic input processes for simulation, in: S. Henderson, B. Biller, M. Hsieh, J. Shortle, J. Tew, R. Barton (Eds.), Proceedings of the 2007 Winter Simulation Conference, New Jersey: Institute of Electrical and Electronics Engineers, 2007, pp. 63–76.

[39] W. J. McBride, C. W. McClelland, Pert and the beta distribution, Engineering Management, IEEE Transactions on (4) (1967) 166–169.

[40] D. Trietsch, L. Mazmanyan, L. Govergyan, K. R. Baker, Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation, European Journal of Operational Research 216 (2012) 386–396.

[41] A. N. Pettitt, M. A. Stephens, The kolmogorov-smirnov goodness-of-fit statistic with discrete and grouped data, Technometrics 19 (2) (1977) 205–210.

[42] G. Marsaglia, W. W. Tsang, J. Wang, Evaluating kolmogorov's distribution, Journal of Statistical Software 8 (18) (2003) 1–4.

[43] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshirani, The elements of statistical learning, Vol. 2, Springer, 2009.

7

On the use of multivariate regression methods for longest path calculations from earned value management observations
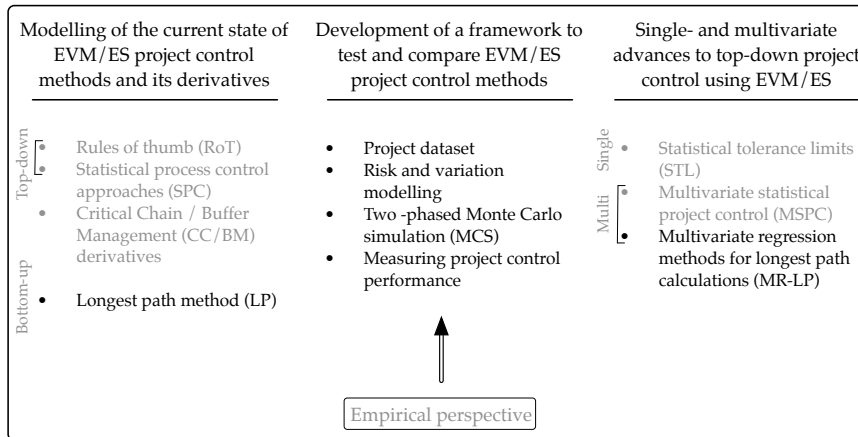
Figure 7.1: Schematic overview of the elements, of the larger body of research in this book, that are discussed in chapter 7.

Abstract

This paper explores the use of multivariate regression methods for project schedule control, within a statistical project control framework (Colin and Vanhoucke 2014). These multivariate regression methods monitor the activity level performance of an ongoing project from the earned value management/earned schedule (EVM/ES) observations that are made at a high level of the work breakdown structure (WBS). These estimates can be used to calculate the longest path in the project and to produce warning signals for project schedule control. The effort that is spent by the project manager is thereby reduced, since a drill-down of the WBS is no longer required for every review period. An extensive computational experiment was set up to test and compare four distinct multivariate regression methods on a database of project networks. The kernel principal component regression method, when used with a radial base function kernel, was found to outperform the other presented regression methods.

7.1   Introduction

The topic of schedule control has received a lot of attention in the recent literature on project management. The baseline schedule for a project is often calculated under the limited availability of resources [1]. In this paper, the use of a such baseline schedule in a realistic stochastic environment has

been put into the right perspective. A baseline schedule can only act as a single point of reference for the actual progress, during the dynamic execution of a project [2–4]. The earned value management/earned schedule (EVM/ES) methodology [5, 6] provides schedule performance measures to assist a project manager in his/her decision making [7, 8]. However, in order for the schedule control process not to be too time consuming and interruptive, Lipke et al. (2009) [6] argue that it should be implemented at a high level of the work breakdown structure (WBS) of the project. This means that aggregated measures of schedule performance from the EVM/ES system should be monitored regularly during the project execution at a high level of the WBS. This is denoted in literature as top-down schedule control [9, 10]. Consequently, the decision making that follows from these EVM/ES measures is often impeded, since activity level detail might be lost when the aggregated measures are calculated.

Recently, improvements for the top-down schedule control process using EVM/ES have been proposed in the literature. These studies suggest the use of statistical methods to derive control limits for the EVM/ES schedule performance measures that are observed at a high level of the WBS [10–13]. The necessary *statistical project control* framework to develop and test such control limits was presented by Colin and Vanhoucke (2014) [10]. However, a recent study [14] shows that the most efficient project control methodology still requires dynamic longest path calculations during the execution of the project. This means that, at each review period during the execution of the project, the complete activity level performance of the project needs to be recorded in order to calculate the actual longest path. This places a heavy burden on the project manager in terms of the effort that is required to drill down the WBS. In this paper, we propose a solution to this problem by using the EVM/ES metrics that are observed at a high level of the WBS to estimate the durations of the individual activities. In doing so, the dynamic longest path calculations can be performed using only the directly observed EVM/ES schedule performance measures. In section 7.2, we will first discuss how in prior literature, the top-down schedule control process is characterised by an implicit inference of activity level performance. We will then proceed with a more explicit characterisation of this inference process. This allows us to use multivariate regression methods to estimate the activity level schedule performance of a project from EVM/ES schedule performance measures that are observed at a high level of the WBS. Consequently, the longest path through the project at each review period can then be calculated, without requiring drilling down the WBS. We apply four distinct multivariate regression models (principal component regression (PCR), partial least squares regression (PLSR), kernel PCR (kPCR) and kernel PLSR (kPLSR)) for this purpose, which we present in section 7.3.

This paper presents a contribution to the theory on project management

and control that is threefold. First, we present a theoretical basis to the problem of estimating the activity level performance in a project execution from EVM/ES observations, that are made at a high level of the WBS. Second, we implement four distinct multivariate regression models, which are tested in an extensive computational experiment and for which the parameters are optimised. Third, this study also presents a practical use for the estimation of the activity durations from high WBS observations, since we combine them with Lipke (2012) [15]'s longest path (LP) method in order to produce warning signals for project schedule control. The remaining sections that constitute this paper and which have not been introduced, will now be outlined. In section 7.4, we describe the simulation experiments that were conducted to test and to compare the proposed multivariate regression models. The results for these experiments, which we will present in section 7.5, show how the project control processes that use an estimated activity level schedule performance can approximate the performance of a process that requires the drilling down of the WBS by the project manager. We will draw overall conclusions in section 7.6. In addition, an appendix was added at the end of this paper. Appendices A.6 and A.7 presents the theory on kernel transformations and provides the mathematical formulations of the multivariate regression methods, as they were implemented in our research.

7.2   Literature

In this literature section, we will focus on the recent advances that have been proposed in the literature for top-down project schedule control using EVM/ES. The term top-down schedule control was formally defined by Vanhoucke (2011) [9]. He compared the application of the EVM/ES schedule performance measures, monitored at a high WBS level, with a bottom-up approach using schedule risk analysis (SRA [16]) metrics. His results showed that serial project network structures allow for a better top-down schedule control using EVM/ES, whereas parallel project networks benefit more from a bottom-up approach using SRA.

A combined approach for schedule control using EVM/ES was proposed by Elshaer (2013) [17]. More precisely, this author proposed to incorporate SRA information into the calculation of EVM/ES project duration forecasts. The EVM/ES project duration forecast formulae, calculated at a high WBS level, are adapted to accurately represent activity level SRA details. In addition, multiple other researchers recently proposed different approaches to improve final duration forecasts of the project using EVM/ES. These approaches range from the application of statistics and artificial intelligence methods [18–20] to newly developed variables to be

calculated in a EVM/ES system [3, 8].

These aforementioned authors focus on the dynamic forecast of the final duration during the execution of the project. Moreover, they are interested in the final outcome of the project, rather than the dynamic process of getting there. More precisely, they do not pay attention to the activity level performance that is the real driver of the project progress in a stochastic environment. Other researchers however, have effectively considered this activity level performance and have defined an "acceptable state of schedule control at the activity level" [10] that is inferred from signals produced at a high level of the WBS by EVM/ES performance measures. These signals arise from the application of statistical process control charts [11, 12, 21], statistical tolerance limits [10] or multivariate schedule control metrics [13]. A framework to develop and test such *statistical project control* methods and to interpret the produced warning signals was presented by Colin and Vanhoucke (2014) [10].

In a recent study that implements this framework [14], the focus is both on the top-down use of the EVM/ES system and the activity level schedule performance of the project. This study describes and tests different control points in the project schedule at which the EVM/ES measures can be applied for schedule control. In line with prior research findings, this study concludes that in order to be as efficient as possible, the project schedule control process requires the activity level schedule performance of the project to be known at each review period during the project's execution. More precisely, the LP method was found to outperform all other tested control approaches using EVM/ES. Lipke (2012) proposed the use of EVM/ES schedule performance measures on only those activities that lie on the dynamic longest path in the project's schedule during the execution. This method however, places a heavy burden on the project manager in terms of control effort, since at each review period a drill-down of the WBS is needed to obtain a detailed overview of the activity level performance and the longest path needs to be calculated. Effectively, this method should therefore be characterised as bottom-up, since it requires the activity level performance to be monitored dynamically.

In this paper, we expand on the findings of Colin and Vanhoucke (2014) [14]. We will also adopt the LP method, but instead of calculating the longest path in the project at each review period from the actual activity level schedule performance, we will use the EVM/ES schedule performance measures, observed at a high level of the WBS, to approximate the activity level performance. By using only high WBS level observations, we conjecture that the control effort is reduced, since a physical drill down of the WBS is no longer required for each review period in order to calculate the LP. The outline of this process is described along the following lines. For now, let the vector $\mathbf{d}$ represent the activity durations of a

project, for which $\mathbf{x}$ represents the EVM/ES observations, at a high level of the WBS. Using Monte Carlo simulations (MCS), we are able to generate a large number of fictitious executions of this project and consequently, this set of executions can be used to learn the statistical relation between $\mathbf{d}$ and $\mathbf{x}$. In this paper, we will deploy either of the multivariate regression methods (PCR, PLSR, kPCR or kPLSR) that were introduced earlier in order to learn the statistical relation. Given this statistical relation, we are then able to estimate $\widehat{\mathbf{d}}$ from $\mathbf{x}$, when the project is actually being executed. Ultimately this removes the necessity to drill down the WBS during the project, and the LP method can be used to produce warning signals for project schedule control, while requiring less effort.

7.3   Methodolgy

In this section, we will introduce the multivariate regression methods that we will apply to estimate the activity level schedule performance from a vector of EVM/ES observations. First, in section 7.3.1, we will introduce the EVM/ES schedule performance measures that should be observed periodically at a high level of the WBS in the project. Second, in section 7.3.2, we will discuss how these measures should be structured, in order to apply the multivariate regression methods. Finally, in section 7.3.3, we will introduce the four distinct multivariate regression methods.

7.3.1   EVM/ES data for top-down project schedule control

In a typical implementation of EVM/ES for top-down project schedule control [5, 9], aggregated performance measures are monitored periodically at a high level of the WBS and are calculated from the earned value (EV) and the planned value (PV). For these schedule performance measures, the actual schedule performance of the project (EV) is compared to the preliminary planning variable (PV). The EV represents the budgeted value of the work that is performed in a project since its initiation, whereas the PV represent the work that should have been performed if everything had gone according to the baseline schedule [22]. The schedule variance (SV=EV-PV) and the schedule performance index (SPI=EV/PV) can be directly calculated from the EV and PV and hence, are expressed in monetary values. The earned schedule (ES) translates the EV into time units and expresses the time since the beginning of the project at which the current EV should have been accrued if everything had gone according to the baseline schedule. When the ES is compared to the actual time (AT) since the beginning of the project, it allows the schedule variance using ES (SV(t)=ES-AT)

and the schedule performance index using ES (SPI(t)=ES/AT) to be calculated [4, 23].

In this paper, we will assume that this set of schedule performance measures {SV, SPI, SV(t), SPI(t)} can be reported at fixed percentage complete (PC = EV/BAC) points throughout the execution of the project [10, 13]. BAC denotes the budget at completion, i.e. the total budgeted value of the project. If we assume a total K of reporting periods during the execution of the project, we can denote the observations made at a certain PC point $\kappa \in 1 \ldots K$ as $\mathbf{x}_\kappa = $ [SV, SPI, SV(t), SPI(t)]. Consequently, the preliminary vector of all observations made up to the point κ can be denoted as $\mathbf{x}_{1\ldots\kappa}$ while the vector of all observations along the lifetime of the project can be denoted as $\mathbf{x} = \mathbf{x}_{1\ldots K}$. A similar approach was followed in Colin and Vanhoucke (2014) [10].

7.3.2   The multivariate regression process

In this section we will introduce the necessary background and notations in order to formalise the multivariate regression process that allows us to estimate the activity level schedule performance of a project from the EVM/ES observations. First, we consider the vector of the real activity durations $\mathbf{d}$ in a project. These real activity duration will, through the calculations of the EVM/ES system, give rise to a vector of observations $\mathbf{x}$. Consequently, in a real life project, the vector $\mathbf{x}$ will therefore be an expression of the underlying activity durations $\mathbf{d}$. In the statistical analysis, we interchange the position of the two variables and say that $\mathbf{d}$ can be estimated from $\mathbf{x}$. To comply with the standard notation in statistical regression, we will denote the vector of activity durations $\mathbf{d}$ as $\mathbf{y}$ and hence, our model can be expressed as $\mathbf{y} \sim \mathbf{x}$. At best, we can produce estimates and therefore, $\hat{\mathbf{y}}$ will be used to denote the estimate for the vector of activity durations, based on the multivariate observation $\mathbf{x}$. Moreover, since a project control procedure should be applied dynamically during the execution of a project, we will denote the estimate of the vector of activity durations $\hat{\mathbf{y}}_\kappa$, that is based on $\mathbf{x}_{1\ldots\kappa}$, i.e. the EVM/ES observations in a project up to a reporting period κ.

In order to learn the statistical relation between $\mathbf{y}$ and $\mathbf{x}$, we need to obtain data to which the multivariate regression methods can be applied. In a control procedures for a process in manufacturing or services, two matrices X and Y would be recorded during the on-going processes under "normal operation conditions" [24], where for different $\mathbf{x}$ a corresponding $\mathbf{y}$ can be found. However, since a project is a one-time endeavour with a unique start and finish, observations cannot be made continuously during operations and building a reference X and Y is completely different. The possibility of building a reference from historical data of similar projects

has been proposed in the literature [11, 21]. However, we prefer the use of a two-phased MCS model to produce reference matrices X and Y for a project under controlled conditions [10].

Phase 1: Data collection
Let a first phase MCS produce n fictitious executions of a given project, where the deviations from the baseline schedule are captured using EVM/ES schedule performance measures at a high level of the WBS. The matrix X then consists of n observations for the multivariate variable $\mathbf{x}$. More concretely, each row of X is then formed by the $1 \times 4K$ vector $\mathbf{x}$ for a different fictitious execution of that project. X is therefore a $n \times 4K$ matrix, at the end of the simulated project runs. Correspondingly, if the project contains N activities, the $(n \times N)$ matrix Y is then composed of the n $(1 \times N)$ vectors of activity durations $\mathbf{y}$, corresponding to each fictitious execution of the project.

Phase 2: Real project progress
The multivariate regression models are used to learn the statistical relation between $\mathbf{y}$ and $\mathbf{x}$ from the reference matrices X and Y, produced in the first phase MCS for a project. In the actual execution of the project, these methods can then be used to produce an estimate for the activity durations $\hat{\mathbf{y}}_\kappa^*$ at a reporting period κ, based on the observed vector of EVM/ES schedule performance measures $\mathbf{x}_{1\ldots\kappa}^*$. In the computational experiment, presented in section 7.4, this execution will also be fictitious, as generated by the second phase MCS. We will provide details on the two phases in the simulation experiment in section 7.4.3. However, in a real-life application of these methods for a controlled project, only the first phase MCS would have to be performed. Consequently, $\mathbf{x}_{1\ldots\kappa}^*$ would then represent the practically observed EVM/ES schedule performance measures and $\hat{\mathbf{y}}_\kappa^*$ would represent an estimate of the real activity level schedule performance for the controlled project. This estimate could then be used to calculate the LP method in order to produce warning signals for the control of the schedule of that project.

In order to avoid the problems associated with missing and incomplete observations [13, 25] in multivariate statistical regression, K models will be build for each multivariate regression methods. Therefore, at a review period κ, only the first 4κ columns of X will be used to build a regression model to estimate $\hat{\mathbf{y}}_\kappa^*$ from $\mathbf{x}_{1\ldots\kappa}^*$.

7.3.3 Multivariate regression models

In this section, we will introduce the multivariate regression methods that will be applied to learn the statistical relation between **y** and **x** from the Y and X matrices, generated in the first phase MCS. First, we will discuss their structure and properties in section 7.3.3. Second, we will describe them individually and pay special attention to the parameters that influence their performance in section 7.3.3.

Structure and properties

In this section, we discuss the multivariate regression methods with respect to their structure and properties. Two main aspects of multivariate regression are therefore considered. First, we will discuss *collinearities and noise* and we will proceed with a discussion of *non-linearities*.

Collinearities and noise
In multivariate regression from a multivariate variable **x** towards a multivariate variable **y**, it is indispensable that in some form the inverse of the correlation matrix $X^T X$ is calculated [26]. This inverse, however is often badly defined due to the collinearities between the columns of matrix X. As a result, the margin of error on the estimates obtained from this inverse is unacceptable. For our project control application, it is a reasonable assumption that some columns of X are strongly correlated, since the EVM/ES performance measures are likely influenced by only a limited number of activities [13]. Consequently, if inference is made from the multivariate variable X, the existing collinearities should be addressed [27].

A solution for this problem has been proposed in the principal component regression (PCR) and partial least squares regression (PLSR) methods. Both methods have been successfully applied for the on-line control of batch processes [24, 28–30]. In these methods, the dimensionality of the problem is reduced by the projection of either the matrix X or the matrix Y or both onto a lower-dimensional space. The new coordinate base for this space in PCR and PLSR is always chosen to be perpendicular and therefore, in the projections of the data onto this space, collinearities are removed. Consequently, the matrix inversion is then computationally more stable and moreover, undesirable noise is removed from the measurement system [24].

Non-linearities
In addition to the application of PCR an PLSR to remove collinearities and

noise from the system, we also consider the non-linear characteristics of the EVM/ES system.

From a detailed study of the EVM/ES calculations that are required to produce the schedule performance measures, it can be seen that this transformation is unlikely to be linear with respect to the real activity durations. The reader can verify this by calculating the SPI(t) for a single activity project over the course of its lifetime, for example. Moreover, the non-linear character increases when the deviation between the baseline estimate duration (used in the calculation of the PV) and the real activity duration increases. Since both PCR and PLSR were originally developed to model linear relations between the input space X and the output space Y, they might not be suited to estimate activity durations from a vector of EVM/ES observations. However, kernel algorithms were proposed for both the PCR and the PLSR method in the literature. Kernel principal component regression (kPCR, [31]) and kernel partial least squares regression (kPLSR, [32]) implement a kernel transformation of the input space into a higher dimensional feature space. Although the dimensionality of the problem is thereby (artificially) increased, the properties of the new coordinate base can still hold and redundancies and noise are not re-introduced [31]. Moreover, the linear relations that are modelled in this higher dimensional space correspond to non-linear regression models in the input space. Thereby, it is conjectured that the non-linear character of the EVM/ES measurement system is more correctly accounted for. Conceptually, the kernel algorithms (kPCR and kPLSR) are not too different from their linear counterparts, except for that they start from the Kernel Gram matrix G instead of from the matrix X in the input space. In order to obtain this $n \times n$ matrix G, all input points should first be mapped into the higher dimensional space, defined by the kernel transformation function. In our research, we only considered a polynomial and radial base function (rbf) transformation [20, 31]. After the mapping (or transformation) of all points in the input space, the matrix G can then be constructed from the cross dot products between all mapped input points. The reader is referred to the appendix A.6 for more details concerning our choice for the polynomial and rbf kernel functions and the construction of the matrix G.

Description and parameters

In this section, we will describe the multivariate regression methods that were implemented in our research. We start with the PCR and PCR-PC, where the latter can be considered as a straightforward extension of the former. Subsequently, the kPCR method is presented. These methods are based on principal components analyses, and are characterised by a separate modelling of the outer relations (the data structure of X or G and

Y) and the inner relations (dependencies between Y and the input space). The multivariate regression techniques, which are based on partial least squares (PLSR and kPLSR) combine these two steps in one integrated modelling of both the input space (X or G) and the output space (Y). We will discuss them in the subsequent paragraphs. Finally, we will present a table to provide an overview of these methods and to review the parameters deployed in each.

In order not to aggravate this paper with too many mathematical formulations, we have added a more detailed discussion of the presented multivariate methods in the appendix A.7 to this paper. We implemented the algorithms in these appendices in the statistical programming language R [33], in order to apply them for project schedule control.

PCR and PCR-PC
In this paragraph, we will discuss the PCR and PCR-PC regression models sequentially. In these multivariate regression models, which are based on principal components, the outer and inner relation are modelled separately.

In principal component regression (PCR), the structure of the input space (X) is analysed using a principal component decomposition. This decomposition was applied in a previous study for project control [13] in order to derive multivariate control limits. From the principal component decomposition of X, a total of $4K$ principal components can be found, where K denotes the number of reporting periods. However, in order to remove the effects of collinearity in the data and to remove noise from the system, only the first l principal components should be used. It is therefore conjectured that a number l exists for which all valuable information is contained within the first l principal components of a matrix. Consequently, in PCR, the inner relation between the columns of Y (i.e. the activity durations) and the first l principal components is modelled using multiple linear regression (MLR, [26]). The resulting regression coefficients can then be used to estimate the activity durations for a new project execution from the vector of EVM/ES observations, recorded at a high level of the WBS.

In PCR-PC the same approach is followed, although first an additional principal component decomposition is performed for the output space (Y). As activity durations in practice can be assumed to contain dependencies [34], it is a logical consequence that collinearities and noise can also be removed from the matrix Y. Again, we conjecture that a natural number (m) can be found for which the first m principal components of Y contain all valuable information. The inner relation is then modelled by performing a MLR between the first l principal components of X and the first m principle components of Y. Consequently, for a new project execution, this

regression can be used to estimate the durations of the activities from the vector of EVM/ES observations that are recorded at a high level of the WBS.

kPCR

The kernel variant for PCR (kPCR) exploits the use of kernel transformations in order to deal with non-linear dependencies between the input space and the output space. This is done by first decomposing the matrix G into its principal components. This time, a total of n such kernel principal component can be found. However, we again conjecture that a number l exists for which all of the valuable information is contained within the first l kernel principal components. The inner relation between Y and these first l kernel principal components can then be modelled using MLR. This allows to estimate activity durations of a new project execution from a vector of EVM/ES observations, recorded at a high level of the WBS.

The mapping of the data points in the input space into a higher dimensional feature space is done through a polynomial or rbf kernel. The mathematical formulation for these can be found in appendix A.6. Here, we will only emphasize the parameters that can be varied for these functions in order to find an adequate transformation to be applied in our project control application. The polynomial kernel function takes a parameter d, which corresponds to the order of the polynomial that is used in the transformation. The reader can verify that when $d = 1$, the polynomial kernel function coincides with a linear transformation [31]. The rbf kernel function takes a scaling parameter c, for which an intuitive explanation cannot be easily given. Therefore, the reader is referred to Vapnik (2000) [35] for more information concerning the rbf kernel.

PLSR

In the principal component-related multivariate regression methods that were just presented, the outer relations (the principal components of X or G and Y) and the inner relations (MLR) are modelled separately. In partial least squares regression however, these steps are performed in an integrated manner.

In PLSR, both the X and Y matrices are decomposed simultaneously. This decomposition results in latent variables, instead of in principal components. The reader is referred to Rosipal and Trejo (2002) [32] and the appendix A.7 for more details concerning the latent variables. For now, it suffices to note that in order to remove the effects of collinearities and to remove noise from the system, only those latent variables that contain valuable information should be retained in the analysis. We will therefore conjecture that a number l exist for which all relevant information from

the system is contained within the first l latent variables. Consequently, in our project control application, we will use these latent variables to estimate the activity durations, of a new project execution, from the EVM/ES variables that are observed at a high level of the WBS.

kPLSR

The kPLSR method is the kernel variant to the PLSR method. In kPLSR, the partial least squares algorithm is applied to the Kernel Gramm matrix G, in order to calculate kernel latent variables. Again, we will conjecture that a number l exists for which all valuable information is contained within the first l kernel latent variables and that from these, the activity durations for a new project execution can be estimated, based on the vector of EVM/ES observations that are recorded at a high level of the WBS. The kernel Gramm matrix will be produced using the polynomial and rbf kernel functions, which respectively take d and c as a parameter.

Overview

Table 7.1 provides an overview of the presented multivariate regression methods and reviews the parameters that are deployed in each. The columns of table 7.1 respectively present the name of the method, how the outer and inner relation are modelled, whether a kernel function is applied and which parameters should be considered. As previously described, the principal component-related methods (PCR, PCR-PC and kPCR) contain a separate modelling of the outer relation and the inner relation. The partial least squares-related methods (PLSR and kPLSR) perform these two steps in an integrated manner.

7.4 Computational experiment

In this section, we describe the computational experiment that was carried out in order to test the multivariate regression methods for project control, when used in combination with the LP method. First, in section 7.4.1 we will discuss the dynamic project progress model and the dataset of project networks that was deployed in our experiment. In section 7.4.2, we will introduce the activity duration distributions modelling that serves as input to our simulations. In section 7.4.3, the two phases for the MCS model will be described. Finally, we will present the parameter settings in section 7.4.4 that were used to test both the *general performance* and the *robustness* of the multivariate regression methods.

7.4.1   Dynamic project progress model

We deploy the dataset of Vanhoucke (2011) [9] in order to ensure a maximal diversity in terms of network structures within the project that were used. This diversity allows us to produce results that are as generalisable as possible. The 900 project networks in this dataset were generated using RanGen [36] and contain 30 activities with baseline estimate durations (expressed in working hours) that were uniformly sampled between 8 and 56. The project progress modelling in the simulations that were carried out for this research was implemented in the statistical programming language R [33]. The activity durations in a project serve as input for these project progress situations. In the subsequent section, we will describe in more detail how these activity durations are generated. For now, it suffices to note that, for each fictitious project execution, two project schedules were compared. The first is the baseline schedule that was calculated from the baseline estimates for the activity durations and which acts as a reference for all simulated project executions. The second schedule is calculated using sampled durations for all activities. From these two schedules, the EVM/ES data was computed based on the assumption that both the PV and EV for an activity accrue linearly during its execution. Moreover, both the fictitious project executions and the resulting EVM/ES data were validated using P2 Engine [37], a scripting-enabled project scheduling and control tool that was previously used for project control studies [10, 14, 20].

7.4.2   Activity duration distributions modelling

In order to produce reliable and generalisable results from a simulation experiment, it is required that the input to this experiment is given careful consideration. We have implemented the combined input modelling for activity durations that was previously applied by Colin and Vanhoucke (2014) [10, 14]. The uncertainty at the activity level of the project is modelled using an expression of both *variation* and *risk* in the duration of the activities of the project. The reader should consult the references for more details on these two aspects of the modelling of uncertainty. For the purpose of this paper, it suffices to note that for both of these aspects, a separate probability density function (pdf) should be chosen. The generalised beta distribution [38] is an appropriate choice for both [14]. We chose to let a parameter vector $\boldsymbol{\omega} = (a, b, m, \mu)$ characterise the pdf for a random variable that is distributed according to the generalised beta distribution ($\beta(\boldsymbol{\omega})$). Here, a is used to denote the minimum, b is used to denote the maximum, m represents the mode and μ represents the mean. Colin and Vanhoucke (2014) [14] argue that these values should all be expressed as a fraction

of the baseline estimate duration $\widehat{d_i}$ for an activity $i \in 1 \ldots N$, in order to represent the pdf of a random duration of that activity as $\beta(\boldsymbol{\omega})\widehat{d_i}$.

As discussed previously, we should assign two different pdf's to model variation and risk. Let $\boldsymbol{\omega}_V$ and $\boldsymbol{\omega}_R$ characterise these respective pdf's. According to Colin and Vanhoucke (2014) [14], the procedure to draw a Monte Carlo sample for a fictitious project execution can then be described as follows.

- *Risk:* First, a bias term B should be drawn from a generalised beta distribution $\beta(\boldsymbol{\omega}_R)$. This bias term can easily be perceived as a consistent over- or underestimation of activity durations, or a project-wide effect of uncertain events. The concept was introduced by Trietsch et al. (2012)[34], when they considered linear association for project uncertainty modelling.

- *Variation:* Subsequently, the variation should be taken into consideration and for each activity $i \in 1 \ldots N$ in the fictitious project execution, an activity duration d_i should be drawn from the distribution $\beta(\boldsymbol{\omega}_V)B\widehat{d_i}$. Moreover, this sample can then be seen as if it was drawn from a generalised beta distribution $\beta(\boldsymbol{\omega}_V)$ with a independently chosen parameter vector that represents the variation $\boldsymbol{\omega}_V$, which is applied to a scaled baseline estimate $(B\widehat{d_i})$ of the real activity duration.

7.4.3   Two-phased simulation model

The simulation model that has been used in this paper in order to compare the multivariate regression methods constitutes two phases. In the *first phase*, a MCS is run in order to produce the reference matrices X and Y from which the statistical relation between the EVM/ES schedule performance metrics (**x**) and the activity level schedule performance (i.e. the activity durations **y**) can be learned. Consequently, the *second phase* is then used to calculate the performance of the different multivariate regression methods in combination with the LP method. We will describe these two phases in more detail along the following paragraphs. It is important to note that the two phases were performed for each of the project networks in the RanGen dataset. In other words, for a single project network, both phases are always performed. In section 7.4.4 we will provide the parameter vectors $\boldsymbol{\omega}$ that were used in the two phases for different simulation experiments.

Phase 1
In the first phase MCS, we generated fictitious project executions for which

both **x** and **y** were recorded. These respectively make up the reference matrices X and Y from which the statistical relation can be learned by the multivariate regression methods. Since the focus of the LP method is both on the projection of the project makespan (i.e. the total duration of the execution of the project's schedule, real duration (RD)) as well as on detecting disruptive events in the activity level performance of the project [14], we decided on a deadline for the project schedule to be completed. This deadline was chosen, such that a fixed percentage (30%) overrun of the deterministic duration of the schedule (planned duration, PD) was allowed. This planned duration was determined by the critical path method for all projects in the dataset by using the provided baseline estimate durations $\widehat{d_i}, \forall i \in 1 \ldots N$. Since the reference matrices should contain only those runs that have a desirable outcome [24], we included only those fictitious project executions in X and Y that have a makespan that is lower than the imposed deadline (RD $\leq$ (1+30%)PD).

Phase 2

In the second phase MCS, we generated another set of fictitious project executions. These executions were used to emulate the practical application of the multivariate regression methods. Thus, for each of those fictitious project executions, the EVM/ES schedule performance metrics ($\mathbf{x}^*$) were used to form an estimate of the activity durations ($\widehat{\mathbf{y}^*}$), based on the X and Y reference matrices that were generated in the first phase. As mentioned earlier, this was done dynamically, during project progress, at K review periods. K was chosen to be equal to 19 in this study, so that the progress of a project is monitored when it is $5\%, 10\%, \ldots$ and 95% complete. Consequently, at each review period $\kappa \in 1 \ldots 19$, the LP method was then used in order to produce warning signals for project schedule control. A warning signal is said to be false when the LP method indicates that the project deadline will not be met although at the end of the project the deadline was not exceeded (RD $\leq$ (1+30%)PD) for that fictitious project execution. A warning signal is said to be true when the real duration exceeds the deadline (RD $>$ (1+30%)PD) at the end of the project. From the collection of all fictitious project executions that produce a false signal, the *probability of overreactions* was calculated. From the collection of all fictitious project executions that produce a true signal, we calculated the *detection performance*. We refer the reader to Colin and Vanhoucke (2014) [10] for additional details on these performance measures for project control. The probability of overreactions and the detection performance were combined into a single measure: the *area under the curve*(AUC). The AUC is a well-known performance measure for classification testing and is extensively used in the literature on machine learning [26]. A completely randomised classification procedure is most likely to result in an AUC of 0.5. For better classification procedures AUC values in between 0.5 and 1 can be found. An AUC of

1 would mean that the detection performance is equal to 100%, while the probability of overreactions can be kept at 0%. For a project control procedure, this would mean that only signals are generated when the project will not meet the imposed deadline, while an execution that finishes before this deadline will never trigger a warning signal.

7.4.4 Parameter vectors

In this section, we will discuss which parameter vectors were used in the different simulation experiments that were performed for this study. Table 7.2 shows the 14 parameter vectors ω that we used in the experiments. We will discuss their use in the *general performance* and *robustness* experiments.

General performance experiment
The left side of table 7.2 presents three parameter vectors ω_1, ω_2 and ω_3 that were used to test the *general performance* of the multivariate regression methods when they are used in combination with the LP method. They are respectively used to represent durations that, on average, are finished early, on-time or late. These parameter vectors allowed us to define three different scenarios.

- Early: This scenario is used to record the performance of the multivariate regression methods when, on average, activities are finished early. In the first phase of the experiment we generated 1,000 fictitious project executions for which we assigned either ω_1 or ω_2 to either ω_V or ω_R. More precisely, half of the runs in the first phase (500) were generated using the combination $(\omega_R, \omega_V)=(\omega_1, \omega_2)$, while the other 500 runs were generated using the combination $(\omega_R, \omega_V)=(\omega_2, \omega_1)$. The 1,000 runs for the second phase were also generated using the same combinations.

- On time: This scenario is used to record the performance of the multivariate regression methods when, on average, activities are finished on time. The 1,000 fictitious project executions for both the fist and second phase of the MCS were generated by using $\omega_R = \omega_V = \omega_2$

- Late: This scenario is used to record the performance of the multivariate regression methods when, on average, activities are finished late. Half of the runs in the first phase (500) were generated using the combination $(\omega_R, \omega_V)=(\omega_2, \omega_3)$ while the other 500 runs were generated using the combination $(\omega_R, \omega_V)=(\omega_3, \omega_2)$. The 1,000 runs for the second phase were also generated using the same combinations.

Robustness experiment

In this paper, we use the outcome of a first phase MCS in order to learn the statistical relation between the EVM/ES schedule performance measures, observed at a high level of the WBS, and the activity level performance. In a practical setting however, it might not always be possible to provide the distributional input to this first phase MCS which exactly corresponds to the stochastic nature of the activity durations during the execution of the project in real life. Therefore, we conducted a robustness test in order to test the sensitivity of the multivariate regression methods. The sensitivity with respect to over- or underestimations of the *mean* and *standard deviation* of the underlying activity distributions were focussed on. On the right side of table 7.2, we present two columns with respectively five and six parameter vectors. The first columns contains five parameter vectors ($\omega_{2\mu_1}$, $\omega_{2\mu_2}$, $\omega_{2\mu_3}$, $\omega_{2\mu_4}$, $\omega_{2\mu_5}$) that were derived from ω_2, but have a mean μ that was changed. The second columns contains six parameter vectors (ω_{2s_1}, ω_{2s_2}, ω_{2s_3}, ω_{2s_4}, ω_{2s_5}, ω_{2s_6}), for which the pdf has a standard deviation (σ) that was varied gradually, compared to that of ω_2. We will discuss how these parameter vectors were used to test the sensitivity with respect to the mean and standard deviation along the following lines.

- Mean: In order to test the sensitivity with respect to an over- or underestimation of the mean of the activity durations distribution, we first generated 1,000 runs in the first phase MCS by using $\omega_R = \omega_V = \omega_2$. Consequently, we performed the second phase MCS five times to generate 1,000 runs each, with respective means of the underlying activity durations: 0.7, 1, 1.3, 1.6 and 1.9. For $j \in 1 \ldots 5$, 500 runs were produced using the parameter vector combination (ω_R, ω_V)=(ω_2, $\omega_{2\mu_j}$), while the other 500 runs were generated by using (ω_R, ω_V)=($\omega_{2\mu_j}$, ω_2). In other words, for $j = 1$, an overestimation of the mean of the underlying activity durations distribution was made in the first phase MCS. For $j = 2$, the mean of the underlying activity duration distribution was correctly estimated and for $j \in 3 \ldots 5$, the mean was underestimated in the first phase MCS.

- Standard deviation: The same procedure was followed in order to test the sensitivity with respect to the standard deviation of the activity durations distribution. After generating 1,000 runs for the first phase MCS by using $\omega_R = \omega_V = \omega_2$, we performed the second phase MCS six times with respective standard deviations of the underlying activity durations: 0.5, 0.42, 0.35, 0.28, 0.20 and 0.15. For $j \in 1 \ldots 6$, 500 runs were produced using the parameter vector combination (ω_R, ω_V)=(ω_2, ω_{2s_j}), while the other 500 runs were generated by using (ω_R, ω_V)=(ω_{2s_j}, ω_2). $j \in 1 \ldots 3$ correspond to an underestimation of the standard deviation of the activity durations distribution, while $j \in 4 \ldots 6$ correspond to an overestimation.

7.5 Results and discussion

In this section, we will discuss the AUC values that were recorded for the project control procedures that combine a multivariate regression method with the LP method. In section 7.5.1, we will present the results from the *general performance experiment* and in section 7.5.2, the results from the *robustness experiment* will be discussed. In order to show results, some parameters had to be optimised for the multivariate regression methods. Since a full discussion of the influence of these parameters could not be included within the scope of this paper, we show only those values that performed best, in table 7.3. More precisely, the values of l (the number of latent variables in X), m (the number of latent variables in Y), d (the power in the polynomial kernel function) and c (the scaling parameter in the rbf kernel function) are shown for all multivariate regression methods, if applicable. We used "-" to denote that a certain parameter does not apply to a certain method. The combination of parameters that is presented for each multivariate regression method was found to give the overall best median AUC value when used in combination with the LP method, on the 900 project networks dataset. In addition, the ranges of values that were tested in our experiments are presented in the footnotes in table 7.3. The parameter values in table 7.3 show that only a few principal components or latent variables (l and m) should be incorporated into the analysis for project control. This observations confirms the findings of Colin and Vanhoucke (2014) [13] with respect to the use of principal components in an EVM/ES measurement system, when applied for project schedule control. The values for the parameters of the kernel functions are not as easily validated, since there was little guidance to be found in the literature on how to determine these [20]. However, at least to an order of magnitude, the values for the polynomial kernel and rbf kernel function parameters correspond to those that were used by Wauters and Vanhoucke (2014) for project duration and cost forecasting [20].

7.5.1 General performance experiment

Figure 7.2 shows the calculated AUC values for the multivariate regression methods when they are used in combination with the LP method. The three scenarios of section 7.4.4 are depicted along the x-axis with the combination of the parameter vectors that are used in both phases of the MCS. "LP" is used in figure 7.2 to denote the performance of the LP method when at each review period, during the execution of the project, a drill-down of the WBS is performed in order to retrieve the exact activity level performance. This AUC value is uniformly equal to 1, for the three tested scenarios. This indicates that, if the drill-down of the WBS is performed

General performance
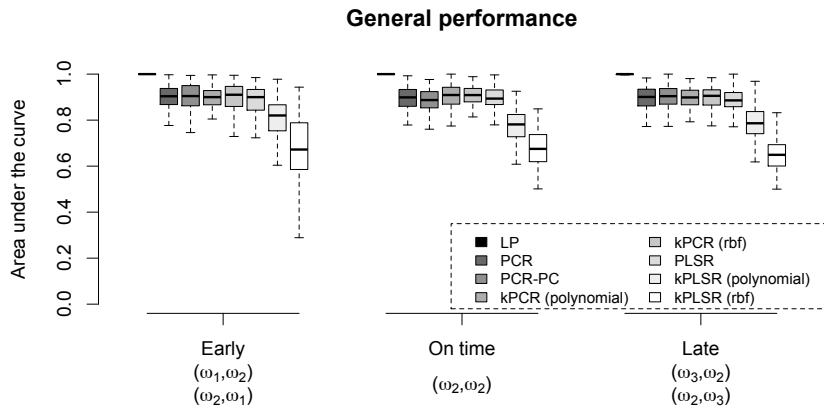
Figure 7.2: The calculated AUC values for the general performance experiment

periodically, only true warning signals will be produced from the dynamic longest path calculations. However, the intent of this paper was to reduce the project control effort by not requiring the drill-down of the WBS at each review period. Therefore, in total, seven multivariate regression methods were used to estimate the activity level performance from EVM/ES schedule performance metrics, observed at a high level of the WBS. The results that are shown were produced with the parameter settings that were presented in table 7.3.

We observe that in general, the use of the multivariate regression methods does not perform as well as a physical drill-down of the WBS. The AUC values are consistently lower than 1. Moreover, the performance seams to be slightly better in the *early* scenario than in the two other scenarios, although no real large differences can be observed. Figure 7.2 also shows that, among the multivariate regression methods, some differences can be observed. Considering the principal component-based regression methods, we note that the performance improves with increasing complexity of the statistical model. For PCR, a median AUC of around 0.9 was calculated. This median performance slightly improves when a second principal component analysis is performed for the Y matrix (PCR-PC) or when a kernel transformation is applied (kPCR). More precisely, the most prominent improvement in the AUC is observed when the rbf kernel (kPCR (rbf)) is applied to the observations in the input space (X) in order to produce the matrix G. For the partial least squares-related methods, we cannot observe this trend. Rather, the performance of the PLSR method seems to decrease drastically when kernel transformations are applied prior to the calculation of the latent variables.

As a summary from figure 7.2, we can note that the kPCR method in combination with the rbf kernel function should be used when the project manager wants to estimate the activity level performance from EVM/ES schedule performance measures. This results in a project schedule control procedure that has an AUC that is only slightly lower than when a physical drill-down of the WBS is performed periodically.

7.5.2 Robustness experiment

Figure 7.3 shows the calculated AUC values in the robustness experiment. The reader should recall that the first phase MCS, that generated the reference matrices from which the statistical relation could be learned, was performed with $\omega_R = \omega_V = \omega_2$. On the left of figure 7.3, the results for the experiment that tested the sensitivity with respect to the mean of the activity durations distribution are shown. The x-axis depicts the five parameter vectors $(\omega_{2\mu_1}, \omega_{2\mu_2}, \omega_{2\mu_3}, \omega_{2\mu_4}, \omega_{2\mu_5})$ that were combined with ω_2 in the second phase MCS. On the right of figure 7.3, the results for the experiment that tested the sensitivity with respect to the standard deviation of the activity durations distribution are shown. On the right x-axis, the six parameter vectors $(\omega_{2s_1}, \omega_{2s_2}, \omega_{2s_3}, \omega_{2s_4}, \omega_{2s_5}, \omega_{2s_6})$ are shown, which were combined with ω_2 in the second phase MCS.

Figure 7.3 shows how all of the multivariate regression methods, that are proposed in this paper, are quite robust to over- or underestimations of the mean and standard deviation of the activity durations distribution. Since, for both the mean and standard deviation, no obvious trends can be detected in figure 7.3 and overall, the calculated AUC values are to a great extent in accordance with those presented in the general performance experiment. We conjecture that this robustness is largely due to the use of only the first principal components ($l = 1$, $m = 1$) in the regression formulae. These first principal components can be considered as an expression of a general trend, rather than as an expression of a detailed characteristic of the activity level performance [13]. Therefore, its use in regression models is robust with respect to changes in a characteristic of the activity level performance. This conjecture was confirmed by additional tests that were done when more principal components were included. The kPLSR method using a polynomial kernel and rbf kernel function respectively deploy four ($l = 4$) and two ($l = 2$) latent variables in order to estimate the activity level performance, and their performance is shown, in figure 7.3, to be slightly more sensitive to over- or underestimations of the mean and standard deviation.
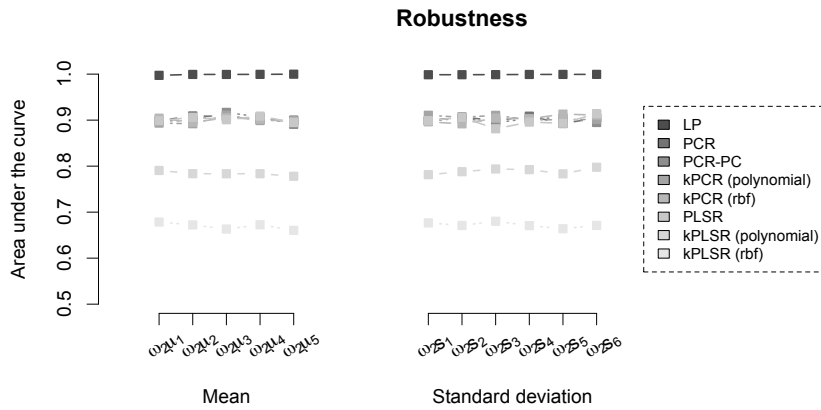
Robustness

Figure 7.3: The calculated AUC values for the robustness experiment

7.6 Conclusions

In this paper, we explored the use of multivariate regression methods to estimate the activity level performance from EVM/ES schedule control metrics, that are observed at a high level of the WBS. This estimated activity level performance could then be used in dynamic longest path calculations in order to control the schedule of an ongoing project. By using the statistical relation between EVM/ES observations and the underlying activity durations in a project, the drill-down of the WBS by the project manager could be replaced, which should effectively reduce the effort that is spent in the project control process. The kPCR method in combination with a rbf kernel (with the scaling parameter $c = 10^3$) is shown to outperform all other tested multivariate regression methods. Moreover, the use of only the first latent variable makes this regression method robust to over- or underestimates of the mean and standard deviation that are used in the MCS that generates the reference matrices, from which the statistical relation is learned.

The research method that was applied in this study will be given some consideration along the following lines. The two main strengths of our research methodology should be correctly identified. First, the computational experiment that was performed in this study is in line with recent literature. Both the datatset of projects as well as the dynamic project progress simulation modelling was validated in recent papers on project management. Second, for the first time, an explicit top-down inference process to obtain the activity level performance from EVM/ES observations was implemented. Moreover, we explored two categories of multi-

variate regression methods that have received a lot of attention from academics. Principal component-based and partial least squares-based regression methods were tested and compared for use in project control. The weaknesses of the research methods, with which this study was conducted, are also presented in this paragraph. In order to allow a practical implementation of the proposed multivariate regression methods, historical data should be available to which the input of the MCS should be calibrated. This data should therefore be interpreted using the uncertainty modelling that we used in this paper. In addition, a minimum of theoretical knowledge on both project scheduling theory and multivariate regression methods should be present in the organization, in order to periodically calculate the longest path from the estimated activity durations.

Our contribution to the theory on project management and control is threefold. First, we presented a theoretical basis to the problem of estimating the activity level performance in a project execution from EVM/ES observations, that are made at a high level of the WBS. Second, we have implemented a total of seven distinct multivariate regression methods (PCR, PCR-PC, kPCR (polynomial kernel), kPCR (rbf kernel), PLSR, kPLSR (polynomial kernel) and kPLSR (rbf kernel)) which were tested in an extensive computational experiment and for which the parameters were optimised. Moreover, this study also presents a practical use for the estimation of the activity durations from high WBS observations, since we combined them with the LP method in order to produce warning signals for project schedule control.

The practical implications of our study should also be given some thoughts. The use of the proposed multivariate regression methods (especially the kPCR (rbf kernel) method) in combination with the LP method allow a project manager to control the execution of a project, without requiring a periodic drill-down of the WBS. This reduces the effort that is spent in the project control process, while only $\sim 10\%$ is lost in terms of performance (AUC).

In order to draw further conclusions on the practical implications from this study, we advise that future studies expand on the research that is presented in this paper. More advanced artificial intelligence and machine learning method could be applied to the problem discussed in this paper, building on the theoretical basis that is provided. Moreover, a case study could be implemented to test whether our research findings still hold in a more realistic project environment. The influence of the industry-specific characteristics, cost distributions for the activities or the number of activities in a project could then be explored.

Method	Outer relation X/G	Outer relation Y	Inner relation	Kernel function	Parameters	
PCR	PC	-	MLR	-	l	number of principal components in X
PCR-PC	PC	PC	MLR	-	l	number of principal components in X
					m	number of principal components in Y
kPCR	PC	-	MLR	polynomial	l	number of kernel principal components in G
					d	Order of the polynomial used in the kernel transformation
				rbf	c	Scaling parameter used in the rbf kernel transformation
PLSR	Integrated		Integrated		l	Number of latent variables from X and Y
kPLSR	Integrated		Integrated	polynomial	l	Number of kernel latent variables from G and Y
					d	Order of the polynomial used in the kernel transformation
				rbf	c	Scaling parameter used in the rbf kernel transformation

Table 7.1: Overview of the multivariate regression methods and the parameters that are used in these methods.

General performance experiment

ω	(a, b, m, μ)	σ
ω_1	$(0.1, 1.2, 0.7, 0.6)$	0.38
ω_2	$(0.2, 4, 0.9, 1)$	0.30
ω_3	$(0.9, 4, 1.3, 1.4)$	0.38

Robustness experiment

Mean			Standard deviation		
ω	(a, b, m, μ)	σ	ω	(a, b, m, μ)	σ
$\omega_{2\mu_1}$	$(0.2, 4, 0.51, 0.7)$	0.3	ω_{2s_1}	$(0.2, 4, 0.40, 1)$	0.50
$\omega_{2\mu_2}$	$(0.2, 4, 0.90, 1.0)$	0.3	ω_{2s_2}	$(0.2, 4, 0.70, 1)$	0.42
$\omega_{2\mu_3}$	$(0.2, 4, 1.22, 1.3)$	0.3	ω_{2s_3}	$(0.2, 4, 0.85, 1)$	0.35
$\omega_{2\mu_4}$	$(0.2, 4, 1.57, 1.6)$	0.3	ω_{2s_4}	$(0.2, 4, 0.92, 1)$	0.28
$\omega_{2\mu_5}$	$(0.2, 4, 1.89, 1.9)$	0.3	ω_{2s_5}	$(0.2, 4, 0.96, 1)$	0.20
			ω_{2s_6}	$(0.2, 4, 0.98, 1)$	0.15

Table 7.2: Parameter vectors for the generalised beta distribution

Table 7.3: Parameters for the multivariate regression methods that produce the best overall AUC values when used in combination with the LP method for project schedule control

Method	l [*]	m [†]	d [‡]	c [§]
PCR	1	-	-	-
PCR-PC	1	1	-	-
kPCR (polynomial kernel)	1	-	7	-
kPCR (rbf kernel)	1	-	-	10^3
PLSR	1	-	-	-
kPLSR (polynomial kernel)	4	-	7	-
kPLSR (polynomial kernel)	2	-	-	10^{18}

[*]: from $l \in 1 \ldots 4K$ for PCR, PCR-PC and PLSR, with $K = 19$
and from $l \in 1 \ldots n$ for kPCR and kPLSR, with n the number of project executions for which RD < (1+30%)PD

[†]: from $m \in 1 \ldots N$, with $N = 30$

[‡]: from $d \in 1 \ldots 10$ [20]

[§]: from $c \in \{10^{24}, 10^{21}, \ldots, 10^{-3}\}$ [20]

References

[1] S. Hartmann, D. Briskorn, A survey of variants and extensions of the resource-constrained project scheduling problem, European Journal of Operational Research 207 (2010) 1–15.

[2] M. Plaza, O. Turetken, A model-based DSS for integrating the impact of learning in project control, Decision Support System 47 (2009) 488–499.

[3] H. Khamooshi, H. Golafshani, EDM: Earned duration management, a new approach to schedule performance management and measurement, International Journal of Project Management 32 (6) (2014) 1019–1041.

[4] M. Vanhoucke, Measuring Time - Improving Project Performance using Earned Value Management, Vol. 136 of International Series in Operations Research and Management Science, Springer, 2010.

[5] F. Anbari, Earned value project management method and extensions., Project Management Journal 34(4) (2003) 12–23.

[6] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[7] H. Hunter, R. Fitzgerald, D. Barlow, Improved cost monitoring and control through the earned value management system, Acta Astronautica 93 (0) (2014) 497 – 500. doi:http://dx.doi.org/10.1016/j.actaastro.2012.09.010.
URL http://www.sciencedirect.com/science/article/pii/S0094576512003670

[8] H. L. Chen, Improving forecasting accuracy of project earned value metrics: Linear modeling approach, Journal of Management in Engineering 30 (2) (2014) 135–145.

[9] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[10] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega The International Journal of Management Science 49 (2014) 107–122.

[11] S. S. Leu, Y. C. Lin, Project performance evaluation based on statistical process control techniques, Journal of Construction Engineering and Management 134 (2008) 813–819.

[12] R. Aliverdi, L. Moslemi Naeni, A. Salehipour, Monitoring project duration and cost in a construction project by applying statistical quality control charts, International Journal of Project Management 31 (3) (2013) 411–423.

[13] J. Colin, M. Vanhoucke, A multivariate approach for top-down project control using earned value management, Submitted to an international journal (working paper version available at www.projectmanagement.ugent.be/sites/default/files/files/wp/ColinVanhoucke2014d.pdf).

[14] J. Colin, M. Vanhoucke, A comparison of the performance of various project control methods using earned value management systems, Expert Systems with Applications-doi:http://dx.doi.org/10.1016/j.eswa.2014.12.007.

[15] W. Lipke, Speculations on project duration forecasting, The Measurable News 3 (2012) 3–7.

[16] D. Hulett, Schedule risk analysis simplified, Project Management Network 10 (1996) 23–30.

[17] R. Elshaer, Impact of sensitivity information on the prediction of project's duration using earned schedule method, International Journal of Project Management 31 (2013) 579–588.

[18] M.-Y. Cheng, Y.-W. Wu, Evolutionary support vector machine inference system for construction management, Automation in Construction 18 (5) (2009) 597–604.

[19] B. Kim, K. Reinschmidt, Probabilistic forecasting of project duration using kalman filter and the earned value method, Journal of Construction Engineering and Management 136 (2010) 834–843.

[20] M. Wauters, M. Vanhoucke, Support vector machine regression for project control forecasting, Automation in Construction 47 (2014) 92–106.

[21] G. T. Bauch, C. A. Chung, A statistical project control tool for engineering managers, Project Management Journal 32 (2001) 37–44.

[22] Q. Fleming, J. Koppelman, Earned value project management. 3rd Edition, 3rd Edition, Project Management Institute, Newton Square, Pennsylvania, 2010.
URL http://books.google.be/books?id=ZMRVngEACAAJ

[23] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[24] T. Kourti, Application of latent variable methods to process control and multivariate statistical process control in industry, International Journal of adaptive control and signal processing 19 (4) (2005) 213–246.

[25] F. Arteaga, A. Ferrer, Dealing with missing data in mspc: several methods, different interpretations, some examples, Journal of chemometrics 16 (8-10) (2002) 408–418.

[26] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshirani, The elements of statistical learning, Vol. 2, Springer, 2009.

[27] G. W. Stewart, et al., Collinearity and least squares regression, Statistical Science 2 (1) (1987) 68–84.

[28] P. Geladi, B. R. Kowalski, Partial least-squares regression: a tutorial, Analytica chimica acta 185 (1986) 1–17.

[29] J. A. Westerhuis, T. Kourti, J. F. MacGregor, Analysis of multiblock and hierarchical pca and pls models, Journal of chemometrics 12 (5) (1998) 301–321.

[30] N. Kettaneh, A. Berglund, S. Wold, Pca and pls with very large data sets, Computational Statistics & Data Analysis 48 (1) (2005) 69–85.

[31] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural computation 10 (5) (1998) 1299–1319.

[32] R. Rosipal, L. J. Trejo, Kernel partial least squares regression in reproducing kernel hilbert space, The Journal of Machine Learning Research 2 (2002) 97–123.

[33] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2013).
URL http://www.R-project.org

[34] D. Trietsch, L. Mazmanyan, L. Govergyan, K. R. Baker, Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation, European Journal of Operational Research 216 (2012) 386–396.

[35] V. Vapnik, The nature of statistical learning theory, springer, 2000.

[36] E. Demeulemeester, M. Vanhoucke, W. Herroelen, Rangen: A random network generator for activity-on-the-node networks, Journal of Scheduling 6 (2003) 17–38.

[37] M. Vanhoucke, Integrated Project Management and Control: First come the theory, then the practice, Management for Professionals, Springer, 2014.

[38] M. E. Kuhl, E. K. Lada, N. M. Steiger, M. A. Wagner, J. R. Wilson, Introduction to modeling and generating probabilistic input processes for simulation, in: S. Henderson, B. Biller, M. Hsieh, J. Shortle, J. Tew, R. Barton (Eds.), Proceedings of the 2007 Winter Simulation Conference, New Jersey: Institute of Electrical and Electronics Engineers, 2007, pp. 63–76.

[39] H. Abdi, Partial least squares regression and projection on latent structure regression (pls regression), Wiley Interdisciplinary Reviews: Computational Statistics 2 (1) (2010) 97–106.

[40] J. Henseler, On the convergence of the partial least squares path modeling algorithm, Computational Statistics 25 (1) (2010) 107–120.

[41] I. Jolliffe, Principal component analysis, Wiley Online Library, 2005.

8

Limitations, future research and conclusions

8.1   Limitations

In this concluding chapter of this book on single- and multivariate methods for statistical project control using EVM/ES, we will first outline some limitations of the research that was presented here.

In connection to the general introduction that we have presented in section 1.1, we have to emphasise that the methods and tools, that are proposed in this book, are only potent in advancing the project management practice from the hard skills paradigm [1]. In other words, the research of this book only adheres to the "science" aspect of the definition for project management that was put forward by Turner (1996) [2]. This is in itself the largest limitation of our research, since it marginalises the wider potential role of project managers as competent social and political actors in complex project-labelled arrangements [3].

Moreover, the "instrumental" rationality for project management, that we follow in this book, has been met with much criticism in literature. According to Williams (2004) [4] the acceptance and application of such a rationality "does not eliminate project failures, nor does it guarantee project succes." To add insult to injury, it has been argued in literature that, in or-

ganisational innovation and change endeavours, the very reason for using project management methodology is at the heart of project failures [5, 6]. As a result, we must stress that we certainly do not propose our research as a universal solution that can be applied to tackle the whole spectrum of organisational problems that are in practice labelled as "projects". More specifically, it is probably only suited to be applied in the operations of traditional project-oriented industries (such as defence, aerospace and construction), where techniques for planning, monitoring and control have since long been promoted, tried and tested [5, 7]. In other words, for the project management research that is presented in this book to be applied in practice, the operating company should posses some project management maturity in the area of risk management and the project execution phase [8].

With respect to this required maturity, it is of value to consider in more detail the level of knowledge that should be present in an organisation in order to apply the methods and results of this book. First, the reader, or whoever wants to consider our research for practical purposes, should have at least a basic understanding of the operations research (OR)-based quantitative techniques that have been around since to 1960s and which are applied to project scheduling, project control and risk analysis [3]. Second, in order to implement the newly developed single- and multivariate methods for EVM/ES, the reader should be acquainted with the basics of statistics and linear algebra. We propose the book by Hastie et al. (2009) [9] as an excellent starting point to explore the elements of statistical learning that are applied in this book. A third an final prerequisite, in order to apply our methods and results, is a study of the uncertainty that underlies activity durations in a real-life stochastic project execution. At the least, the implementer should be able to reflect on whether the applied uncertainty models are appropriate for their use-case and whether the defined state of schedule control accurately expresses the subjective desire and needs of the stakeholders in the project.

A further limitation to the research of this book is that its lion share is a theoretical exercise on how EVM/ES can be applied in practice. As the English proverb goes "the proof of the pudding is in the eating", we have not accomplished to apply it in practice. Although case studies, templates and illustrative examples are provided throughout this book or in its appendix, they are mostly constructed around a what-if analysis and are not empirical records of real-life applications. We believe that this could not be obtained within the scope of this research, that is conducted in order to obtain doctorate degree, but suggest it as a challenge for the future research that builds on the results of this book. We will come to this subject shortly, in the second section of this chapter. For now, we will present some more specific restrictions that were made in order to limit the scope of our research, as it is presented in this book.

Throughout the preceding chapters, we have remained a restricted focus on the time aspect of the control process in the execution of a project. It is our belief that the modelling, that is applied in the Monte Carlo simulations for project executions, is most rightfully justified for the time aspect only. Given the uncertainties that are added to activity durations, it is possible to model the time progression of project executions realistically. Although the manifestation of this uncertainty could also be translated to the actual costs in a project, we believe that this would lead to results that are not as generalizable as those with respect to the time aspect. Only costs that are directly associated with the duration of activities could have been incorporated. In order to include additional cost structures, we would have needed to consider resources, which were chosen not to be incorporated into the scope of our work. The remaining dimension of the "iron triangle" (i.e. scope) is also not considered in this work. As Atkinson (1999) [10] named it quite adequately "a phenomenon", we found it inopportune to try to include it in the mathematical modelling for simulations that stands at the basis of the results that are shown in this book.

As a consequence to our restricted focus on the time aspect of project control, we have tested and compared all of the discussed methods (using EVM/ES) based on their ability to generate warning signals, if a project progress situation does not concur with a pre-defined state of schedule control. More specifically, we have excluded a part of the control process from the analysis. This part would represent the project manager's actions to bring the project back (or closer) to this pre-defined state. Although such corrective actions have been described in previous literature [11, 12], we chose to leave them for further research endeavours to incorporate them into our statistical project control approach.

8.2 Future research

In this future research section, we will first address some challenges (section 8.2.1) that follow from the limitations of our research. Next, in section 8.2.2, we shed some light on some potentially interesting research directions for future studies on project control using EVM/ES.

8.2.1 Challenges that follow from our research

From the discussion, in the forgoing section, on the limitations of the research that is presented in this book, we derive some general challenges. They represent opportunities for follow-up research or signify the hurdles

that need to be tackled if the results of this book are to be brought into the project management practice.

The first challenge is the development of a software package that implements the methods that are presented in this book and makes them available for academics and practitioners. The latter however, are not to keen on new tools and technology. In a survey that was done by White and Fortune (2002) [13], it was shown that most respondent from the project management practice use only a small number of methods, tools and techniques with project management software and Gantt charts being the most widely used aids. To make things worse, almost half of the respondents reported drawbacks to the methods, tools and techniques they had employed. These findings contrast strongly with e.g. the plethora of newly developed models to calculate and interpret EVM/ES in recent literature [14–19]. In order for any research on tools and techniques for EVM/ES to be taken up by the project management practice, it is indispensable that practitioners are presented with a software suite that is easily accessible and from which the practical benefits become immediately clear. For academics however, the newly developed methods can already, in part, be accessed in the P2 Engine [20] software and R [21] implementations can be made available by the author, upon request.

A second challenge arises from the inputs of the simulation models, that we apply in order to fuel the newly developed single- and multivariate methods for statistical project control using EVM/ES. During the review process of the corresponding papers, that describe these methods, it has come to our attention that the concept of setting a desirable state of schedule control is not easily understood. In *the first phase* of our simulation models, it was possible to set a pre-defined state of schedule control. The methods where then tested in *the second phase* with respect to their ability to accurately signal a departure from said state of control. This state of schedule control should also be specified in a practical setting, for a real-life execution of a project, such that the project manager will only be signalled with a warning if the project progress situation deviates from that state. The observation that this concept is not entirely clear, from a practitioner's point of view, provokes the believe that additional material should be developed. This material could come in the form of simulated case-based exercises which provide the user with hands-on practice regarding the concept of the state of schedule control. We believe that this a viable manner to proceed with, in order to introduce the statistical project control framework into practice, since according to Walker et al. (2008), understanding of a theory comes through reflection in learning, by challenging it and testing it in practical ways [22].

A final challenge that is provoked, by the research in this book, is to perform an integrated summary study of all of the methods that are described

here. This could not be realised within the scope of the research project that produced this book but also, the time-frame* in which this book needed to be finalised did not allow so. In order for this integrated summary study to be possible (or better: opportune), at least some of the presented methods for statistical project control should have been met with either criticism in literature, empirical evidence or practitioners' feedback. This is most rightfully so for the concept of the pre-defined (desirable) state of schedule control (see the previous paragraph), the empirical classification of chapter 4, the risk and variation input modelling for uncertainty and the different probability distributions they are used with. If, in some years, the project management community, both academics and practitioners, has shed its light on these elements, then it will be sensible to incorporate their finding and remarks. Using those, it will be possible to perform a summary study, in which all of the methods for EVM/ES schedule control, that are discussed in this book, are tested and compared in an integrated manner.

8.2.2   Potential directions for future research on project control using EVM/ES

We now proceed with a discussion of future research directions on project schedule control using EVM/ES systems. First, we will provide a short overview of potential techniques that can be included in our statistical project control framework for top-down schedule control. Next, we will introduce two potentially interesting studies that link the usage of the EVM/ES system with risk analysis indicators that can be used in a bottom-up control strategy for project control.

Top-down statistical project control

With respect to directions for the future research on top-down statistical project control, we can be brief. We believe that the models and simulation frameworks, that are presented in this book, provide a sufficient basis for testing, comparing and developing newly proposed techniques for the use of EVM/ES in a top-down project control framework. We aspire that academics will take up some of the work that is presented in this book, and published in its papers, in order to validate it (see the last paragraph of section 8.2.1) and to include additional techniques. From a study of the current literature, we identify two main sets of techniques that can be applied in our statistical project control framework.

---

*This book was finalised in order for the author to obtain his doctorate degree, after 5 year of intensive study and research.

Forecasting techniques: We believe that the methods and techniques that have been proposed for forecasting the final duration of a project can all be easily integrated into our framework. In forecasting techniques using EVM/ES, an estimate is made for the final real duration of a project, at any review period during the execution of that project. These dynamic forecasts can therefore also be produced for the set of fictitious project executions that are generated in *the first phase* of the simulation framework, that is deployed throughout the research in this book. An empirical cumulative distribution function of this estimate can then be produced from which (through sample quantiles) tolerance limits can be constructed. These tolerance limits can then be tested in *the second phase*, with respect to their ability to generate warning signals, if the project progress situation deviates from a pre-defined state of schedule control. We provide the reader with a non-exhaustive list of techniques, which have been presented in the literature, that we see fit to be applied to the statistical project control framework [14–19, 23–25].

Classification techniques: The set of classification techniques, that have been presented in statistical literature, presents and even more straight-forward avenue for the future research in the statistical project control framework. Such procedures have already been proposed for classifying records of projects that have been executed in real life (linear discriminant analysis [26], support vector machines [27]) or for estimating the required effort in a project (bagging [28], neural networks [28, 29], support vector machines [30]), but we believe that they can also be applied for dynamic schedule control, using our statistical project control framework.

Bottom-up statistical project control

We will also present two future research directions for the use of the EVM/ES system in order to define schedule risk metrics for the activities of a project. Although this might seem almost contradictory to our discussion, in this book, of the top-down use of EVM/ES, we believe that potentially interesting results can also be obtained when the EVM/ES system is transformed in order to be used as a bottom-up approach. Along the following paragraphs, we will discuss two ways in which schedule risk analysis (SRA)-like indicators can be produced for the activities in a project, from the observed dynamics of the EVM/ES system in the statistical project control framework. These SRA-like indicators could then be used for a bottom-up project control approach, such as is discussed in the study by Vanhoucke (2011) [31].

Setting tolerances on estimated activity durations: In chapter 7, we

have shown how multivariate regression methods can be used to estimate the activity durations in an execution of a project, based on the EVM/ES schedule performance measures that are observed at the highest level of the WBS. In the MR-LP methods, these estimated activity durations are then used to calculate the longest path trough the project network in order to produce a forecast for the total project duration.

However, one could take this procedure a step back and consider the estimated activity durations on their own. Consider an activity $i \in \mathcal{N}$, for which $\mathcal{N}$ represents the set of all activities in a single project. Let d_i represent the real duration of that activity, $\tilde{d}_i$, its baseline estimate duration and $\widehat{d}_i$, the estimate that is produced by a regression method from the EVM/ES performance measures that are observed at a given review period during the execution of the project. The estimate $\widehat{d}_i$ will be subject to some uncertainty, which is associated with the calculations of the regression coefficients (due to numerical instabilities in the calculation of the inverse of matrices). Let us assume, for now and for the sake of the argument, that the resulting uncertainty is normally distributed with mean 0 and variance σ_i^2. This would mean that the real duration can be assumed to be equal to $\widehat{d}_i$, except for an error term ($N(0, \sigma_i^2)$).

This probability distribution for the real duration (which we will denote as pdf_1) can then be checked against the probability distribution that was set around the baseline estimate $\tilde{d}_i$ and that was defined by the state of schedule control in the statistical project control framework. Whether the state of schedule control has been set using the "acceptable variation" from chapter 2, the generalised beta distributions (defined by ω_R and ω_V) in chapters 3, 6 and 7 or the "in-control" classification of chapter 5, a probability distribution around the baseline estimate $\tilde{d}_i$ has always been set in our statistical project control framework (we will denote this distribution as pdf_2).

Given the distribution of the real duration (derived from the estimate $\widehat{d}_i$, pdf_1) and the distribution around the baseline estimate $\tilde{d}_i$ (pdf_2), we can approximately calculate the probability that the real duration will not be included in pdf_2. For now, let us denote this probability as p. If p is large, it will be likely that the project progress situation deviates from pre-defined state of schedule control. If p is small, this deviation is much less likely. In order to derive an actionable warning signal, one could set a tolerance limit on this p, which would be derived from a sample quantile (α) of all the p's that are recorded in *the first phase* of our simulation framework.

So far, we have still followed the top-down rational of producing

warning signals from EVM/ES measures, that are observed at a high level of the WBS. The sample quantile α, discussed earlier, can represent a fixed quantile for all activities i. Better however, a optimization model could be build, which optimizes the quantile α_i for all activities, such that more true, and less false, warning signals are generated. In other words, a simultaneous maximization of the *detection performance* and minimization of the *probability of overreactions*. Let α_i^* denote this optimal quantile for an activity i. These values α_i^* can then be compared amongst activities. A small value for α_i^* would mean that, for activity i, the (albeit estimated) duration needs to be monitored closely. Alternatively, a large value for α_i^* would mean that the activity i does not need to be monitored closely. It now becomes apparent how these α_i^* can be interpreted as SRA-like indicators which can be used in a bottom-up schedule control approach.

Tolerances synthesis for activity durations: An alternative to construct SRA-like indicators for the activities in a project, based on the EVM/ES system, is provided by what is described as "tolerance synthesis" in literature [32, 33]. Here, the state of schedule control is assumed to be expressed at the project level of the WBS, through the observed EVM/ES schedule performance measures. This state of schedule control is then considered to be a specification limit that is imposed on the quality variables (EVM/ES schedule performance measures) of the project. These specification limits consequently impose restrictions on the values of the underlying variables (i.e. the activity durations).

According to the theory described by González and Sánchez (2009) [34], the specification limits therefore impose the need to optimally design the tolerances for the activity durations. The design methodologies that are described by González and Sánchez (2009) incorporate the assumption of correlation between the underlying variables. This assumption has also been suggested and validated for activity durations in projects [35, 36], and therefore presents a good fit to be applied for a project management application.

It remains to be seen, from an in-depth study, what value the theory of tolerance synthesis provides for project control. From a preliminary investigation however, we are tempted to say that the produced tolerances for the different activities in a project could, when compared to one other, represent SRA-like schedule performance indicators. Alternatively, the tolerances could be used directly as thresholds for taking actions in a bottom-up study, such as was proposed by Vanhoucke (2011) [31]. Moreover, the specification of tolerances could also embody the potential to be used as guidelines on how (not only where) specific actions should be taken in a bottom-up project

control approach.

We perceive the aforementioned research directions for bottom-up project control approaches to be more challenging than those that were described for top-down statistical project control. Nevertheless, while the latter lie closer to the research that we have done for this book, and the former are further from what we have accomplished in our studies, we only present them both as potentially interesting avenues for future research. Since it was beyond the scope of this book to include them, we provide them only as "food for thoughts", rather than completely outlined research actions that need to be taken in order to produce insightful results.

8.3   Conclusions

In this book, we have presented research on top-down schedule control using EVM/ES for project management. This research was structure around the concept of statistical project control, which we have rephrased in this book as "the use of simulated project progress situations in order to develop and test practically applicable approaches for project schedule control using EVM/ES". Using this statistical project control concept, we have modelled and tested methods from the current state of EVM/ES control, in practice (rules of thumb (RoT), statistical process control (SPC) chart, longest path, (LP)) and a critical chain/buffer management (CC/BM) derivative. In order to test these methods, a state-of-the-art simulation framework was developed, with which we were able to produce fictitious project progress situations for a large dataset of project networks. Moreover, this simulation framework deployed a separate modelling for the uncertainty that is applied to activity durations (risk and variation) and typically consisted out of two distinct phases. The first phase emulates a simulation that can also be done prior to the execution of a project, in a practical real-life application. The second phase imitated a large number the execution of that project, in order to test the performance of the applied methods for schedule control. The performance of these methods represents their ability to produce warning signals if a project progress situation deviates from a pre-defined state of schedule control. The distinction between the first phase and the second phase is important, since the former has been put into place specifically in order for newly developed single- and multivariate methods for EVM/ES to be applied. The single- (statistical tolerance limits(STL)) and multivariate (multivariate statistical project control (MSPC) and multivariate regression methods for longest path calculations (MR-LP)) specifically use the output of this first phase simulation in order to fuel their statistics/machine learning algorithms, which produces the warning signals for project schedule control. We will now summarize the

conclusions, with respect to the different methods that are presented in this book, along the following paragraphs.

The paper, that constitutes chapter 2, has introduced the concept of statistical project control in the literature on project management. In addition, the two-phased simulation setup was shown for the first time. This setup allows statistical tolerance limits (STL) to be calculated, which are shown to outperform the RoT, LP and CC/BM approaches with respect to their ability to produce warning signals for project schedule control.

The main contribution, of the research that is presented in chapter 3, lies in the introduction of the theoretical basis of the CC/BM methodology to the practice of project control using EVM/ES systems. Moreover, the mathematics on how to apply EVM/ES at different control points in a project baseline schedule was shown.

Chapter 4 presents an empirical perspective on the inputs that can be supplied to the simulation and testing framework in our statistical project control approach. We provide the first application of a theoretical distribution to a large database of empirical records in project management. From this application, we derived an empirical classification for records of project executions, which can assist in the interpretation of historical data and serve as input to our simulation and testing framework. In addition, we have shown how the calibration procedure and the empirical classification can be usefully applied in the project management practice, and validated this by a case study.

Chapter 5 applied the aforementioned classification in a study on the use of SPC charts for project schedule control. Four SPC procedures, previously suggested in literature, were tested against one another with respect to their ability to generate warning signals. The procedure that was proposed by Leu and Lin (2008) [37] was shown, in an extensive computational experiment, to outperform the others.

The MSPC method, that is presented in chapter 6, is shown to outperform the STL approach. It is reasoned that the principal component analysis of the correlation structure of the EVM/ES schedule performance measures, which produces the multivariate metrics (T^2 and SPE), improves the warning signals for schedule control, due to the removal of noise and collinearities.

Chapter 7 has taken this analysis a step further. It explored the use of multivariate regression methods in order to estimate the activity durations from the EVM/ES schedule performance measures that are observed at a high level of the WBS. By doing so, the requirement to physically drill down the WBS at each review period is removed, as the longest path through the project can be calculated from these estimates. Four distinct multivariate regression methods were compared with one another and with the LP

method in an extensive computational experiment. The kPCR (with a rbf kernel) method is shown to be the most promising. It reduces the effort that is spent in the project control process (since the periodic drill down of the WBS by the project manager is no longer required) , while only $\sim 10\%$ is lost in terms of the performance of the warning signals is generates.

With respect to the research questions that directed our research in the statistical project control framework in an informal manner, we are now able to summarize the following:

Q1: How well does the EVM/ES system work for project schedule control?

- *EVM/ES is able to distinguish a project progress situation that deviates from a pre-defined state of schedule control, but only when the calibration to a simulated set of project executions is included (RoT).*

Q2: How does the performance of the EVM/ES system compare to that of its alternatives?

- *The project control effectiveness can be improved by the application of SPC charts, by including multiple control points in a schedule (CC/BM) or by controlling the project with a bottom-up perspective (LP). If the project control efficiency is considered, the results are more nuanced and the aforementioned trend is no longer as evident from our findings.*

Q3: How can the use of the EVM/ES system, in practice, be improved?

- *The project control practice using EVM/ES can be improved by including machine learning and statistics techniques. In particular, tolerance limits, multivariate project control metrics and multivariate regression methods can allow projects to be controlled more effectively, while keeping the effort within reasonable limits.*

References

[1] J. Pollack, The changing paradigms of project management, International Journal of Project Management 25 (3) (2007) 266–274.

[2] J. Turner, Editorial: International project management association global qualification, certification and accreditation, International journal of project management 14 (1) (1996) 1–6.

[3] S. Cicmil, D. Hodgson, New possibilities for project management theory: A critical engagement, Project Management Journal 37 (3) (2006) 111–122.

[4] T. Williams, Assessing and building on the underlying theory of project management in the light of badly over-run projects, in: Project Management Institute Research Conference, 2004.

[5] H. Maylor, Beyond the gantt chart:: Project management moving on, European Management Journal 19 (1) (2001) 92–100.

[6] A. Clarke, A practical use of key success factors to improve the effectiveness of project management, International Journal of Project Management 17 (3) (1999) 139–145.

[7] J. D. Frame, Project Management Competence: Building Key Skills for Individuals, Teams, and Organizations, Wiley, Jossey-Bass, 1999.

[8] Y. H. Kwak, C. W. Ibbs, Assessing project management maturity, Project Management Journal 31 (1) (2000) 32–43.

[9] T. Hastie, R. Tibshirani, J. Friedman, T. Hastie, J. Friedman, R. Tibshirani, The elements of statistical learning, Vol. 2, Springer, 2009.

[10] R. Atkinson, Project management: cost, time and quality, two best guesses and a phenomenon, its time to accept other success criteria, International journal of project management 17 (6) (1999) 337–342.

[11] M. Vanhoucke, Project Management with Dynamic Scheduling: Baseline Scheduling, Risk Analysis and Project Control, Vol. XVIII, Springer, 2012.

[12] W. Herroelen, R. Leus, On the merits and pitfalls of critical chain scheduling, Journal of Operations Management 19 (2001) 559–577.

[13] D. White, J. Fortune, Current practice in project managementan empirical study, International journal of project management 20 (1) (2002) 1–11.

[14] H. Khamooshi, H. Golafshani, EDM: Earned duration management, a new approach to schedule performance management and measurement, International Journal of Project Management 32 (6) (2014) 1019–1041.

[15] B. Kim, K. Reinschmidt, Probabilistic forecasting of project duration using kalman filter and the earned value method, Journal of Construction Engineering and Management 136 (2010) 834–843.

[16] B. Kim, K. Reinschmidt, Probabilistic forecasting of project duration using bayesian inference and the beta distribution, Journal of Construction Engineering and Management 135 (2009) 178–186.

[17] M. Wauters, M. Vanhoucke, Support vector machine regression for project control forecasting, Automation in Construction 47 (2014) 92–106.

[18] L. Moslemi Naeni, A. Salehipour, Evaluating fuzzy earned value indices and estimates by applying alpha cuts, Expert systems with Applications 38 (7) (2011) 8193–8198.

[19] T. Narbaev, A. De Marco, An earned schedule-based regression model to improve cost estimate at completion, International Journal of Project Management 32 (6) (2014) 1007–1018.

[20] OR-ASResearch, Visit the P2 Engine website at www.p2engine.com.

[21] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2013). URL http://www.R-project.org

[22] D. H. Walker, S. Cicmil, J. Thomas, F. Anbari, C. Bredillet, Collaborative academic/practitioner research in project management: theory and models, International Journal of Managing Projects in Business 1 (1) (2008) 17–32.

[23] G. A. Barraza, R. A. Bueno, Probabilistic control of project performance using control limit curves, Journal of Construction Engineering and Management 133 (12) (2007) 957–965.

[24] S. Bhokha, S. O. Ogunlana, Application of artificial neural network to forecast construction duration of buildings at the predesign stage, Engineering, Construction and Architectural Management 6 (2) (1999) 133–144.

[25] Ö. Hazır, A review of analytical models, approaches and decision support tools in project monitoring and control, International Journal of Project Management.

[26] D. Dvir, S. Lipovetsky, A. Shenhar, A. Tishler, In search of project classification: a non-universal approach to project success factors, Research policy 27 (9) (1998) 915–935.

[27] Y. Hu, J. Huang, J. Chen, M. Liu, K. Xie, Software project risk management modeling with neural network and support vector machine approaches, in: Natural Computation, 2007. ICNC 2007. Third International Conference on, Vol. 3, IEEE, 2007, pp. 358–362.

[28] P. L. Braga, A. L. Oliveira, S. R. Meira, Software effort estimation using machine learning techniques with robust confidence intervals, in: Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on, Vol. 1, IEEE, 2007, pp. 181–185.

[29] I. F. de Barcelos Tronto, J. D. S. da Silva, N. SantAnna, An investigation of artificial neural networks based prediction systems in software project management, Journal of Systems and Software 81 (3) (2008) 356–367.

[30] A. L. Oliveira, Estimation of software project effort with support vector regression, Neurocomputing 69 (13) (2006) 1749–1753.

[31] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[32] J. Kharoufeh, M. Chandra, Statistical tolerance analysis for non-normal or correlated normal component characteristics, International journal of production research 40 (2) (2002) 337–352.

[33] P. Varghese, R. N Braswell, B. Wang, C. Zhang, Statistical tolerance analysis using frpdf and numerical convolution, Computer-Aided Design 28 (9) (1996) 723–732.

[34] I. González, I. Sánchez, Statistical tolerance synthesis with correlated variables, Mechanism and Machine Theory 44 (6) (2009) 1097–1107.

[35] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega The International Journal of Management Science 49 (2014) 107–122.

[36] D. Trietsch, K. R. Baker, Pert 21: Fitting PERT/CPM for use in the 21st century, International Journal of Project Management 30 (4) (2012) 490–502.

[37] S. S. Leu, Y. C. Lin, Project performance evaluation based on statistical process control techniques, Journal of Construction Engineering and Management 134 (2008) 813–819.

A

Appendix

A.1 R template for the proposed classification of the data of Batselier and Vanhoucke (2014)

This appendix presents a working R template to reproduce the classification of the empirical records of [1]. The reader should note that the results that are presented in this paper are subject to some numerical instability and therefore, small differences can be expected when the R template is run. In particular, the uncertainty associated with the estimation of the coefficients for the linear regression models might cause slightly different coefficients for each execution of the R template. In this appendix, we first list the files that should be present on the workstation where this R template is executed. Second, we present the outline of the R template and finally, the actual R code is shown.

A.1.1 Required inputs

From the `statistical project control` research page available at `www.projectmanagement.ugent.be` the following files should be downloaded to the working directory of the R process:

- `Batselier_Vanhoucke_2014.xlsx`

- `Trietsch_etal_2012.R`

- `Grouping_tests.R`

- `Looney_and_Gulledge_1985.txt`

A.1.2   Outline

Preamble

The preamble of the R template includes statements to clear the workspace, to load a package to read excel files and to define some subsidiary functions.

Data

The activity-level data from the projects in the dataset of [1] are read from the Microsoft Excel file (`Batselier_Vanhoucke_2014.xlsx`). This Microsoft Excel file contains the planned and actual durations for all activities in the 24 projects that are suited for the analysis.

Four-step procedure of [2]

The four-step procedure of [2] is applied to these 24 projects. If the Parkinson distribution with a lognormal core is not confirmed for a certain project, the four-step procedure returns `NA` as output. If however, the distributional model for a project is confirmed, the function returns a list for that project which includes:

- the step `(P0,P1,P1.1,P1.2)` in which the hypothesis was confirmed

- the s and m estimates ($\hat{s}$ and $\hat{m}$)

- the p value

- a list of the values in the empirical relative distribution

Classification

The classification of table 4.3 was produced from the grouping of the projects with respect to their $\hat{s}$ and $\hat{m}$ values. This grouping can be reproduced through the application of Levene's and Welsch's tests to the output of the

four-step procedure. First the statistical interpretation of $\hat{s}$ can be reproduced, and next, the statistical interpretation of $\hat{m}$ can be repeated. The latter includes the implementation of the enumeration algorithm.

A.1.3   Template

```
1  #-------------------------------------------------------------------------
2  #    R worksheet to reproduce the results for:
3  #
4  #    Colin and Vanhoucke (2014): "An empirical perspective on activity
5  #    durations for project management simulation studies"
6  #
7  #-------------------------------------------------------------------------
8
9
10 ##------------------------------------------------------------------------
11 ##----------------------------- PREAMBLE  --------------------------------
12 rm(list=ls(all=TRUE))          # clear workspace
13 library("xlsx")               # to read excel files
14 source("Trietsch_etal_2012.R")
15 source("Grouping_tests.R")
16
17 sublist <- function(l,indices){
18   out <- sapply(indices,function(i){
19     l[[i]]
20   },simplify=FALSE)
21   names(out) <- names(l)[indices]
22   return(out)
23 }
24
25 chunk <- function(x,n){
26   split(x, factor(sort(rank(x)%%n)))
27 }
28
29 ##------------------------------------------------------------------------
30 ##------------------------------- DATA -----------------------------------
31 ##  Read the (cleaned) data of
32 ##  Batselier and Vanhoucke 2014
33
34 d_over_d_hat <- sapply(c("C2011-07","C2011-12","C2011-13","C2012-13","C2013
       -01","C2013-02","C2013-03","C2013-04","C2013-05","C2013-06","C2013-07","
       C2013-08","C2013-09","C2013-10","C2013-11","C2013-12","C2013-13","C2013
       -14","C2013-15","C2013-16","C2013-17","C2014-01","C2014-02","C2014-03"),
       function(project.name){
35   print(project.name)
36   d_and_d_hat <- read.xlsx("Batselier_Vanhoucke_2014.xlsx",sheetName=project.
         name)
37   return(apply(d_and_d_hat,1,function(x) x[1]/x[2]))
38 })
39
40 ##------------------------------------------------------------------------
41 ##----------------------- FOUR-STEP PROCEDURE ----------------------------
42 ##  Apply the four-step procedure of
43 ##  Trietsch et al. 2012
44
45 d_over_d_hat.model <- sapply(1:length(d_over_d_hat),function(i){
46   print(names(d_over_d_hat)[i])
47   ##-- Apply "Trietsch et al. 2012"
48   T <- Trietsch_etal_2012(d_over_d_hat[[i]])
49   ##-- Add confidence intervals
50   if(!is.logical(T)){
51     beta <- 0.05
```

```
52     n <- length(T$x)
53     T[["s_CI"]] <- c(sqrt((n-1)*T$s^2/qchisq(1-beta/2,n-1)),sqrt((n-1)*T$s^2/
           qchisq((beta/2),n-1)))
54     T[["m_CI"]] <- c(T$m-qt(1-beta/2,n-1)*T$s/sqrt(n),T$m+qt(1-beta/2,n-1)*T$
           s/sqrt(n))
55   }
56   return(T)
57 })
58 names(d_over_d_hat.model) <- names(d_over_d_hat)
59
60 ##  Print the output
61
62 d_over_d_hat.confirmed <- sublist(d_over_d_hat.model,(1:24)[sapply(d_over_d_
       hat.model,function(x){!is.logical(x)})])
63 d_over_d_hat.confirmed
64
65 ##-----------------------------------------------------------------------
66 ##------------------------- CLASSIFICATION ----------------------------
67 ##  Statistical interpretation of hat(s)
68 ##  Levene's test
69
70 grouping_test(sublist(d_over_d_hat.confirmed,1:12),"sd")
71
72 ##  Pooled hat(s)
73
74 pooled_s <- c(sum((sapply(sublist(d_over_d_hat.confirmed,1:12),function(T)
       length(T$x))-1)*sapply(sublist(d_over_d_hat.confirmed,1:12),function(T) T
       $s)^2)/sum((sapply(sublist(d_over_d_hat.confirmed,1:12),function(T)
       length(T$x))-1)), d_over_d_hat.confirmed[[13]]$s)
75
76 ##  Statistical interpretation of hat(m)
77
78 d_over_d_hat.confirmed <- lapply(1:length(d_over_d_hat.confirmed),function(i)
       { #-- recalculate the CI's around hat(m) using the pooled_s
79   T <- d_over_d_hat.confirmed[[i]]
80   beta <- 0.05
81   n <- length(T$x)
82   T[["m_CI"]] <- c(T$m-qt(1-beta/2,n-1)*c(rep(pooled_s[1],12), pooled_s[2])[i
       ]/sqrt(n),T$m+qt(1-beta/2,n-1)*c(rep(pooled_s[1],12), pooled_s[2])[i]/
       sqrt(n))
83   return(T)
84 })
85 names(d_over_d_hat.confirmed) <- names(sublist(d_over_d_hat.model,(1:24)[
       sapply(d_over_d_hat.model,function(x){!is.logical(x)})]))
86
87 ##  Welsch's test
88
89 grouping_test(sublist(d_over_d_hat.confirmed,1:13),"mean")
90 grouping_test(sublist(d_over_d_hat.confirmed,1:12),"mean")
91 grouping_test(sublist(d_over_d_hat.confirmed,10:12),"mean")
92 grouping_test(sublist(d_over_d_hat.confirmed,1:9),"mean")
93
94 ##  Enumeration alorithm
95
96 combination.list <- sapply(1:13,function(i) combn(1:13,i))           #-- List
       all combinations
97 F.list <- lapply(combination.list,function(l) apply(l,2,function(x){    #--
       Welsch's test for all combinations
98   out <- grouping_test(sublist(d_over_d_hat.confirmed,x),"mean",print=FALSE)
99   out["p.value"]
100 }))
101 p.list <- sapply(1:13,function(i){                               #-- Test whether the
       remaining projects can be organized in another group
102   sapply(1:length(F.list[[i]]),function(j){
103     if(is.na(F.list[[i]][j])|any(combination.list[[i]][,j]==13)){
104       F.list[[i]][j] <- 0
105     }
```

```
106       if(F.list[[i]][j]>0.05){
107         out <- grouping_test(sublist(d_over_d_hat.confirmed,setdiff(1:13,c(
              combination.list[[i]][,j],13))),"mean",print=FALSE)
108         if(is.na(out["p.value"])){
109           out["p.value"] <- 0
110         }
111         if(out["p.value"]>0.05){
112           out["p.value"] + F.list[[i]][j]
113         }else{
114           return(NA_real_)
115         }
116       }else{
117         return(NA_real_)
118       }
119   })
120 })
121 output.list <- sapply((1:13)[sapply(p.list,function(p) max(sapply(p,function(
        p_i) max(p_i,na.rm=TRUE))))>0],function(i){ #-- Print all combinations
        that confirm in-group equality of mean
122   sapply((1:dim(combination.list[[i]])[2])[!is.na(p.list[[i]])],function(j){
123     comb <- combination.list[[i]][,j]
124     print(paste("Combined p-value: ",p.list[[i]][j],sep=""))
125     print(paste("{",paste(comb," ",collapse=""),"}/{",paste(setdiff(1:13,c(
          comb,13))," ",collapse=""),"}/{",13,"}",sep=""))
126   })
127 })
128 rbind(grouping_test(sublist(d_over_d_hat.confirmed,c(4,5,7,9:12)),"mean"),
        grouping_test(sublist(d_over_d_hat.confirmed,c(1,2,3,6,8)),"mean"),
        grouping_test(sublist(d_over_d_hat.confirmed,13),"mean"))           #--
        Print the combination that was chosen (table 2)
129
130 ##  Pooled hat(m)
131
132 pooled_m <- c(sum(sapply(sublist(d_over_d_hat.confirmed,c(4,5,7,9:12)),
        function(T) length(T$x))*sapply(sublist(d_over_d_hat.confirmed,c
        (4,5,7,9:12)),function(T) T$m))/sum(sapply(sublist(d_over_d_hat.confirmed
        ,c(4,5,7,9:12)),function(T) length(T$x))),sum(sapply(sublist(d_over_d_hat
        .confirmed,c(1,2,3,6,8)),function(T) length(T$x))*sapply(sublist(d_over_d
        _hat.confirmed,c(1,2,3,6,8)),function(T) T$m))/sum(sapply(sublist(d_over_
        d_hat.confirmed,c(1,2,3,6,8)),function(T) length(T$x))),sum(sapply(
        sublist(d_over_d_hat.confirmed,c(13)),function(T) length(T$x))*sapply(
        sublist(d_over_d_hat.confirmed,c(13)),function(T) T$m))/sum(sapply(
        sublist(d_over_d_hat.confirmed,c(13)),function(T) length(T$x))))
```

Chapter4/Colin2014b_Technical_template.R

A.2   Calculating principal components

In the very early years of the 20$^{\text{th}}$ century PCA was developed by Pearson
[3]. Ever since, PCA has been a popular procedure to reduce the dimen-
sionality of a variable space. A full coverage on the basics of linear algebra,
fundamental to PCA, lies outside the scope of this paper and the reader is
referred to the recent book of Jolliffe [4]. Instead, we will only introduce
the properties of PCA using a graphical representation and give a brief
overview of the matrix calculus that is required.

PCA assumes that the true rank of a matrix of observations X is less than
the number of observations which are made. Consequently, it conjectures
that the observations can be projected onto a new set of coordinate axes,
thereby removing redundancy and noise from the system. The PCA de-
composition method first calculates the principal components of the ob-
servation space, i.e. the directions that will make up the new coordinate
axes. These principal components are often also named the latent vari-
ables, since they represent the underlying (unobservable) factors really in-
fluencing the system dynamics. Historically, the development of the Non
Iterative Partial Least squares (NIPALS) algorithm meant the real break-
through for PCA. Consider a $(n \times P)$ matrix X that is a collection of n
measurements for a P-variate randomly distributed variable $\mathbf{x}$. The first
principal component of $\mathbf{x}$ is defined as the vector of coefficients $\mathbf{p}_1$ for
which the linear combination $t_1 = \mathbf{x}\mathbf{p}_1$ captures as much as possible of the
variance contained in X, subject to $|\mathbf{p}_1| = 1$. The second principal compo-
nent is then the vector of coefficients $\mathbf{p}_2$ for which the linear combination
$t_2 = \mathbf{x}\mathbf{p}_2$ contains as much of the variance from X that is not captured
within t_1. Additional principal components up to P are similarly defined.
Some of the literature on the NIPALS algorithm is worth a read as it can
provide insight into the characteristics of PCA [5, 6]. In addition, Shlens [7]
gives an intuitive explanation of how PCA is used to reduce the dimension
of the problem.

In more recent work, PCA is always performed using the computationally
more efficient singular value decomposition (SVD, [8]) of X. The SVD al-
gorithm produces three matrices, a $(n \times P)$ matrix U, a $(P \times P)$ diagonal
matrix L and a $(P \times P)$ matrix A.

$$X = \mathrm{ULA}^T \tag{A.1}$$

The matrix U, whose columns contain the left singular vectors of X is mul-
tiplied with the diagonal matrix L, containing the singular values of X, to
form the matrix of scores T. The matrix A contains the right singular vec-
tors of X and is equal to the matrix of loadings P. The PCA decomposition,
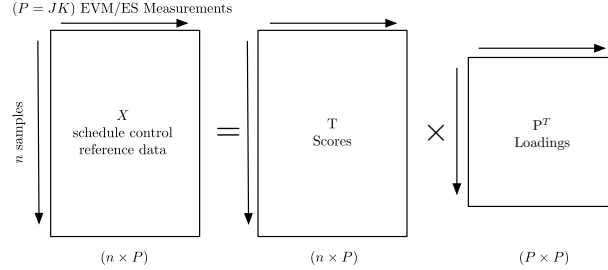
Figure A.1: Principle component analysis as a data decomposition method (Macgregor (1997)).

illustrated in figure A.1, can then be written as:

$$X = \text{TP}^T \tag{A.2}$$

The singular values on the diagonal of L are equal to the square roots of the eigenvalues ($\sqrt{l_i^2}, \forall i \in \{1, \ldots, P\}$) of $X^T X$. From these, we can find the standard deviation of the ith principal component:

$$s_{\mathbf{t_i}} = \sqrt{\frac{l_i^2}{n-1}} \tag{A.3}$$

The score-loading nomenclature is very common in PCA literature and is therefore adopted here. The matrix of loadings $\text{P} = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 & \ldots & \mathbf{p}_P \end{bmatrix}$ can be seen as a $(1 \times P)$ collection of $(P \times 1)$ vectors of coefficients for the linear combination $t_i = \mathbf{x}\mathbf{p}_i$ that defines the ith principal component. A related term "matrix of rotations" expresses the geometrical interpretation of the loadings, as they represent a new coordinate space onto which $\mathbf{x}$ is projected. The $(n \times P)$ matrix of scores T can then be interpreted as the values in the new coordinate space for the collection of n measurements of $\mathbf{x}$. We illustrate this geometrical interpretation of PCA using a trivial example in appendix A.3. The reader is referred to [6, 7] for further illustrations. Such a geometrical interpretation can be particularly helpful to obtain insights in the relative importance of the different principal components, expressed by their standard deviations. We will now explore this further, while discussing whether all principal components should be retained in the analysis or not.

If all of the P principal components are retained, for a further analysis of the data, an observation for the P-variate vector of observations $\mathbf{x}$ can be
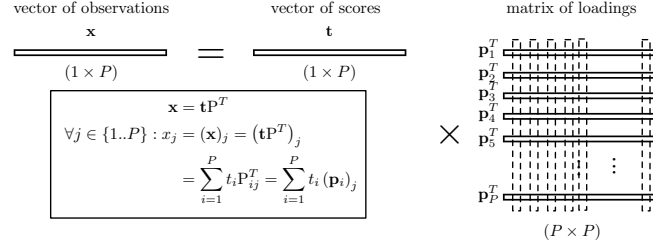
Figure A.2: Principle component analysis interpretation for a vector of observations

reconstructed from its $(1 \times P)$ vector of scores $\mathbf{t}$:

$$\mathbf{x} = \mathbf{t}\mathrm{P}^T$$

$$= \sum_{i=1}^{P} t_i \mathbf{p}_i \qquad (A.4)$$

as depicted in figure A.2. However, since PCA is used to find a solution to the problems associated with the multivariate nature of project data, it will try to reduce the dimensionality of the problem in a structured manner, without losing valuable information. We assume that an integer $0 < k \leq P$ exists such that the last $P - k$ principal components do not represent valuable information for our system. By definition, each principal component will only explain a very small part of the original variation contained in X. Later in this paper we will test different choices for k principal components to retain in the analysis. For now we state that if only k principal components are retained, the original observation for the P-variate vector of observations $\mathbf{x}$ can be estimated as $\hat{\mathbf{x}}$ from its $(1 \times k)$ vector of scores $\mathbf{t}$:

$$\hat{\mathbf{x}} = \mathbf{t}\mathrm{P}^T$$

$$= \sum_{i=1}^{k} t_i \mathbf{p}_i \qquad (A.5)$$

as depicted in figure A.3. For the collection of n observations for $\mathbf{x}$ in X, the matrix form is:

$$X = \sum_{i=1}^{k} \mathrm{T}_k \mathrm{P}_k^T + \mathrm{E}_k \qquad (A.6)$$

where T_k is the $(n \times k)$ matrix of scores, P_k is the $(P \times k)$ matrix of loadings when only k principal components are retained and E_k is the $(n \times P)$ error

Figure A.3: Interpretation for the PCA estimate of a vector of observations

matrix. E_k can be seen as the collection of error vectors $\mathbf{e}$, each corresponding to the vector $\mathbf{x}$ when only k principal components are retained:

$$\mathbf{e} = \mathbf{x} - \hat{\mathbf{x}}$$

$$= \sum_{i=k+1}^{P} t_i \mathbf{p}_i \tag{A.7}$$

A.3   Geometrical interpretation of a principal component analysis.
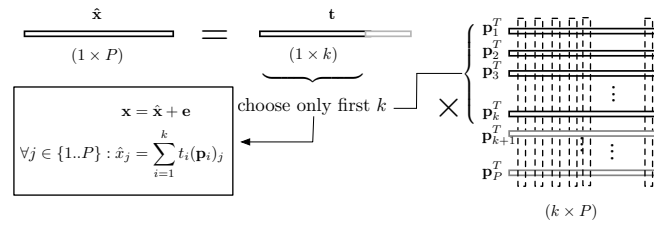
Suppose we have a vector of two observations $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 \end{bmatrix}$, for which $n = 100$ samples are taken. Now let us assume a linear relation between x_1 and x_2, with a small noise component added: $x_2 = x_1 + e$, where $e \sim \mathrm{N}(0, \sigma_e)$ is normally distributed with zero mean and standard deviation σ_e. Figure A.4 shows this situation on the left, with values for x_1 uniformly sampled between 0 and 10 and $\sigma_e = 0.5$. The matrix of loadings P is produced by the SVD algorithm, applied to the matrix X. X is the matrix composed out of 100 rows, which are the independent observations for the vector $\begin{bmatrix} x_1 & x_2 \end{bmatrix}$. The diagonal matrix L also presented here, but the matrix U is left out due to its size.

$$\mathrm{P} = \begin{bmatrix} .7071 & -.7071 \\ .7071 & .7071 \end{bmatrix} \qquad \mathrm{L} = \begin{bmatrix} 83.51 & 0 \\ 0 & 3.60 \end{bmatrix}$$

A closer study of P reveals that it is very similar to the rotation matrix of a counter-clockwise rotation over an angle $\theta = \pi/4$ in the cartesian $x_1 x_2$-plane.

$$\mathrm{P} \approx \begin{bmatrix} \sqrt{2}/2 & -\sqrt{2}/2 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix} = \begin{bmatrix} \cos \pi/4 & -\sin \pi/4 \\ \sin \pi/4 & \cos \pi/4 \end{bmatrix}$$

Effectively, the PCA decomposition finds the direction of maximal variance ($x_2 = x_1$ in this case, "Varimax rotation" [10]) and looks for an additional direction perpendicular to this. These directions are depicted as arrows on the left pane of figure A.4. The matrix of scores T can now be interpreted as the two-dimensional coordinates of the points in this new rotated coordinate space, as displayed on the right-hand side of figure A.4. It should be noted that the arrows representing the principal components are scaled in length to match the standard deviation explained by each, found from applying equation A3 in Appendix A.2 to the matrix L ($s_{\mathbf{t}_1} = 8.40$, $s_{\mathbf{t}_2} = 0.36$).
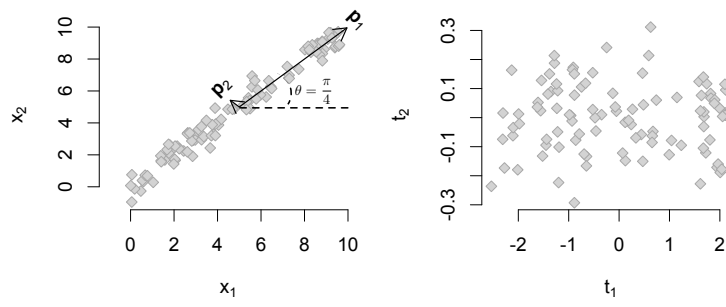
Figure A.4: Geometrical interpretation of PCA

A.4   Tolerance limits for the multivariate schedule control metrics

Hotelling's T^2 statistic is conjectured to scale with an F distribution according to [11]. Tolerance limits for this schedule control metric can be found using:

$$T_\alpha^2(k) = \frac{(n-1)(n+1)k}{n(n-k)} F_\alpha(k, n-k) \tag{A.8}$$

where $F_\alpha(k, n-k)$ is the critical value for an F distribution with k and $n-k$ degrees of freedom and a tolerance level α.

[11] has also shown that SPE metric follows a weighted chi-squared distribution $g\chi_h^2$, within a reasonable approximation, if the assumption for the multivariate normality of **x** holds. The weight (g) and the degrees of freedom (h) of $g\chi_h^2$ can then be both expressed as functions of the loadings matrix P (in equation 1). For convenience and with a proven effectiveness, g and h can also be estimated based on the method of matching moments between a $g\chi_h^2$ distribution and the distribution of SPE for the reference matrix. The mean and variance of the $g\chi_h^2$ ($\mu = gh, \sigma^2 = 2g^2h$) are then estimated by the sample mean (m) and variance (v) of the SPE sample, provided that the number of SPE observations is sufficiently large. A tolerance limit for SPE with a tolerance level α can then be found using:

$$SPE_\alpha = (v/2m)\chi_{2m^2/v, \alpha}^2 \tag{A.9}$$

where $\chi_{2m^2/v, \alpha}^2$ is the critical value of the chi-squared distributed variable with $2m^2/v$ degrees of freedom at a tolerance level α.

A.5 Illustrative example for multivariate approaches for statistical project control

In order to illustrate the use of the Hotelling's T^2 and the SPE metrics for top down project schedule control, we simulate the project network depicted in figure A.5 [12, 13]. The project consists of 10 non-dummy activities, with the predefined baseline duration estimate ($\hat{d}_i | i \in 1..10$, in days) and the budget at completion ($\mathrm{BAC}_i | i \in 1..10$) denoted respectively above and below the nodes in the network of figure A.5. It should be noted that the calculations presented for this example may be subject to rounding errors. For the sake of readability, rounding to two significant digits has been performed in the matrix calculations, presented in this appendix.

To illustrate the mechanisms presented in this paper, we define the state of activity level schedule control from the 10 fictitious project executions presented in table A.1. For each of these executions of the project, the activity durations ($d_i | i \in 1..10$), the project makespan and the K-S statistic are displayed. Given a level of significance $\eta = 0.001$ and a corresponding critical value K-S$_\eta = 0.65$ [14], we allow all 10 fictitious project executions to be included in the schedule control reference.

Table A.2 displays the EV along the project duration (in days). Each activity is assumed to follow a linear earned value accrue. The EV for each individual activity will start at 0, at its actual start time, and will progress linearly towards its budget at completion (BAC_i) when the activity is finished.

The metrics designed for schedule control in the earned value and earned schedule methodology (SV, SPI, SV(t), SPI(t)) can all be calculated from the EV and PV time series displayed in table A.2. For more details regarding their calculations, the reader is referred to the detailed works by [15] and [16]. In order to structure the matrix X to be used in a PCA decomposition, the measurements taken after each day need to be transformed to measurements equally spaced along a PC axis. For the sake of the readability of this example, and to keep the data in a presentable format, we will only use SPI and SPI(t) at $K = 4$ distinct PC instances of the project. The transformed data can be found in table A.3. The two last rows of table A.3 present the sample mean and sample standard deviation for each of the columns. If the data of table A.3 are normalized column-wise, using the sample estimates for the mean and standard deviation, they can be restructured into the matrix X in equation A.10. The matrix unfolding using $X_{\kappa,j}$ with $\kappa \in \{1, 2, 3, 4\}$ and $j \in \{\mathrm{SPI}, \mathrm{SPI}(t)\}$ was introduced in section 3.2.

The PCA in equation A.10 was performed in the statistical programming

language R [17] using singular value decomposition. Both the matrix of scores T and the matrix of loadings P were obtained directly. This example handles only a restricted number of variables (8) and its purpose is foremost to provide the reader with a hands-on illustration. Therefore, we chose the number of principal components to retain in the further analysis as two ($k = 2$), in order to be able to present our calculations in a presentable format.

Equation A.11 presents the calculations needed to derive the T^2 and SPE value for the row printed in bold font. If this is done for all the fictitious executions in the schedule control reference, the values presented in table A.4 are found. Figures A.6 and A.7 present the histograms for the calculated values of respectively the T^2 and the SPE metric. These histograms (or alternatively, the empirical cumulative distribution functions) allow us to derive the tolerance limit by using the αth quantiles. In the literature on batch process control, the tolerance limits are often derived from theoretical distribution functions (see Appendix A.4). We present the scaled F distribution and weighted Chi-squared distribution along with the histograms of respectively the T^2 and SPE schedule control metric. We have chosen not to incorporate these theoretical tolerance limits in our approach, since we do not want to impose restrictions with respect to the multivariate normality of $\mathbf{x}$ onto our project control procedure. Figure A.7 shows how the sample of observed SPE values is not likely to have come from the theoretically predicted weighted Chi-squared distribution. The proposed empirical tolerance limits are also presented in table A.4, with $\alpha = 0.95$. The theoretical tolerances are included between brackets for the sake of completeness.

In the remainder of this section, we will introduce an additional project execution that has a value for the K-S statistic that is higher than the critical value. In other words, the new project execution does not conform to our specified state of schedule control. We will illustrate how the T^2 and the SPE metric can be used to detect this at the highest level of the WBS.

A.5.1   Post-project schedule control inference

Table A.5 presents the activity durations for an additional fictitious project execution. Note that the K-S statistic for this fictitious execution (0.9) is much higher than the critical value established in our schedule control reference. The measurements of the project EV after each day are given in table A.7. In order to be used as a new vector of observations $\mathbf{x}_{new}$, the SPI and SPI(t) values have to be calculated, and subsequently transformed to the PC-dependent format. This results in the values found in table A.8. The two bottom rows of table A.8 are copied from the schedule control

reference data of table A.3. We re-use these sample estimates to normalize the data from the additional fictitious project execution that is under study. This results in the $\mathbf{x}_{new}$ vector in equation A.12 on page 267. Using $\mathbf{x}_{new}$ and the matrix of loadings P (with $k = 2$), we can calculate the scores for this new vector of observations and its corresponding T^2 and SPE values.

The values calculated in equation A.11 for the T^2 and SPE metric are 2.3 and 3.8 respectively. Only the value calculated for the SPE metric exceeds the $\alpha = 0.95$ tolerance limit. For this illustrative example, after completion of the project, we can correctly deduce from the measured SPE that the underlying activity level performance does not conform to the pre-defined state of schedule control.

A.5.2 Dynamic schedule control inference

We now investigate whether this conclusion could have been drawn when the project was not yet fully completed ($PC = 0.6$). The vector of observations $\mathbf{x}_{new,\kappa}$ has $1/4$ of its values missing at that point ($\kappa = 3$), as illustrated at the top of equation A.13. For this dynamic monitoring of project performance, we use the estimation procedure outlined in section 4.2. Conditional mean replacement provides us with an estimate for the scores, from which the T^2 and SPE values can be derived.

It is important to note that these values for the T^2 and the SPE metrics can not be referenced directly to the schedule control reference values shown in figure A.6 and A.7. We need to calculate a reference for the estimated T^2 and SPE, handling the 10 fictitious project executions from the matrix X as if they would also have the same measurements missing. This reference (at $\kappa = 3$) is presented in table A.6.

On the basis of the calculated values for the T^2 and the SPE metric in equation A.13, we can once more conclude that the SPE metric allows a correct inference to be made of the activity level performance, even when the project is only 60% completed.

$$
\mathbf{X} = \begin{bmatrix}
1.4 & 0.7 & 1.4 & 0.8 & 1.1 & 1.1 & 0.6 & 0.7 \\
-0.9 & -2.2 & -0.6 & -1.4 & -1.3 & -1.6 & -1.7 & -2.1 \\
0 & -0.1 & -0.1 & -0.2 & -0.4 & -0.2 & -0.1 & 0.1 \\
-0.5 & -0.2 & -0.6 & -0.3 & -0.4 & -0.1 & -0.1 & 0 \\
1.1 & 1.3 & 1.1 & 1.6 & 1.2 & 1.7 & 1.8 & 1.2 \\
-1.1 & -0.8 & -1.1 & -0.9 & -0.6 & -0.6 & -0.3 & -0.2 \\
1.5 & 1 & 1.5 & 1.5 & -1.2 & -1.1 & -0.9 & -0.8 \\
0.2 & -0.2 & 0.1 & -0.2 & -0.5 & -0.3 & -0.6 & -0.5 \\
-1.1 & 0.1 & -1.1 & -0.9 & 1 & 0.5 & 0.8 & 0.9 \\
-0.4 & 0.3 & -0.4 & -0.2 & 1 & 0.7 & 0.7 & 0.9
\end{bmatrix}
$$

$$
= \begin{bmatrix}
-2.7 & -0.6 & -0.8 & -0.2 & -0.2 & 0 & 0.1 & 0.1 \\
4.5 & -0.5 & -0.9 & 0.2 & 0.2 & 0.1 & 0 & -0.3 \\
0.7 & 0.5 & 0.2 & 0 & 0.1 & 0 & 0.1 & 0 \\
-4.1 & 1 & 0 & 0.6 & -0.1 & 0.1 & -0.2 & 0.3 \\
\mathbf{2} & -3.5 & 0.3 & 0.1 & 0.1 & 0 & 0.1 & -0.1 \\
-0.4 & -0.7 & 0.5 & -0.1 & 0.1 & -0.1 & 0 & -0.2 \\
0.8 & 2.5 & -0.2 & -0.4 & -0.2 & 0.2 & 0.2 & 0.2 \\
-0.2 & 1.5 & 0.3 & -0.3 & -0.3 & 0 & 0 & 0.2
\end{bmatrix}
\times
\begin{bmatrix}
-0.3 & -0.3 & -0.4 & -0.4 & -0.3 & 0.1 & 0.3 & 0.6 \\
-0.5 & -0.5 & -0.1 & -0.4 & 0.4 & 0.3 & 0 & -0.7 \\
-0.3 & -0.3 & 0.6 & 0.2 & -0.4 & -0.3 & 0.1 & 0.3 \\
-0.1 & -0.1 & -0.4 & 0.3 & 0.6 & 0.1 & -0.5 & -0.3 \\
-0.3 & -0.1 & 0.2 & 0.3 & 0.6 & -0.4 & 0.3 & -0.5 \\
0.1 & 0.3 & -0.4 & 0 & -0.7 & 0.4 & 0.4 & 0.4 \\
0.3 & 0 & -0.6 & -0.1 & -0.1 & 0.6 & -0.4 & -0.4 \\
0.6 & -0.7 & -0.1 & 0.2 & 0.2 & -0.1 & 0 & 0.1
\end{bmatrix}^{T}
= \mathbf{TP}^{T} \tag{A.10}
$$

$$
\hat{\mathbf{x}} = \mathbf{t}\mathbf{P}^{T} = \begin{bmatrix} 2 & 1 \end{bmatrix} \times
\begin{bmatrix}
-0.3 & -0.5 \\
-0.3 & -0.5 \\
-0.4 & -0.1 \\
-0.4 & -0.4 \\
-0.3 & 0.4 \\
-0.4 & 0.4 \\
-0.4 & 0.3 \\
-0.4 & 0.3
\end{bmatrix}^{T}
$$

$$
= \begin{bmatrix} \mathbf{-1.1} & \mathbf{-1.1} & \mathbf{-0.9} & \mathbf{-1.2} & \mathbf{-0.2} & \mathbf{-0.5} & \mathbf{-0.5} & \mathbf{-0.5} \end{bmatrix} \tag{A.11}
$$

$$
\implies \quad T^{2} = \sum_{i=1}^{k=2} \left(\frac{[2 \;\; 1]_{i}}{[2.3 \;\; 1.5]_{i}} \right)^{2} = 1.2
$$

$$
SPE = \|\mathbf{e}\|^{2} = \|\mathbf{x} - \hat{\mathbf{x}}\|^{2} = \| \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0.1} & \mathbf{0.3} & \mathbf{-0.4} & \mathbf{-0.1} & \mathbf{0.2} & \mathbf{0.3} \end{bmatrix} \|^{2}
$$

$$
= 0.4
$$

$$\mathbf{x}_{new} = \begin{bmatrix} -1.7 & -1.6 & 0 & -1.1 & 1 & 0.6 & -1.1 & -1 \end{bmatrix}$$

$$\mathbf{t}_{new} = \mathbf{x}_{new}P = \begin{bmatrix} -1.7 & -1.6 & 0 & -1.1 & 1 & 0.6 & -1.1 & -1 \end{bmatrix} \times \begin{bmatrix} -0.3 & -0.5 \\ -0.3 & -0.5 \\ -0.4 & -0.1 \\ -0.4 & -0.4 \\ -0.3 & 0.4 \\ -0.3 & 0.3 \\ -0.4 & 0.3 \\ -0.4 & 0.3 \end{bmatrix}$$

$$= \begin{bmatrix} 1.7 & 2 \end{bmatrix}$$

$$\hat{\mathbf{x}}_{new} = \mathbf{t}_{new}P^T = \begin{bmatrix} 1.7 & 2 \end{bmatrix} \times \begin{bmatrix} -0.3 & -0.5 \\ -0.3 & -0.5 \\ -0.4 & -0.1 \\ -0.4 & -0.4 \\ -0.3 & 0.4 \\ -0.3 & 0.3 \\ -0.4 & 0.3 \\ -0.4 & 0.3 \end{bmatrix}^T$$

$$= \begin{bmatrix} -1.5 & -1.5 & -0.9 & -1.5 & 0.3 & -0.1 & -0.1 & -0.1 \end{bmatrix} \implies \begin{matrix} T^2_{new} = 2.3 \\ SPE_{new} = 3.8 \end{matrix} \tag{A.12}$$

$$\mathbf{x}_{new,\kappa} = \begin{bmatrix} -1.7 & -1.6 & 0 & -1.1 & 1 & 0.6 & - & - \end{bmatrix}$$

$$\mathbf{x}^* = \begin{bmatrix} -1.7 & -1.6 \end{bmatrix}$$

$$P = \begin{bmatrix} P^* \\ P\# \end{bmatrix} = \begin{bmatrix} -0.3 & -0.5 \\ -0.3 & -0.5 \\ -0.4 & -0.1 \\ -0.4 & -0.4 \\ \hline -0.3 & 0.4 \\ -0.3 & 0.3 \\ -0.4 & 0.3 \end{bmatrix}$$

$$\Theta = \frac{1}{n-1}T^TT = \begin{bmatrix} s_{t_1}^2 & 0 \\ 0 & s_{t_2}^2 \end{bmatrix} = \begin{bmatrix} 5.3 \\ 2.2 \end{bmatrix}$$

$$\hat{\mathbf{r}} = \mathbf{x}^* S_{22}^{-1} S_{21} P\# + \mathbf{x}^* P^*$$

$$= \begin{bmatrix} 0.4 & 0.3 \end{bmatrix}$$

$$\mathbf{x}^{\hat{*}} = \hat{\mathbf{r}} P^{*T}$$

$$= \begin{bmatrix} -1.6 & -1.6 & -0.5 & -1.4 & 1.1 & 0.7 & 0.7 \end{bmatrix}$$

$$T^2 = 4$$

$$SPE = \|\mathbf{x}^* - \hat{\mathbf{x}}^*\|^2$$

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \implies$$

$$S_{21} = \begin{bmatrix} 0.3 & 0.3 \\ 0.3 & 0.3 \\ 0.8 & 0.6 \\ 0.6 & 0.6 \\ 0.9 & 0.9 \\ 1 & 1 \end{bmatrix}$$

$$S_{22} = \begin{bmatrix} 1 & 1 & 0.7 & 0.7 & 1.1 \\ 1 & 0.7 & 0.7 & 1.1 & 1.1 \\ 0.7 & 0.7 & 0.9 & 0.9 & 0.5 \\ 1.1 & 1.1 & 0.9 & 1.2 & 0.3 \\ 0.3 & 0 & 0.5 & 0.3 & 0.8 \\ 0.3 & 0.3 & 0.8 & 0.6 & 0.9 \end{bmatrix}$$

$$= \|\begin{bmatrix} -0.1 & 0 & 0.5 & 0.3 & -0.1 & -0.1 & -1.8 & -1.7 \end{bmatrix}\|^2$$

$$= 6.5 \tag{A.13}$$

Figure A.5: An illustrative project network (Source: Vanhoucke (2010))

Table A.1: 10 fictitious project executions to represent the first phase of the Monte Carlo experiment

Activity	2	3	4	5	6	7	8	9	10	11	K-S	RD
1	3	8	1	5	4	1	9	6	2	4	0.18	15
2	5	6	1	5	6	1	7	6	3	3	0.12	16
3	4	7	1	4	5	1	6	7	2	3	0.22	15
4	4	12	1	3	5	1	9	8	4	2	0.21	16
5	3	11	1	3	4	1	8	10	2	4	0.19	16
6	**5**	**8**	**1**	**3**	**5**	**1**	**6**	**9**	**3**	**3**	**0.16**	**17**
7	3	6	1	4	6	1	7	7	2	4	0.10	15
8	4	6	1	5	5	1	8	8	2	3	0.12	17
9	5	9	1	3	4	1	7	8	3	2	0.12	16
10	4	12	1	4	4	1	7	6	4	2	0.13	16

**Scaled F distribution
tolerance limits**
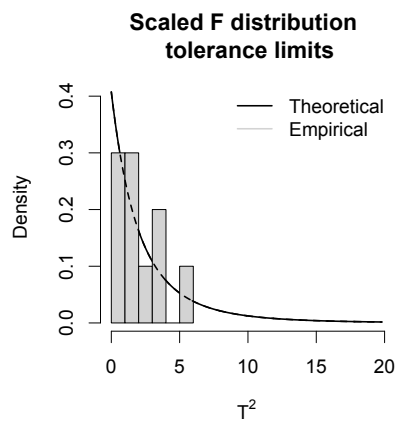
Figure A.6: Finding tolerance limits for
the T^2 schedule control metric

**Weighted Chi-squared distribution
tolerance limits**
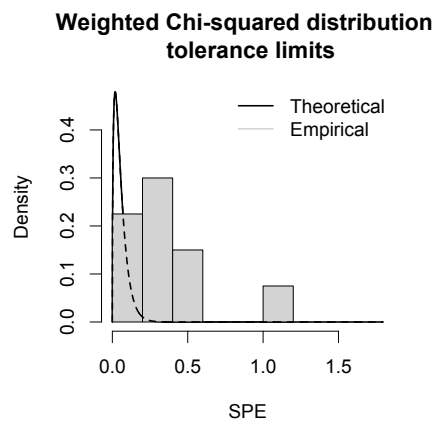
Figure A.7: Finding tolerance limits for
the SPE schedule control metric

Table A.2: The Earned Value (EV) and Planned Value (PV) along the project duration

Scenario	1	2	3	4	5	6	7	8
1	56	92	128	154	180	320	358	396
2	47	72	98	123	148	174	195	333
3	50	79	108	138	167	196	341	388
4	48	75	102	129	163	196	346	382
5	55	90	125	160	195	345	366	388
6	**46**	**70**	**94**	**119**	**144**	**179**	**330**	**384**
7	58	94	130	159	188	218	244	368
8	51	81	111	141	167	193	332	374
9	45	70	95	120	145	296	359	423
10	48	76	104	132	160	303	359	415
PV	49	77	105	133	161	189	333	380

Scenario	9	10	11	12	13	14	15	16	17
1	417	439	443	446	450	454	456		
2	374	413	437	442	446	450	455	456	
3	392	416	441	446	450	455	456		
4	419	425	431	438	442	447	452	456	
5	411	437	444	448	452	453	454	455	456
6	**422**	**429**	**436**	**442**	**448**	**452**	**455**	**456**	
7	387	406	426	445	449	453	456		
8	412	436	440	443	447	450	454	455	456
9	431	437	443	449	453	454	455	456	
10	422	430	437	444	450	453	455	456	
PV	408	434	440	446	450	454	455	456	

Table A.3: The SPI and SPI(t) for PC intervals after interpolation

	SPI				SPI(t)			
PC	**20%**	**40%**	**60%**	**80%**	**20%**	**40%**	**60%**	**80%**
1	1.197	1.127	1.502	1.068	1.273	1.133	1.145	1.074
2	0.933	0.782	0.751	0.908	0.914	0.89	0.87	0.875
3	1.031	1.036	1.03	1.022	1.041	1.023	1.017	1.029
4	0.972	1.027	1.039	1.022	0.964	1.01	1.025	1.024
5	1.168	1.206	1.532	1.151	1.23	1.219	1.208	1.109
6	**0.901**	**0.946**	**0.974**	**1.004**	**0.875**	**0.94**	**0.975**	**1.012**
7	1.214	1.175	0.789	0.962	1.298	1.203	0.925	0.967
8	1.054	1.028	1.007	0.986	1.071	1.02	1.001	0.986
9	0.905	1.065	1.467	1.082	0.88	0.945	1.089	1.09
10	0.989	1.089	1.478	1.08	0.985	1.015	1.103	1.087
$\bar{\mathbf{x}}_{\kappa,j}$	1.036	1.048	1.157	1.028	1.053	1.04	1.036	1.025
$s_{\mathbf{x}_{\kappa,j}}$	0.119	0.121	0.306	0.07	0.161	0.111	0.102	0.071

	Fictitious project executions										Tolerance limits $\alpha = 0.95$	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**		
T^2	1.54	3.94	0.03	0.2	3.18	1.2	5.47	0.34	2.79	1.19	4.78	(11.03)*
SPE	1.08	0.59	0.16	0.24	0.37	0.4	0.58	0.1	0.21	0.06	0.85	(0.98)**

* α^{th} quantile from a weighted (2.457) F distribution, d.f. 2 and 8

** α^{th} quantile from a weighted (0.125) Chi-squared distribution, d.f. 3.033

Table A.4: T^2 and SPE for the control reference

Table A.5: An additional out-of-control fictitious project execution

Activity	2	3	4	5	6	7	8	9	10	11	K-S	RD
	5	15	2	5	3	1	5	11	2	3	0.9	21

	Fictitious project executions										Tolerance limits $\alpha = 0.95$
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	
T^2	1.86	3.51	0.07	0.2	3.38	1.21	6	0.18	2.35	1.14	4.87 (11.03)*
SPE	1.4	0.58	0.21	0.24	0.46	0.54	0.73	0.12	0.26	0.06	0.86 (1.24)**

* α^{th} quantile from a weighted (2.457) F distribution, d.f. 2 and 8

** α^{th} quantile from a weighted (0.168) Chi-squared distribution, d.f. 2.733

Table A.6: T^2 and SPE for the control reference at $\kappa = 3$

Table A.7: The Earned Value (EV) along the project duration (in days) for the additional fictitious execution

1	2	3	4	5	6	7	8	9	10
31	62	87	112	137	275	318	361	403	426

11	12	13	14	15	16	17	18	19	20	21
433	436	439	443	446	449	453	454	455	455	456

Table A.8: The SPI and SPI(t) at PC for the additional fictitious execution

	SPI				SPI(t)			
PC	**20%**	**40%**	**60%**	**80%**	**20%**	**40%**	**60%**	**80%**
	0.83	1.05	1.449	0.953	0.789	0.918	1.096	0.952
$\bar{\mathbf{x}}_{\kappa,j}$	1.036	1.048	1.157	1.028	1.053	1.04	1.036	1.025
$s_{\mathbf{x}_{\kappa,j}}$	0.119	0.121	0.306	0.07	0.161	0.111	0.102	0.071

A.6   Introduction to kernel transformations for multivariate regression models

We will describe the kernel transformation, in general, as a transformation of the input space ($\mathbb{R}^{4K}$ is the space of all $1 \times 4K$ real-valued vectors of observations $\mathbf{x}$) to a generic feature space F.

$$\Phi : \mathbf{x} \in \mathbb{R}^{4K} \to \Phi(\mathbf{x}) \in F$$

The reader should note that this feature space F can have an arbitrarily large, possibly infinite, dimensionality, and that $\Phi(\mathbf{x})$ is used to represents the map of the vector $\mathbf{x}$ into this space F. The explicit calculation of the mapping of the complete input space onto a possibly infinite dimensional feature space, at first glance, seems very inopportune from a computational perspective. However, the properties the PCR and PLSR algorithms do not require this. Since we are only interested in the n observations that were made in the input space, we can restrict ourselfs to finding a solution to the regression problem in the span of these observations in the input space $\Phi(\mathrm{X}_{i.}), \forall i \in 1 \ldots n$, where $\mathrm{X}_{i.}$ represents the ith row of the matrix X. It is therefore only required that the dot product $(\Phi(\mathrm{X}_{i.}) \cdot \Phi(\mathrm{X}_{j.}))$ can be calculated in F for all $i, j \in 1 \ldots n$, without explicitly calculating the map Φ. Consequently, we use the kernel representation of the form

$$k(\mathbf{x}, \mathbf{x}') = (\Phi(\mathbf{x}) \cdot \Phi(\mathbf{x}'))$$

for two vector $\mathbf{x}, \mathbf{x}' \in \mathbb{R}^{4K}$, in order to drastically simplify the calculations in kPCR and kPLSR. Now, all occurrences of dot products in the algorithms can be substituted by a priori chosen kernel function. In this study, we will only consider the polynomial and radial base function (rbf) kernels.

$$k(\mathbf{x}, \mathbf{x}') = (\mathbf{x}\mathbf{x}'^{T})^{d} \qquad \text{polynomial kernel} \qquad \text{(A.14a)}$$

$$k(\mathbf{x}, \mathbf{x}') = e^{-\frac{\|\mathbf{x} - \mathbf{x}'\|^{2}}{c}} \qquad \text{rbf kernel} \qquad \text{(A.14b)}$$

Since, for these kernels, it has been shown that for all possible values of their parameter (respectively d and c) k is continuous kernel of a positive integral operator which acts as a dot product in a feature space F (Mercer's theorem [18]). Consequently, the kPCR and kPLSR algorithms are not too different from their linear counterparts, except for that they start from the kernel Gram matrix G of the cross dot products between all mapped input data points instead of starting from the matrix X in the input space. Let Φ denote the matrix for which the ith row represents the map of the ith row of X: $\Phi_{i.} = \Phi(\mathrm{X}_{i.}), \forall i \in 1 \ldots n$. We can write

$$\mathrm{G} = \Phi\Phi^{T}$$

and consequently,

$$\mathrm{G}_{ij} = k(\mathrm{X}_{i\cdot}, \mathrm{X}_{j\cdot}) \qquad\qquad \forall i, j \in 1 \dots n$$

In the following paragraphs, we will show how the multivariate regression models are implemented in our research. We start with the PCR and PCR-PC in section A.7, where the latter can be considered as a straightforward extension of the former. Subsequently, the kPCR algorithm is presented in section A.7. These algorithms, which are based on principal components analyses, are illustrated structurally in figure A.8. They are characterised by a separate modelling of the outer relations (the data structure of X or G and Y) and the inner relations (dependencies between Y and the input space). The multivariate regression techniques, which are based on partial least squares (PLSR and kPLSR) combine these two steps in one integrated modelling of both the input space (X or G) and the output space (Y). We will discuss these respectively in sections A.7 and A.7.

A.7 Details on the multivariate regression models

This section presents more details on the multivariate regression models that were applied in our research on project schedule control. The following section should therefore be consulted for the mathematical formulations of how respectively the PCR and PCR-PC, kPCR, PLSR and kPLSR methods were implemented in our research. These aforementioned multivariate regression methods were used to estimate the activity durations in a new project execution, from a vector of EVM/ES observations that are made at a high level of the WBS. The algorithms in these appendices were implemented in the statistical programming language R [17].

PCR and PCR-PC

In this section, we will discuss the PCR and PCR-PC regression models sequentially. In these multivariate regression models, which are based on principal components, the outer and inner relations are modelled separately. This modelling for both PCR and PCR-PC are represented graphically in part A of figure A.8.

PCR

In principal component regression (PCR), the structure of the input space is analysed using a principal component decomposition of X. In earlier literature on PCR, this composition was performed using the non-iterative partial least squares (NIPALS) algortihm [6]. Therefore, a vector of loadings $\mathbf{p}$ is first initialised, which results in a vector of scores $\mathbf{t}$ after projection of X onto this vector of loadings $\mathbf{p}$. Again, from $\mathbf{p}$, a vector $\mathbf{t}'$ can then be produced. This procedure should be repeated until convergence is reached, or more precisely until $\|\mathbf{t} - \mathbf{t}'\|$ becomes sufficiently small (we apply a threshold of 10^{-5} [19]). This iterative sub-procedure is represented by the arrows between $\mathbf{t}$ and $\mathbf{p}$ in figure A.8. Upon convergence, $\mathbf{p}$ represents the first principal component of X and the matrix X should be deflated as follows $E = X - \mathbf{t}\mathbf{p}^T$. The next principal component is then found by proceeding with E instead of X. In total, for a $1 \times 4K$ matrix X, $4K$ principal components can be found. When structured in a matrix of loadings P, the principal component decomposition of X can be written as $X = TP^T$, where T is a $n \times 4K$ matrix of scores and P a $4K \times 4K$ matrix of loadings.

It should be noted that not all $4K$ principal components should be retained in the decomposition. It is conjectured that a number l exists for which all valuable information in contained within the first l principal components of a matrix. Hence, in order to remove the effects of collinearity in the data

and to remove noise from the system, only the first l principal components should be retained. An estimate for X can then be obtained from these first l components by $\widehat{X} = T_l P_l{}^T$. Moreover, although the NIPALS algorithm provides valuable insights into the dynamics and properties of a principal component decomposition, more numerically efficient algorithms have been proposed. In our research, we have implemented a singular value decomposition (SVD) to calculate the principal components of the matrix X, although numerous applications of the QR decomposition for principal components can also be found in the literature.

In PCR, the outer relation between Y and the scores of X is modelled using multiple linear regression. Each column of Y is regressed against the $n \times l$ matrix of scores T_l. If the jth column of Y is represented by $Y_{\cdot j}$, the coefficients for the regression of this column can be found from $(T_l{}^T T_l)^{-1} T_l{}^T Y_{\cdot j}$. The reader should note that the calculation of the inverse is now numerically stable due to the removal of collinearities and the fact that the columns of T_l are independent [4].

As a summary, an estimate for the jth activity duration of a new project execution $\widehat{\mathbf{y}^*}$ can then be obtained from a new vector of observations $\mathbf{x}^*$ as:

$$(\widehat{\mathbf{y}^*})_j = \mathbf{t}_l^* (T_l{}^T T_l)^{-1} T_l{}^T Y_{\cdot j} \qquad \forall j \in 1 \dots N \qquad (A.15)$$

with: $\qquad \mathbf{t}_l^* = \mathbf{x}^* P_l$

The reader should note that we have omitted the index κ from the formulae presented above, in order not to aggravate the notations used in this section. Implicitly however, we have assumed that all calculations are made at a review period κ and include only the data that is available for the project up to its current percentage complete at a review period κ. We will continue this practice for the remaining multivariate regression models that are presented hereafter.

PCR-PC

In PCR-PC, the procedure that is followed is the same as in PCR, but an additional matrix decomposition is performed for the $\mathbf{y}$ variable in the output space. Part A of figure A.8 shows how not only a principal component decomposition can be done for X, but also Y can be decomposed into its matrix of scores U and its matrix of loadings (principal components) Q. As activity durations in practice can be often assumed to contain dependencies, it is a logical consequence that the structure of the matrix Y should be analysed prior to the calculation of regression coefficients. Let m denote the number of principal components that should be retained in the decomposition of Y, then an estimate for the jth activity duration of a new project

execution can be calculated as:

$$(\widehat{\mathbf{y}^*})_j = (\mathbf{u}_m^* Q_m{}^T)_j \qquad\qquad \forall j \in 1\ldots N \qquad (A.16)$$

$$\text{with:} \qquad \mathbf{u}_m^* = \mathbf{t}_l^* (\mathrm{T}_l{}^T \mathrm{T}_l)^{-1} \mathrm{T}_l{}^T \mathrm{U}_m$$

$$\text{and} \qquad \mathbf{t}_l^* = \mathbf{x}^* \mathrm{P}_l$$

from a new vector of observations $\mathbf{x}^*$.

kPCR

In this section, we introduce the kernel variant for PCR (kPCR). Since the inclusion of a transformation to a higher, possibly infinite, dimensional feature space of X causes these methods to be conceptually more complex, the algorithmic implementations also require more calculations.

As noted earlier, the kernel algorithms start from the Gramm matrix G, rather than from X. Part B of figure A.8 shows how for G both scores and loadings should be calculated. We implemented the calculations of the principal components in the feature space F as an eigenvalue eigenvector decomposition [20]. Computationally, this can again be performed most efficiently by a SVD or QR decomposition. Let $\mathbf{v}$ denote the principal components of G which have been scaled by their standard deviations. A total of n such vectors $\mathbf{v}$ can be calculated. These constitute the matrix of loadings V. The calculations of the matrix of scores T^{k} however, is not trivial. In order to calculate the projection of the data in the input space (X) onto the new coordinate base formed by V in the feature space F, a straightforward matrix multiplication is not sufficient. In this paper, we only provide an introduction to how this can be done, but the reader is referred to Schölkopf et al. (1998) [20] for additional details. In order to calculate $\mathbf{t}^{\mathrm{k}}$, which is the score vector in the feature space of a data entry $\mathbf{x}$ in the input space, we must calculate the dot product of the vector with all n rows in X. The rth component of this vector of dot products $\mathbf{k_t}$ is $(\mathbf{k_t})_r = k(\mathbf{x}, \mathrm{X}_{r\cdot}), \forall r \in 1\ldots n$. The jth element of the score vector $\mathbf{t}^{\mathrm{k}}$ can then be found by calculating $(\mathbf{t}^{\mathrm{k}})_j = \sum_{r=1}^n (\mathbf{k^t})_r \mathrm{V}_{rj}$. If this is done for all rows $\mathrm{X}_{i\cdot} | i \in 1\ldots n$, the matrix of scores T^{k} is found. Again, we will assume that a number of principal components l exists for which the regression model performs optimally, such that only the first l scores should be calculated for each data entry.

An estimate for the jth activity duration of a new project execution can be calculated using the kPCR method as:

$$(\widehat{\mathbf{y}^*})_j = \mathbf{t}_l^{\mathrm{k}*} (\mathrm{T}_l^{\mathrm{k}^T} \mathrm{T}_l^{\mathrm{k}})^{-1} \mathrm{T}_l^{\mathrm{k}^T} \mathrm{Y}_{\cdot j} \qquad \forall j \in 1\ldots N \qquad (A.17)$$

$$\text{with:} \qquad (\mathbf{t}_l^{\mathrm{k}*})_{j'} = \sum_{r=1}^n k(\mathbf{x}^*, \mathrm{X}_{r\cdot}) \mathrm{V}_{rj'} \qquad \forall j' \in 1\ldots l$$
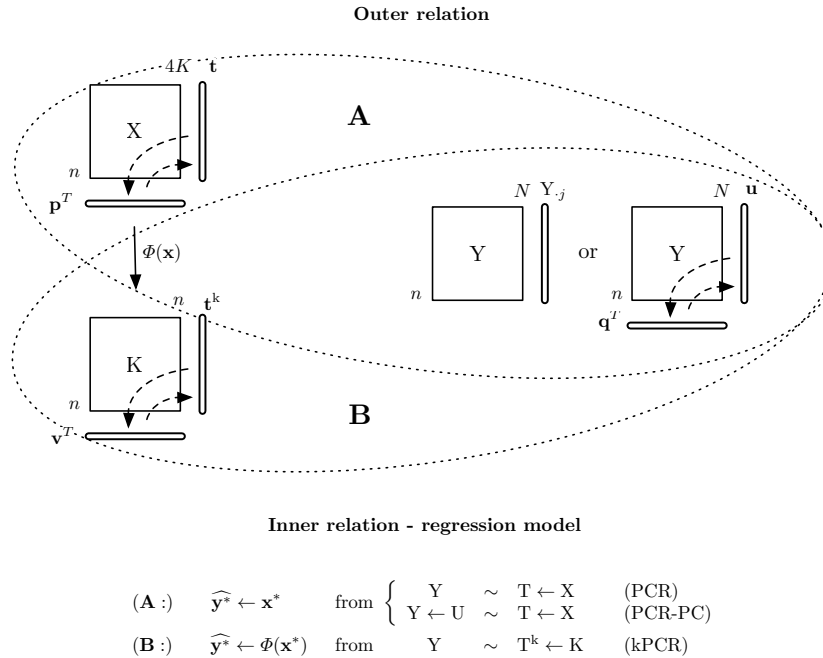
Figure A.8: Graphical representation of the PCR and kPCR algorithms

from a new vector of observations $\mathbf{x}^*$, subject to some rounding operations that have not been included in this discussion, in order not to aggravate this introduction of kPCR.

PLSR

In partial least squares regression, the calculation of scores for both the input space X and output space Y are integrated. More precisely, the inner and outer relation of the data structure is analysed simultaneously. We illustrated PLSR in part A of figure A.9.

Although numerically more efficient algorithms have been developed to calculate principal components, the NIPALS algorithm still is the primary algorithm to compute a PLSR. We implemented the procedure described by Rosipal and Trejo (2002) [21] to sequentially extract the score vectors $\mathbf{t}$, $\mathbf{u}$ and weight vectors $\mathbf{w}$, $\mathbf{c}$ from the X and Y matrices. In comparison with the PCR-PC method, introduced earlier, part A of figure A.9 shows that the calculation of the scores is combined in PSLR. In PCR-PC these were calculated separately in two distinct decompositions of the respective matrices (part A of figure A.8). The calculated scores $\mathbf{t}$ and $\mathbf{u}$ constitute the matri-

ces T and U respectively. Again, we can assume that a number l exists, which describes the amount of latent variables that should be included in the analysis. Consequently, following the description of Rosipal and Trejo (2002) [21], we can produce an estimate for the jth activity duration of a new project execution

$$(\widehat{\mathbf{y}^*})_j = \mathbf{x}^* \ \mathrm{X}^T \mathrm{U}_l (\mathrm{T}_l{}^T \mathrm{X} \mathrm{X}^T \mathrm{U}_l)^{-1} \mathrm{T}_l{}^T \mathrm{Y}_{\cdot j} \qquad \forall j \in 1 \ldots N \qquad (A.18)$$

from a new vector of observations $\mathbf{x}^*$, according to the PLSR algorithm.

kPLSR

Analogously to the PCR algorithm, the PLSR algorithm can also be applied for data from the input space that has been mapped to a larger, possibly infinite, dimensional feature space F. Similarly to the kPCR algorithm, the procedure for kPLSR also departs from the kernel Gramm matrix G and derives a non-linear regression model between the input and output matrices X and Y. We denote the matrices of scores, produced by the NIPALS algorihtm, for respectively G and Y as T^{k} and U. Consequently, we can estimate the jth activity duration of a new project execution as

$$(\widehat{\mathbf{y}^*})_j = \mathbf{k}_{\mathbf{t}}^* \ \mathrm{U}_l (\mathrm{T}_l^{\mathrm{k}}{}^T \mathrm{G} \mathrm{U}_l)^{-1} \mathrm{T}_l^{\mathrm{k}}{}^T \mathrm{Y}_{\cdot j} \qquad \forall j \in 1 \ldots N \qquad (A.19)$$
$$\text{with:} \qquad (\mathbf{k}_{\mathbf{t}}^*)_r = k(\mathbf{x}^*, \mathrm{X}_{r \cdot}) \qquad \qquad \forall r \in 1 \ldots n$$

from a new vector of observations $\mathbf{x}^*$, according to the kPLSR with l latent variables retained in the analysis.

Outer and Inner relation - regression model

$$
\begin{aligned}
(\mathbf{A}:)\quad & \widehat{\mathbf{y}^*} \leftarrow \mathbf{x}^* && \text{from}\ \ Y\ \ \sim\ \ X && \text{(PLSR)}\\
(\mathbf{B}:)\quad & \widehat{\mathbf{y}^*} \leftarrow \varPhi(\mathbf{x}^*) && \text{from}\ \ Y\ \ \sim\ \ K && \text{(kPLSR)}
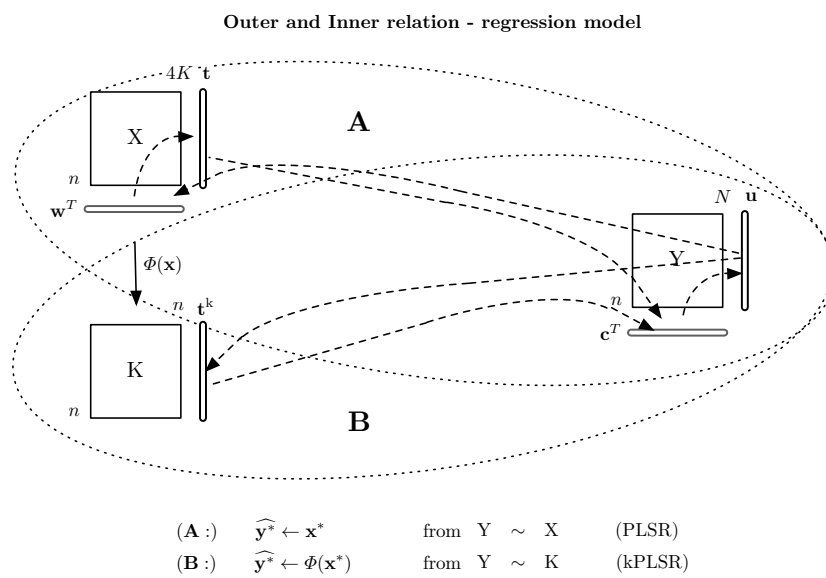\end{aligned}
$$

Figure A.9: Graphical representation of the PLSR and kPLSR algorithms

References

[1] J. Batselier, M. Vanhoucke, Construction and evaluation framework for a real-life project database, International Journal of Project Management.

[2] D. Trietsch, L. Mazmanyan, L. Govergyan, K. R. Baker, Modeling activity times by the Parkinson distribution with a lognormal core: Theory and validation, European Journal of Operational Research 216 (2012) 386–396.

[3] K. Pearson, Liii. on lines and planes of closest fit to systems of points in space, The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science 2 (11) (1901) 559–572.

[4] I. Jolliffe, Principal component analysis, Wiley Online Library, 2005.

[5] T. Kourti, Application of latent variable methods to process control and multivariate statistical process control in industry, International Journal of adaptive control and signal processing 19 (4) (2005) 213–246.

[6] H. Abdi, Partial least squares regression and projection on latent structure regression (pls regression), Wiley Interdisciplinary Reviews: Computational Statistics 2 (1) (2010) 97–106.

[7] J. Shlens, A tutorial on principal component analysis, Systems Neurobiology Laboratory, University of California at San Diego.

[8] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammerling, A. McKenney, et al., LAPACK Users' guide, Vol. 9, Siam, 1999.

[9] J. F. MacGregor, Using on-line process data to improve quality: Challenges for statisticians*, International Statistical Review 65 (3) (1997) 309–323.

[10] H. F. Kaiser, The varimax criterion for analytic rotation in factor analysis, Psychometrika 23 (3) (1958) 187–200.

[11] G. E. Box, Some theorems on quadratic forms applied in the study of analysis of variance problems, i. effect of inequality of variance in the one-way classification, The Annals of Mathematical Statistics 25 (2) (1954) 290–302.

[12] M. Vanhoucke, Measuring Time - Improving Project Performance using Earned Value Management, Vol. 136 of International Series in Operations Research and Management Science, Springer, 2010.

[13] J. Colin, M. Vanhoucke, Setting tolerance limits for statistical project control using earned value management, Omega The International Journal of Management Science 49 (2014) 107–122.

[14] G. Marsaglia, W. W. Tsang, J. Wang, Evaluating kolmogorov's distribution, Journal of Statistical Software 8 (18) (2003) 1–4.

[15] M. Vanhoucke, On the dynamic use of project performance and schedule risk information during project tracking, Omega The International Journal of Management Science 39 (2011) 416–426.

[16] W. Lipke, O. Zwikael, K. Henderson, F. Anbari, Prediction of project outcome: The application of statistical methods to earned value management and earned schedule performance indexes, International Journal of Project Management 27 (2009) 400–407.

[17] R Core Team, R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria (2013). URL http://www.R-project.org

[18] V. Vapnik, The nature of statistical learning theory, springer, 2000.

[19] J. Henseler, On the convergence of the partial least squares path modeling algorithm, Computational Statistics 25 (1) (2010) 107–120.

[20] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural computation 10 (5) (1998) 1299–1319.

[21] R. Rosipal, L. J. Trejo, Kernel partial least squares regression in reproducing kernel hilbert space, The Journal of Machine Learning Research 2 (2002) 97–123.