Improving decision making for incentivised and weather-sensitive projects

Louis-Philippe Kerkhove

Dissertation submitted to the Faculty of Economics and Business Administration, Ghent University, in fulfilment of the requirements for the degree of Doctor of Applied Economic Sciences

I don't know where I'm going from here,
but I promise it won't be boring.

DAVID BOWIE

If wisdom were offered me on the one condition that I should keep it shut away and not divulge it to anyone, I should reject it. There is no enjoying the possession of anything valuable unless one has someone to share it with.

SENECA

Advisor:

Prof. dr. Mario Vanhoucke

Doctoral jury:

Prof. dr. Marc De Clercq	Ghent University, Dean
Prof. dr. Patrick Van Kenhove	Ghent University, Academic Secretary
Prof. dr. Tarik Aouam	Ghent University
Prof. dr. Pierre Bonnal	CERN (Switzerland)
Prof. dr. Tyson Browning	Texas Christian University (USA)
Prof. dr. Dries Goossens	Ghent University
Prof. dr. Broos Maenhout	Ghent University
Prof. dr. Geert Poels	Ghent University
Prof. dr. Mario Vanhoucke	Ghent University
	Vlerick Business School
	University College London (UK)

Dankwoord

Het schrijven van een doctoraat wordt vaak aanzien als een individuele opdracht. Niets is minder waar, het opleveren van dit proefschrift zou nooit gelukt zijn zonder de begeleiding en ondersteuning van een zeer grote groep mensen. Bij deze zou ik deze personen van harte willen bedanken voor hun steun de voorbije jaren.

Eerst en vooral mijn oprechte dank aan Mario Vanhoucke, mijn promotor die mij gedurende vier jaar schitterend heeft ondersteund op academisch gebied, maar die daarnaast ook een goeie vriend is geworden. Ondanks het grote aantal doctoraatsstudenten, vakken en andere projecten stond zijn deur steeds wagenwijd open voor alle mogelijke vragen of advies. Daarnaast waren er ook veel andere momenten wanneer het allemaal niet zo serieus hoefde te zijn; op congres, op het dakterras in Lissabon, in Londen of gewoon tijdens een kort gesprek over de dingen des levens op de faculteit. Mijn oprechte dank voor alles, ik ben ervan overtuigd dat dit niet ons laatste project samen zal zijn!

Daarnaast wil ik ook alle leden van mijn examenjury van harte bedanken voor de tijd die ze investeerden om dit lijvige document te lezen, alsook voor hun constructieve commentaar dat me in staat heeft gesteld deze finale versie naar een hoger niveau te tillen. Tarik, thank you for your exceptionally thorough review of my dissertation, I have really enjoyed discussing your questions and the ideas that ensued from them. I am also grateful to have had you as a colleague in our department, and for the many pleasant conversations at multiple occasions. Broos en Dries, jullie oog voor detail tijdens de voorverdediging heeft ervoor gezorgd dat de inconsistenties uit deze versie van de tekst verdwenen zijn, waarvoor mijn oprechte dank. I am also greatly indebted to the external members of the examination committee, prof. dr. Pierre Bonnal and prof. dr. Tyson Browning for their willingness to be present at the public defence in spite of the considerable distance they had to travel, on top of their thorough analysis of the manuscript. Geert, bedankt om ondanks de zeer drukke agenda de afgelopen maanden toch bereid te zijn als lid van mijn jury op te treden. Ten laatste wil ik ook mijn dank betuigen aan prof. dr. Marc De Clercq en prof. dr. Patrick Van Kenhove die voor de verdediging respectievelijk als decaan en

academisch secretaris optraden.

Los van de schitterende academische omkadering zou dit alles natuurlijk nooit gelukt zijn zonder steun van het thuisfront. Hier zou ik eerst en vooral mijn vriendin Julie willen bedanken. Waar de stress voor de buitenwereld vaak onzichtbaar was, is dat thuis zeker niet altijd het geval geweest. Maar steeds opnieuw kon ik er op rekenen dat jij precies wist wat ik nodig had; van een avondje ontspanning, een weekendje weg tot gewoon a swift kick in the butt om weer wat forward momentum te genereren. Bedankt ook voor het organiseren van de publieke verdediging, dankzij jou wordt het een dag om nooit te vergeten. Na zeven jaar zie ik je nog elke dag liever, en ik kan niet wachten om samen aan een nieuw hoofdstuk te beginnen!

Natuurlijk zou ik ook graag mijn ouders bedanken voor alles, maar vooral ook voor de jarenlange ondersteuning tijdens mijn jaren in Gent. Pa, het hoeft voor ons niet allemaal met zo veel woorden gezegd te worden, maar we zitten toch gemakkelijk op dezelfde golflengte, eens te meer als daar ten gepaste tijde een Omer bij komt kijken. Ma, bedankt voor alles wat je de voorbije jaren hebt gedaan voor mij, waaronder ook steevast het verwerken van een enorme berg met was elk weekend opnieuw. Zonder jullie zou ik ongetwijfeld niet staan waar ik vandaag sta. Ik wil zeker ook mijn zus Louise-Marie en schoonbroer Tom bedanken, die beiden tot de beperkte groep mensen behoorden die steeds geïntereseerd waren in wat ik nu eigenlijk precies aan het doen was. Louise, het was ook steeds leuk om iemand te hebben die mijn interesse voor technologie deelt en met wie ik steeds de laatste nieuwtjes kon bespreken.

Niet te vergeten zijn ook mijn schoonouders, bij wie ik na zeven jaar kind aan huis ben geworden. Bedankt voor de vele leuke momenten doorheen al deze jaren. Chanaz, bedankt voor alles wat je dag in dag uit voor iedereen doet, van het op tafel toveren van heerlijke kookkunsten, de logistieke ondersteuning van drie gezinnen, tot het scheppen van een aangename sfeer waarin eenieder graag vertoeft - geen mens die weet hoe je er altijd toch weer tijd voor vindt. Philippe, we hebben de afgelopen jaren samen al menig etalageraam bewaakt, maar gelukkig was het vinden van gemeenschappelijke interesses om de tijd te doden zeker geen probleem. Bedankt voor alles wat ik de afgelopen jaren al van je heb mogen leren. Daarnaast wil ik ook mijn schoonzus Barbara, schoonbroer Mathieu en mijn neefje William bedanken voor hun rol wanneer de boog even mocht ontspannen. Barbara, bedankt voor het vastleggen van zo veel mooie herinneringen op de gevoelige plaat, alsook voor je aanwezigheid aan de andere kant van de camera waar je die herinneringen mee vorm helpt geven. Gelukkig zorgt Mathieu voor het broodnodige mannelijke evenwicht

wanneer vrijdagavond het weekend wordt ingezet. Mathieu, je drive om je eigen zaak uit te bouwen is een inspiratie geweest om zelf ook de stap te wagen.

Ook mijn grootouders die zowel vroeger als nu een belangrijke rol hebben gespeeld wil ik graag bedanken voor alles. Oma Agnes en opa Dirk, het doet altijd deugd om eens ongedwongen bij jullie binnen te springen en over koetjes en kalfjes te kunnen praten. Oma Lieve en opa Willy, jullie voorbeeld heeft me geleerd wat een portie gezond verstand in combinatie met de nodige werklust waard zijn in het leven.

Naast mijn grootouders zijn er ook nog enkele andere mensen die veel betekenen. Hier zou ik graag nog mijn tante Isabelle willen bedanken om steeds klaar te staan waarvoor dat ook nodig mocht zijn, en zeker ook voor de vele gezellige avonden waarop we samen met Ella van Carlos' kookkunsten mochten genieten. Daarnaast wil ik zeker ook Denise bedanken voor de levenswijsheden die ze me op jonge leeftijd reeds heeft meegegeven.

Ik heb eveneens het geluk gezegend te zijn met een zeer uitgebreide verzameling aan tantes, nonkels, (achter-)neven en (achter-)nichten, alsook een uitgebreide schoonfamilie. Een volledige oplijsting van deze mensen zou nog ettelijke pagina's toevoegen aan dit reeds uitgebreide document, vandaar vat ik graag de essentie in een zin samen; jullie zijn me allemaal dierbaar.

Voor een moment van deugddoende ontspanning kon ik ook steeds rekenen op de harde Roeselaarse kern van *Los Hombres*; Ramses, Thomas, Matthias, Wim, Daan en Glenn, maar natuurlijk ook het gezelschap van *Las Chicas*; Hannelore, Emily en Ann-Sophie valt zeker niet te vergeten. Of het nu is om een goeie IPA te drinken, een paar kleien uit de lucht te halen, of nog eens naar een keihard concert te gaan, aan partners in crime geen gebrek. Maar ook buiten de historische kern passeerden er al wat invloedrijke spelers de revue. Nikolaas, na vijftien jaar en een six-year stint in het College kennen we elkaar door en door, ik ben blij dat ik je tot mijn vrienden mag rekenen. De gezellige avonden samen met jou en Sophie zijn steevast momenten om naar uit te kijken. Dankzij mijn vriendin is mijn vriendenkring zeker ook nog enkele schitterende personen rijker geworden; Noemi, Julie, Matthias, Andy, Ann-Sophie, Andreas, Delphine, Louis, Karolien en Emiel bedankt voor de vriendschap. Ook aan mijn periode als student in Gent heb ik enkele schitterende vrienden overgehouden; Bernard, Jasper, Katrien, Roeland, Michaël, Frederick, Leen en Willem, al speelt de afstand ons soms wel parten het doet me deugd dat we er toch in slagen het contact niet te verliezen. Ook de equipe van Leweck waarmee we de laatste jaren al menig feestje hebben gebouwd, en dat in de komende jaren ongetwijfeld zullen blijven doen, wil ik hier zeker vermelden. Als laatste wil ik ook graag nog Laurent vermelden met

wie ik na deze zomer een nieuw project start, ik ben benieuwd waarheen de weg ons zal leiden en kijk alvast uit naar de toekomst.

Gedurende de vier jaar van mijn doctoraat heb ik ook het geluk gehad om met enkele schitterende mensen samen te mogen werken. Hier zou ik graag Martine en Machteld willen bedanken, eerst en vooral voor hun sprankelende aanwezigheid, maar zeker ook voor alle ondersteuning bij het geven van lessen en de hulp bij het doorworstelen van alle red tape waarmee een universiteit nu eenmaal gepaard gaat. Daarnaast wil ik ook mijn dank betuigen aan de mensen die samen met mij in de loopgraven stonden als doctoraatsonderzoekers. Jordy, Jonas en Pieter - het is al een heel eind geleden sinds die eerste les in Leuven maar ik ben blij dat ik de afgelopen vier jaar op dergelijke collega's heb mogen rekenen. Ook Mathieu die al een jaartje eerder mocht afzwaaien wil ik graag bedanken voor menig aangenaam nerd-gesprek over supercomputers tot huis-tuin en keuken-elektronica van eigen ontwerp. Doorheen de jaren kwam er ook steeds meer gezelschap bij om gelederen te versterken. Tom, Mick, Annelies en Jeroen, jullie aanwezigheid op bureau en op congres zorgde steevast voor meer leven in de brouwerij. Ook quasi-collega Vincent wil ik graag bedanken voor de hulp die hij me in der tijd heeft verschaft bij het schrijven van mijn masterthesis, ik was steeds blij je op congres nog eens tegen het lijf te lopen.

Louis-Philippe Kerkhove
9 september 2016

Nederlandse Samenvattting

Operationeel onderzoek is de wetenschap die wiskundige modellen en algoritmen gebruikt om processen binnen bedrijven en organisaties te verbeteren. Deze scriptie behandelt een specifiek domein binnen deze onderzoekstak: het beheer van projecten. Traditioneel werd binnen dit domein voornamelijk de nadruk gelegd op het optimaliseren van projectplanningen. De laatste jaren is binnen dit domein echter een steeds breder perspectief toegepast. Dit resulteerde in verschillende nieuwe technieken die beslissingsnemers beter in staat stellen om in te spelen op risico's, alsook om de voortgang van een project effectiever en efficiënter te gaan monitoren. Dit proefschrift onderzoekt twee subdomeinen die tot op heden niet of onvolledig werden behandeld in de literatuur. Hieronder wordt de inhoud van deze scriptie zeer kort samengevat, voor een uitgebreide inleiding tot de verschillende subdomeinen en de gebruikte technieken wordt verwezen naar de introductie in **hoofdstuk 1**.

Het **eerste subdomein** onderzoekt het gebruik van incentive contracten tussen de eigenaar van een project en de uitvoerende partij. Hierover gebeurt de laatste jaren steeds meer onderzoek, maar het merendeel van dit onderzoek legt hierbij de nadruk op een kwalitatieve en beschrijvende analyse. Het doel van dit onderzoek is om deze problematiek kwantitatief te modelleren om op deze manier ook een prescriptieve analyse te bekomen. Deze analyse wordt gemaakt vanuit het perspectief van zowel de eigenaar van het project alsook dat van de uitvoerende partij. Hieronder worden de doelstellingen van de verschillende hoofdstukken in dit eerste deel van het proefschrift kort samengevat.

Hoofdstuk 2 geeft een uitgebreid overzicht van de academische literatuur die is verschenen over het gebruik van incentives in project management. Op basis van dit overzicht kon worden vastgesteld dat een meer kwantitatieve analyse van het gebruik van incentives in projecten tot op vandaag nog ontbrak. Vervolgens wordt de informatie die werd verzameld gebruikt in **hoofdstuk 3**, waar een wiskundig model wordt voorgesteld om de interactie tussen de eigenaar en de uitvoerende partij te modelleren. Gebruikmakend van dit model wordt een computationeel experiment uitgevoerd waaruit enkele duidelijke management richtlijnen voor het ontwerpen van goede incentive contracten worden gedestilleerd. Hierop wordt verder gebouwd in **hoofdstuk 4**, waar een heuristische oplossingsmethode

voor het ontwerpen van contracten wordt voorgesteld. Het verschil met de eerdere analyse is dat hier voor een specifiek scenario een oplossing kan worden aangereikt, eerder dan het stellen van een aantal globale vuistregels. Vervolgens verandert het perspectief van dat van de eigenaar van het project naar dat van de partij die het project uitvoert. Voor deze partij wordt in **hoofdstuk 5** een lineair model ontworpen die hem/haar in staat stelt om de planning van een project correct te balanceren om op die manier de hoogst mogelijke totale winst te bekomen. Naast het maken van een solide planning wordt in **hoofdstuk 6** een nieuwe methode voorgesteld die de opvolging verbetert van projecten die onderworpen zijn aan incentive contracten.

Het **tweede subdomein** van dit proefschrift onderzoekt hoe weermodellen gebruikt kunnen worden om de prestaties van weergevoelige projecten te verbeteren. Tijdens de laatste decennia is er veel klimatologisch en meteorologisch onderzoek gedaan om de kwaliteit van weermodellen en weervoorspellingen te verbeteren. Daarenboven is er steeds meer data beschikbaar doordat er meer meetpunten zijn. Deze trends openen deuren voor het verbeteren van operationele beslissingsmodellen. Ondanks het bestaan van deze weermodellen en de beschikbaarheid van data zijn toepassingen binnen het domein van operationeel onderzoek nog steeds een zeldzaamheid (Regnier, 2008). Bijgevolg is het doel van het tweede deel van dit proefschrift om deze modellen te gaan gebruiken binnen een project management context. Specifiek wordt er gefocust op offshore activiteiten die sterk onderhevig zijn aan golfslag en windsterkte. Hieronder worden de doelstellingen van de verschillende hoofdstukken in het tweede deel van het proefschrift kort samengevat.

Hoofdstuk 7 geeft een overzicht van de recente vernieuwingen in het domein van weermodellering. Op basis hiervan wordt een nieuw operationeel weermodel gebouwd dat in staat is om realistisch gecorreleerde windsnelheden en golfhoogtes te simuleren. Het doel van dit model is het simuleren van realistische weerpatronen op lange termijn, eerder dan het maken van weersvoorspellingen op korte termijn. Dit weermodel wordt vervolgens gebruikt voor twee operationele toepassingen in de twee daaropvolgende hoofdstukken. In **hoofdstuk 8** wordt een nieuwe methode voorgesteld om de planning van offshore constructie projecten te optimaliseren door rekening te houden met de impact van windsnelheid en golfhoogte. Deze methode gebruikt meta-heuristische oplossingstechnieken om verschillende strategieën te testen. Het testen van een dergelijke strategie gebeurt door middel van een simulatiemodel. De voorgestelde techniek maakt eveneens gebruik van informatie over de klimatologische condities om de oplossingsruimte op een intelligente manier in te perken en op die manier een betere dekking van de meest interessante zones van de oplossingsruimte te kunnen garanderen. Vervolgens wordt in **hoofdstuk 9** het weermodel gebruikt voor het optimaliseren van het onderhoud van offshore wind turbines. De productiviteit van een windturbine neemt toe naarmate de windsnelheid stijgt. Bijgevolg is

het interessanter om onderhoud uit te voeren wanneer de windsnelheden laag zijn. Natuurlijk valt op voorhand niet te voorspellen hoe het weer zich zal gedragen (althans niet op lange termijn), bijgevolg is het complex om een goede bovengrens voor de windsnelheid vast te leggen voor onderhoudsactiviteiten. Dit hoofdstuk gebruikt het nieuwe weermodel alsook een aantal technieken uit het domein van artificiële intelligentie om dit proces te verbeteren.

Tot slot worden al de voornaamste bevindingen alsook de mogelijkheden voor toekomstig onderzoek samengevat in **hoofdstuk 10**.

Contents

List of Figures

List of Tables

1

Introduction

1.1 General Introduction

The field of project management has originated from the domain of operational research, which focuses on the mathematical optimization of operational problems. However, in recent decades an increasingly broad perspective has been applied to the field of project management. As such, project management has spawned a number of very active sub-domains, which focus not solely on the scheduling of the project's baseline, but also on the analysis of risk, as well as the controlling of project execution.

This dissertation focuses on two areas where existing literature is still lacking. The first area is the use of incentivised contractual agreements between the owner of a project, and the contractor who is hired to execute the project. Whereas this area has received growing attention in recent years, the majority of studies remained strongly descriptive. Hence, the aim of the first part of this dissertation is to develop a more prescriptive approach from both the owner's and the contractor's perspective. A brief introduction to the first part of this dissertation is given in section 1.2.

The second part of this dissertation investigates the use of dedicated weather models to improve operational performance of weather-sensitive projects. During recent decades, significant effort has been made to improve the quality of weather simulation models. Moreover, the amount of available weather data has been steadily increasing. This opens up a lot of new possibilities for using more precise weather models in order to support operational decision making. In spite of this, the number of applications of these weather models in operational research has remained rather limited (Regnier, 2008). As such, the aim of the second part of this dissertation is to leverage these weather models to improve the scheduling of offshore construction projects, as well as preventive maintenance of offshore wind turbines. A brief introduction to the second part of this dissertation is given in section 1.3.

1.2 Incentive Contracts for Projects

The main research objective of the first part of this dissertation is the creation a comprehensive decision support system to guide the use of incentive contracts in a project management context, from both the project owner's and the contractor's perspective. The adoption of increasingly complex incentive contracts in recent years has sparked practitioner's interest, and consequently more academic research has been conducted on this subject. However, research on the subject has mainly focused on very specific cases, which implies that findings are hard to generalize. Through the design of a comprehensive system, and the execution of extensive computational experiments which test the strategies of the owner as well as the contractor in a multitude of project situations, the quality of the decision making process will be improved.

Contractual agreements are made in order to govern the relationship between economic actors. This governance is especially important in situations such as project contracting, where both parties' profit maximization objectives result in a naturally adversarial relationship (Turner, 2004). To alleviate this situation and the potential agency problems it may cause, contractual agreements often include incentive clauses. These clauses are intended to align the contractor's objectives with those of the project owner.

This alignment is highly important since the manner in which the contract is executed is fully controlled by the contractor. The method of execution chosen by the contractor can be modelled as a trade-off decision which envelops the traditional iron triangle (Marques et al., 2011). This implies that the outcome of a project can be seen as a combination of cost, duration and scope. Given the interrelation of these dimensions, it is impossible to improve the performance in one of these dimensions without negatively affecting one or both of the other dimensions.

Two research domains are highly relevant to the design of incentivised contracts in a project context. The first is the literature on contracting itself which deals explicitly with the negotiation, implementation and perceived effectiveness of contracts. The second is the literature on trade-offs in project management. This is a subdomain of project management literature that aims to quantify the trade-offs present in projects. This includes modelling and optimization of projects using a wide array of techniques.

1.2.1 Core Concepts

(Project) Owner: The economic actor who desires the completion of a project, but lacks certain resources (knowledge, manpower, time...) needed to execute the project. Hence, the project owner enters a relationship with a contractor who executes the project in exchange for a financial compensation. The properties of this relation-

ship are defined in a formal contractual agreement, which may or may not include incentive clauses.

Contractor: The economic actor who agrees to execute a project in exchange for financial compensation. The magnitude of this compensation depends on the degree to which the project outcome corresponds to the outcome specified in the contractual agreement.

Project outcome: The cost, duration and scope of a project as measured upon its completion, from the perspective of the project owner. The scope of the project encompasses all types of performance measures other than time and cost, such as the amount of work performed and the quality of said work. The value of the finished project to the owner is a function of these dimensions.

Incentive: A monetary reward received by the contractor in return for behaving a certain way (i.e. the manner in which (s)he has executed the project). The magnitude of this incentive depends on the way in which the project owner perceives and values the project outcome. Hence it is linked to one or more of the outcome dimensions: cost, duration and scope.

Project scheduling: Constructing a timetable for the activities of the project to indicate when they should start and finish. One of the methods of doing this is through the selection of activity execution modes, which represent certain combinations of outcome dimensions at the activity level. An example of this is the multi-mode resource-constrained project scheduling problem, where the duration of activities can be changed by changing their renewable resource assignments.

Project risk analysis: Analysing a project schedule in order to identify critical activities and/or resources in order to discover the key risks during a project's execution.

Project control: Once the project is in progress, the deviations between the schedule and reality should be monitored using control systems. These control systems should provide timely warnings, which allow project mangers to take action and get the project back on track (Vanhoucke, 2010a).

1.2.2 Dynamics Between Owner and Subcontractor

Incentivised contractual agreements are analysed from the perspective of both the project owner and the contractor. The goal of the owner is to design the contract in a way that aligns the awarded incentives with the project outcomes, in order to maximize his expected utility and minimize the risk of conflicting objectives. From the contractor's perspective, the objective is to optimize the way in which the project is scheduled and controlled, given that (s)he is subjected to a specific set of incentive clauses.

1.2.2.1 Trade-offs

Much has been written on the trade-offs encountered when managing projects and activities. The most traditional trade-off problem being the time/cost trade-off, which was first introduced as a linear relationship (Kelley and Walker, 1959), but is assumed to be convex in more recent literature (Choi and Kwak, 2012; Shr and Chen, 2003). This two-dimensional trade-off can be extended by adding the scope of a project as a third dimension. Again, the relationship between project scope and the other project dimensions is generally assumed to be convex (El-Rayes and Kandil, 2005).

These three dimensions (cost, duration and scope) are used to reflect the outcome of the project as perceived by the project owner. However, these three dimensions are insufficient in order to fully represent the manner in which the project will be (or has been) executed, since parts of the execution strategy - which is decided by the contractor - cannot be directly perceived by the owner.

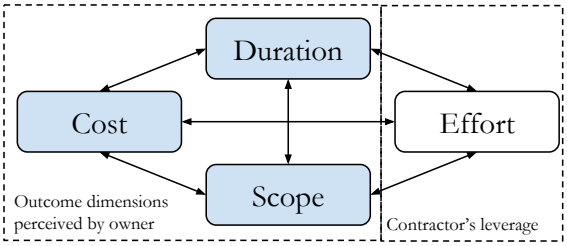


Figure 1.1: Project trade-off dimensions

The introduction of 'contractor effort' as a fourth dimension (see figure 1.1) to this trade-off makes a comprehensive representation of the subcontracted project possible. Contractor effort can be defined as investments made by the contractor in order to enhance one or more of the three outcome dimensions of the project, but to which (s)he is not contractually obliged. Naturally, the contractor hopes to earn a return on these investments by receiving a higher incentive payment. Although some authors use another nomenclature for the concept, contractor effort has already been touched upon several times in project trade-off literature.

The most basic example of contractor effort is the attention given to the project by the contractor's management team (Abu-Hijleh and Ibbs, 1989; Arditi and Yasamis, 1998). This is dubbed managerial attention and can be seen as a scarce resource since only a limited amount is available to be distributed among the contractor's project portfolio. Other examples of contractor effort include insurance policies (Chapman and Ward, 1994), the allocation of more or better skilled personnel (El-Rayes, 2001b; El-Rayes and Kandil, 2005) and hiring project specific equipment to resolve potential issues (Lee and Thomas,

2007). In line with the other dimensions, the relationship of this dimension with the other dimensions can be assumed to be convex (Bayiz and Corbett, 2005).

1.2.2.2 Incentive Contracts

The incentive clauses present in contractual agreements are always linked to one of the three outcome dimensions (cost, duration and scope) perceived by the project owner. Naturally, a single contract can include multiple incentive clauses. A distinction between contracts can also be made based on the way in which the incentive amounts are calculated. Traditionally this was done using simple linear equations, but recently piecewise linear and non-linear incentive agreements have gained popularity.

A simple example of this is a cost sharing agreement between the owner and the contractor where any deviations from the target cost are shared according to an agreed upon sharing ratio. Supposing that this ratio is equal to 50%, half of the cost overrun is to be paid for by the contractor. Inversely, should the final cost be less than the targeted amount, the owner is required to transfer half of these savings to the contractor as a reward for his/her better than expected performance.

The current literature on incentive contracts can be divided into three categories. A first body of literature examines the negotiation phase between the client and the contractor. The second provides descriptive analyses of the types of contracts used in practice, often focussing on cost incentives. Finally, a third body of research empirically evaluates the effectiveness of incentive contracts, often using surveys as a measurement tool. A detailed overview of this literature is presented in section 2.3.

This literature forms a solid base on which a quantitative model to analyse incentive contract structures can be built. As of now no such model exists, which makes an objective comparison of many contracting approaches difficult, and subjective at best. Through the creation of a more comprehensive model a solid foundation is provided, which can be used by both project owner and contractor to make informed decisions on how to manage such projects.

1.2.3 Research Objectives

Whereas the available literature clearly showcases academic and professional interest in the subject, there are still a substantial number of gaps in the literature. Specifically, the following challenges have been identified:

- Studies on contracts are highly fragmented and usually focus on one or a few contract types. There is a clear need for a more comprehensive comparison of the performance of incentive contracts.

- Current literature on the design of contracts is mainly descriptive, in that it describes which types of contracts are being used in practice and how effective these are perceived to be. However, no attempt has been made to move towards a more quantitative perspective, which is a prerequisite to providing proactive design guidelines given the properties of a project.
- The implications of incentivized objectives on the scheduling and controlling of projects have not been investigated to date.

These opportunities have been translated into a number of specific studies, which represent the chapters in the first part of this dissertation. The nature of these specific chapters can be visualised in a two-dimensional space. The first dimension being the perspective used, which can be either that of the owner or that of the contractor. The second dimension expresses the 'level' of the analysis. Specifically, whether the goal of the chapter is to provide strategic guidelines or to provide operational solutions for dealing with incentive contracts.

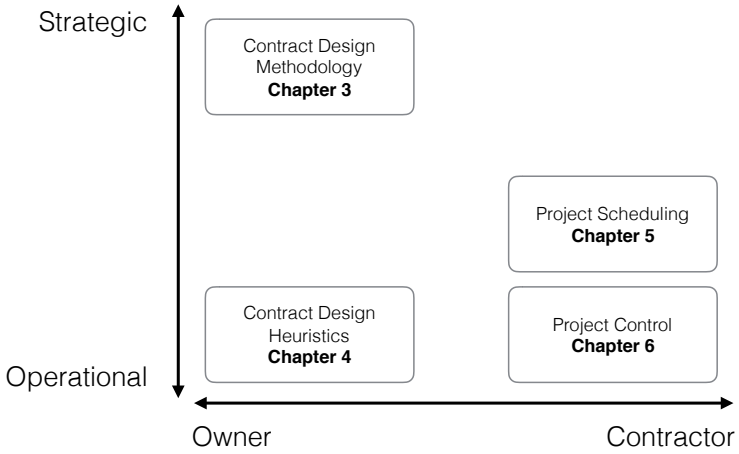


Figure 1.2: Visualisation of the chapters in the first part of this dissertation.

1.2.3.1 A Framework for the Owner's Contract Design Decision

The aim of the first study (chapter 3) is to provide a quantitative framework for incentive contract design in projects, which can be used by the project owner to select the most adequate contract for any given project environment. This quantitative framework consists of three components: a trade-off model describing the nature of the project, an evaluation model describing the valuation of the different outcomes of the project for both the owner and contractor, and a contract model which is capable of representing the majority

of (incentivised) contractual agreements used in practice. The key hypotheses that are investigated using this model are the following:

1. Some types of contract clauses are more effective than others.
2. Specific combinations of contract clauses are likely to be more effective than others.
3. The performance of specific contracts may be influenced by the nature of the project.

By answering these research questions several generalisable strategic guidelines can be provided for incentive contract design. Hence, this study is located in the upper left quadrant of figure 1.2

1.2.3.2 Designing Optimization Techniques for Contract Design

The objective of the second study (chapter 4) is to create a dedicated optimization method which is capable of constructing robust contracts for a given project. This research question also takes the project owner's perspective, but moves towards a more operational application area (i.e. dealing with a specific project rather than specifying general rules of thumb). This is done by designing optimization techniques that provide answers to specific cases, rather than providing broad managerial guidelines. Specifically, a parallel multi-objective scatter search meta-heuristic has been developed to optimise the contract structure.

1.2.3.3 Scheduling Techniques for Incentivized Projects

Next, the perspective of the contractor rather than that of the owner is analysed. The first study (chapter 5) that takes the perspective of the contractor investigates how the scheduling of a project subjected to an incentive contract can be optimized. To do this, the traditional multi-mode project scheduling problem is extended in order to accommodate the four-dimensional trade-off associated with the incentivized environment.

1.2.3.4 Controlling Incentivized Projects

Another key task of the contractor which succeeds the scheduling phase is monitoring the project execution. The key goal of chapter 6 is to adjust existing project control techniques in order to enhance the relevance of the signals they produce for an incentivized project context. This is especially important since the profit margin of the contractor, and hereby also the objectives of the contractor, are inextricably linked to the incentive structure of the projects.

1.2.4 Methodology

This section presents a very brief introduction of the methodology used in this dissertation. A distinction is made between the high-level model used to represent the project owner's contract design problem, and the more operational models used to represent the contractor's decision space. The former is summarised in section 1.2.4.1, and an overview of the latter is given in section 1.2.4.2.

1.2.4.1 The Owner's Perspective

The foundation of the analysis is the high-level model that represents a quantitative relation between the elements of the incentivized contracting problem. This model consists of three components: the contract model, the trade-off model and the evaluation model. Each of these models is thoroughly based on existing literature, but an integrated approach as presented by this research is still lacking. An overview of this model is presented in figure 1.3.

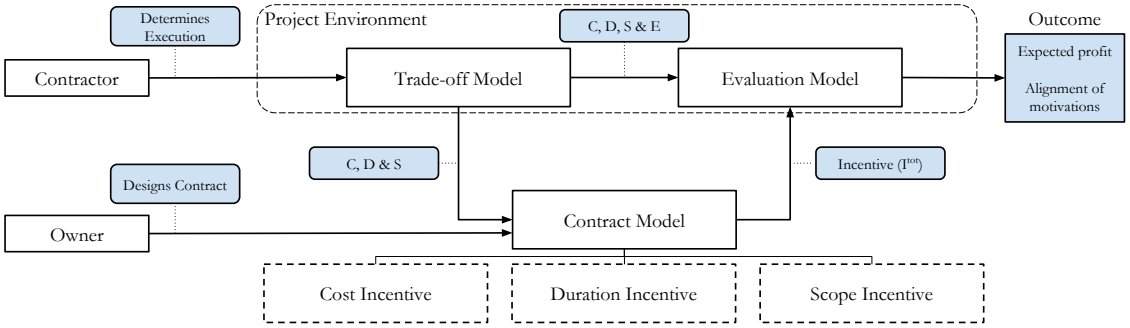


Figure 1.3: Strategic model of the contract design problem.

The nature of the project itself is captured by the trade-off model, which represents all possible ways in which a specific project can be executed. These execution modes are described as a combination of the four trade-off dimensions: cost (C), duration (D), scope (S) and effort (E) (see figure 1.1). As indicated on figure 1.3, it is the contractor who controls the manner in which this trade-off decision is made. The trade-off point chosen by the contractor is then passed on to the other model components: the contract model and the evaluation model.

The contract model uses the dimensions of the trade-off that can be perceived by the project owner (cost, duration and scope - see figure 1.3), in order to calculate the magnitude of the incentive. The contract model is designed to be capable of representing all types of incentive clauses (again for cost, duration or scope) used in practice, while

using a limited set of parameters. As shown on figure 1.3, the value of these parameters is controlled by the owner of the project.

The information on the four outcome dimensions together with the incentive calculated by the contract model forms the input for the evaluation model. The purpose of this model is to calculate the actual monetary value of the way in which the project is executed - including the awarded incentive amounts - from the perspective of both the owner and the contractor. These outcomes can then be summarized along two outcome dimensions: the expected profit and the alignment of the objectives, the latter being a measure for the risk of the project.

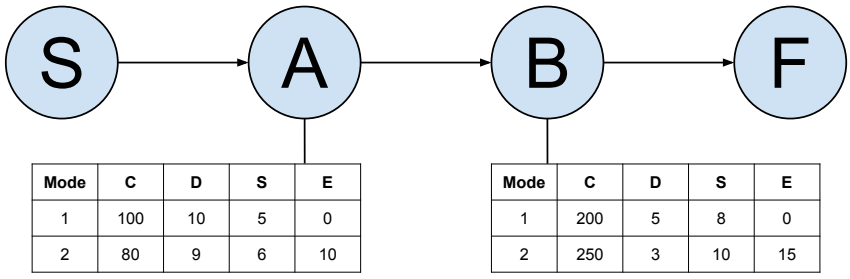
This model is the basis for chapters 3 and 4, which represent the perspective of the owner. As indicated in figure 1.3, the objective of the owner is twofold: maximization of expected profit, and the minimization of objective misalignment. The latter of these two can be viewed as a minimization of risk, since misalignment of objectives will cause a greater volatility in the project outcomes. The fact that two objectives have to be considered complicates the comparison of different solutions (i.e. contracts for a specific project). Hence, in order to optimize the contract multi-objective and multi-population meta-heuristics will be developed in order to determine the Pareto-fronts for the problems under study.

1.2.4.2 The Contractor's Perspective

The contractor's objective is the maximization of her/his profits through the optimization of both project scheduling and controlling strategies, which are the subjects of chapters 5 and 6 respectively. Section 1.2.4.2.1 introduces the methodology used to reflect the contractor's scheduling problem, and section 1.2.4.2.2 gives a brief overview of the project control problem faced by the contractor.

1.2.4.2.1 Project scheduling Creating a detailed schedule for the project execution requires analysing the project at an operational level. This implies that the analysis is made at the level of the individual activities rather than viewing the project as an aggregated whole (as was done when taking the owners perspective, see section 1.2.4.1).

Practically, this is done by defining a set of execution modes for every activity in the project, as shown in figure 1.4. Similarly to the trade-offs for the aggregated model, an activity mode is characterized by four trade-off dimensions: cost (C), duration (D), scope (S) and effort (E). Hence, the manner in which a project is executed can be fully determined by selecting an execution mode for every activity of the project. (e.g. selecting the first mode for activity A and the second mode for activity B results in a total project duration (D) of 13, with a cost (C) of 250, and a total scope (S) of 15 with 15 total effort



Mode	C	D	S	E
1	100	10	5	0
2	80	9	6	10

Mode	C	D	S	E
1	200	5	8	0
2	250	3	10	15

Figure 1.4: Principle of multi-mode project scheduling.

(E) invested by the contractor).

When interpreting the activity modes in figure 1.4, it is important to keep in mind that the cost of an activity only reflects the cost perceived by the project owner (see section 1.2.2.1). As such, the values associated with the second activity mode of activity A could be viewed as the contractor allocating a more experienced staff member to this activity. Because of this staff member's higher skill level this staff member is more expensive to the contractor since this person earns a higher wage, but this cost is carried by the contractor and not the project owner[1]. However, by allocating this person to the activity the performance in the other dimensions that are perceived by the owner will improve. For example, there may be less waste of materials in a construction project, decreasing the cost of the activity. The activity follow-up may be more thorough causing the activity to be performed faster than it otherwise would have been, or the final product may be of a higher quality level due to a higher level of attention to detail (e.g. better welds by a more experienced welder).

1.2.4.2.2 Project control A second goal of the contractor is to optimise the monitoring of an incentivised project once it is being executed. The key objective here is the creation of systems that are effective at providing control signals, while limiting the effort needed to monitor the progress of project components. These control signals enable the contractor to take adequate corrective action. The design of these techniques is based on existing techniques such as earned schedule and earned value management.

[1]This is of course only an example and there are of course projects where this cost is paid directly by the project owner. However, as stated earlier (section 1.2.2.1), contractor effort may take a number of other forms.

1.2.5 Publications

1.2.5.1 International Peer-Reviewed Publications

Kerkhove, L.-P. and Vanhoucke, M. 2015, "Incentive contract design for projects: The owner's perspective" Omega, Article in Press. (doi:10.1016/j.omega.2015.09.002)

1.2.5.2 Working Papers Currently Under Submission

Kerkhove, L.-P. and Vanhoucke, M. 2015, "A parallel multi-objective scatter search for optimizing incentive contract design in projects" Article submitted to the European Journal of Operational Research.
Kerkhove, L.-P. and Vanhoucke, M. 2015, "Multi-mode schedule optimisation for incentivised projects" Article submitted to the European Journal of Operational Research.
Kerkhove, L.-P. and Vanhoucke, M. 2015, "Earned incentive management: Controlling incentivized projects" Article submitted to the International Journal of Project Management.

1.2.5.3 Conference Presentations

Kerkhove L.-P., and Vanhoucke M. (2013, December). "The impact of multi-dimensional incentive contracts on strategy for project owners and subcontractors" Presented at EVM-Europe 2013, Ghent, Belgium.
Kerkhove L.-P., and Vanhoucke M. (2014, July). "Strategic incentive contract design for projects" Presented at IFORS 2014, Barcelona, Spain.
Kerkhove L.-P., and Vanhoucke M. (2016, April). "A quantitative approach to project incentive contracts" Presented at PMS 2016, Valencia, Spain.

1.3 Weather sensitive projects

The second part of this dissertation deals with the use of weather models within the context of operational research. Recent advances have yielded substantial improvements in both the quality of weather models, as well as the available weather data on which these models are based. In spite of this, the number of applications of these weather models in operational research has remained rather limited (Regnier, 2008). As such, the aim of the second part of this dissertation is to leverage these weather models to improve the scheduling of offshore construction projects, as well as maintenance of offshore wind turbines.

1.3.1 Research Objectives

1.3.1.1 Operationally Relevant Weather Model

The first objective of the second part of this dissertation was the creation of an adequate weather simulation model. Importantly, this weather simulation model should be able to simulate weather conditions that are relevant within an operational context. This implies that the weather model should be able to generate weather patterns using operationally relevant time intervals. Moreover, the focus has to lie on the correct weather elements. For instance, a lot of weather models focus on the precipitation, since this is often highly relevant for agriculture. However, this is not necessarily relevant from an operational point of view.

One of the key areas where operational weather models can provide substantial benefits are offshore projects. For this type of project, there are two main risk factors: the wind speed and the wave height. Naturally, these two variables are significantly (yet imperfectly) correlated. Hence, in order to improve the quality of operational decision making for this type of project, a weather model that is capable of simulating realistically correlated wave heights and wind speeds has been created.

Chapter 7 starts by giving an overview of relevant literature regarding this topic, prior to introducing a novel weather model specifically tailored for use in offshore operations. The model itself generates a realistic sequence of wind speeds and wave heights, taking into account long term seasonality. As such, the model can be used to optimise projects that last a significant amount of time (multiple months to multiple years). To do this, the model uses a combination of higher-order Markov chains as well as Weibull distributions to account for sparsely populated regions of the data. The specific model has been trained on data gathered off the Belgian shoreline, but can of course be applied to other datasets.

1.3.1.2 Optimised Scheduling for Offshore Construction

The objective of chapter 8 is to improve the planning of the construction of offshore wind turbines. This is done by employing the new weather model proposed in chapter 7 to make better predictions regarding the outcome of different scheduling strategies. The scheduling of projects is optimised by designing a number of novel chromosome encodings that represent different levels of abstraction in the scheduling of offshore construction projects. These encodings are then optimised using meta-heuristic solution techniques.

1.3.1.3 Weather-Based Maintenance Strategies

Maintenance is a frequently overlooked aspect of the total lifetime cost for wind turbines. In spite of this, the operating and maintenance cost represent approximately 40% of the total cost associated with operating a wind farm. Recently, this topic has received more attention, but literature has mainly focussed on high-level capacity decisions and corrective maintenance (i.e. repairing breakdowns.) Chapter 9 extends the literature by investigating the operational go/no-go decision for preventive rather than corrective maintenance. The key consideration here being that preventive maintenance is best executed at times when wind speeds, and therefore energy production, are low. Again leveraging the capabilities of the weather simulation model presented in section 7, several strategies for making this decision are compared in a computational experiment. Specifically the performance of a static threshold, a workload dependent threshold, support vector machines and an artificial neural networks is compared.

1.3.2 Methodology

Several quantitative techniques are employed to perform the analysis in the second part of this dissertation. The most important of these are briefly summarised in this section.

1.3.2.1 Markov Chains

The weather model presented in chapter 7 uses second-order Markov chains as one of its major components. A Markov chain can be used to represent a memoryless transition probability between specific states. This implies that the likelihood of moving to a specific state only depends on the current state. Higher-order Markov chains simply include slightly more history, i.e. the transition probabilities of a n-th order Markov chain depend on the preceding n states. The second order Markovian model presented in this research is somewhat different in that it depends on the preceding wave state as well as the simulated wind state in order to simulate realistically correlated wave states.

1.3.2.2 Discrete Event Based Simulation

The studies in chapters 8 and 9 both use discrete event based simulation to determine the implications of different scheduling and maintenance strategies. This type of simulation uses an event list to determine the next significant occurrence in the system under study. This type of simulation contrasts with continuous simulation which continually tracks the status of the studied system, at higher computational expense.

1.3.2.3 Solution Heuristics

Because simulation is required to determine the outcome of various strategies, heuristic rather than exact solution strategies are employed in this research. Specifically local-search based meta-heuristics are preferred (see section 1.3.2.4). This includes both meta-heuristics such as simulated annealing, as well as dedicated heuristics which exploit the specific nature of the studied problem to find interesting regions in the solution space.

1.3.2.4 Common Random Numbers

Local search heuristics generally compare the performance of two solutions, the 'current best' and the 'candidate' solution. However, when these two solutions are evaluated by means of simulation, it is important that the differences between these two candidates are statistically significant and not simply due to random chance. Hence, the common random numbers technique was used to reduce the variance of the difference between these techniques, effectively reducing the computation time required to compare two solutions. Doing so effectively increases the total number of solutions that can be considered, improving the coverage of the solution space by the heuristic.

1.3.2.5 Support Vector Machines

In recent years support vector machines have become one of the key staples in machine learning. The main objective of these support vector machines is to find a hyperplane that is capable of separating observations in their respective classes. Because of their nature as categorical classifiers they are highly suited to taking the operational go/no-go decisions investigated in chapter 9.

1.3.2.6 Artificial Neural Networks

Like support vector machines, artificial neural networks are a staple in the domain of machine learning. Similarly to many techniques in the domain of operational research, this technique is inspired by biology - specifically by the central nervous system. Practically, the

central nervous system is modelled as an interconnected set of 'neurons' that are activated when they are stimulated through these connections. The nature of these connections is governed by 'weights' that are trained by exposing the network to specific output and input data in the training data set. Again, this technique can be used to make operational go/no-go decisions, similarly to the support vector machine technique.

1.3.3 Publications

1.3.3.1 International Peer-Reviewed Publications

Kerkhove, L.-P. and Vanhoucke, M. 2016, "Optimised scheduling for weather sensitive offshore construction projects" Omega, Article in Press. (doi:10.1016/j.omega.2016.01.011)

1.3.3.2 Working Papers

Kerkhove, L.-P. and Vanhoucke, M. 2016, "Weather-Based Maintenance Strategies for Offshore Wind Farms" Working paper.

Part I

A Quantitative Analysis of Incentive Contracts for Projects

2

A Review of Literature on Incentives for Projects

This chapter presents an extensive review of relevant literature for dealing with incentives in a project context. This includes the literature on the nature of the incentive contracts, as well as literature describing the way in which the execution of a project can be modelled. Specifically, the relationship between project cost, scope, duration and effort invested by the contractor.

2.1 Introduction

Literature relevant to incentive contract design for projects can be divided into two main categories: literature dealing with the trade-offs in project management, and literature concerned with the design and implications of incentive contracts.

Project management literature dictates that the properties of a project can be described along three dimensions: the costs associated with the project, the duration of the project and the scope of a project (also known as the *iron triangle*, Marques et al. (2011)). These three dimensions are viewed as an interrelated trade-off mechanism. Ceteris paribus, decreasing the cost of a project will be accompanied by an increase in duration and/or a decrease in scope. Similar statements are also true for the duration and scope dimensions.

Within the context of this chapter, these three dimensions are viewed as the outcomes of the project, as perceived by the project owner. The cost reflects the financial payment the owner has to make to the contractor to compensate for the work performed by the latter, as well as the resources used in the project (insofar as this amount is variable). The duration represents the time needed by the contractor to complete the project. The scope of the project is a very general term which encompasses the amount of work performed as well as the quality level of the performance.

When a project is expedited to a contractor, it is the latter who controls the precise way in which the project is carried out. Depending on the way in which the contractor decides to execute the project, (s)he effectively selects one of the possible trade-off points on the cost-duration-scope spectrum.

Due to the fact that the contractor is introduced as a second party, a fourth dimension has to be introduced to extend the traditional cost-duration-scope trade-off: *contractor effort*. The reason for this is that not all actions of the contractor can be directly controlled or perceived by the owner. This concept has already appeared widely in the literature (see section 2.2), albeit sometimes under other aliases. Contractor effort can be defined as an investment in the project by the contractor, which enhances the outcome of one or more of the three traditional dimensions, but is not directly perceived by the owner. Effectively, an increased effort from the contractor increases the total utility which can be derived from the project by the owner. Naturally, an increase in contractor effort comes at a cost to the contractor, which (s)he will wish to compensate by earning a larger incentive fee (see section 2.2 for more information on contractor effort).

The second body of literature studies the design and implementation of incentive contracts, as well as their (perceived) effectiveness. A project owner can have several reasons for including incentive clauses, such as encouraging more innovative solutions (Chan et al., 2010b), efficiently allocating risk, (Chapman and Ward, 2008), improving communication

between both parties (Müller and Turner, 2005), or creating a more cooperative mindset (Rose and Manley, 2011). This research focuses on the two most prominently used and quantifiable measures: the owner's expected profit, and the degree to which the motives of the owner and contractor are aligned. The owner's profit is viewed as an objective measure for the performance of the project (Bower et al., 2002; Chan et al., 2010b; Rose and Manley, 2011; Van Weele and van der Puil, 2013), assuming that all relevant elements have been quantified. A second quantifiable motivation for the implementation of contracts is the degree to which the contract succeeds in transferring risks and motivations to the contractor (Bower et al., 2002). The primary motivation for this is the removal of the risk associated with agency-conflicts (Gibbons, 2005; Sacks and Harel, 2006). Moreover, the degree to which this is done also heavily influences other beneficial effects such as the cooperative atmosphere between the two parties (Bower et al., 2002; Chan et al., 2010b). Hence, the implementation of an incentive agreement can be viewed as being related to economic portfolio theory, maximising the return (i.e. owner profit) for a given level of risk (i.e. objective alignment) (Chapman and Ward, 2008; Markowitz, 1952).

Practically, the owner's (expected) profit can be determined by assuming that the contractor is a profit maximising economic actor (Van Weele and van der Puil, 2013). Under this assumption, the manner in which the project is executed will guarantee maximal attainable profits for the contractor. Hence, the expected profit for the owner is equal to the profit associated with this scenario. Measuring the alignment of motivations is done by comparing the relative outcomes for both parties across the possible project outcomes.

The most relevant works in both domains are listed in table 2.1, where the *"Dimensions"* columns indicate whether the paper deals with incentive contracts (I) for a specific dimension, or the trade-off problem (T) related to a specific dimension.

The trade-off between the cost, duration, scope, and effort levels which is controlled by the contractor should not be confused with the balancing of the different dimensions of the incentive contract, which are linked to the former three dimensions of the contractor's trade-off. The difference between these concepts is shown by figure 3.1, which indicates that the trade-off is a representation of the nature of the project as perceived and controlled by the contractor. The same illustration also shows that the contract uses the outcomes of the project (i.e. the trade-offs) as an input, but the structure of the contract itself is a separate decision controlled by the owner rather than the contractor. To prevent any confusion regarding these concepts, the remainder of this manuscript will only use *'trade-off'* when referring to the trade-off decision of the contractor.

2.2 Trade-off Literature

The most traditional trade-off problem in project management is the one between the cost and duration of a project. The earliest authors modelled this problem using a simple linear relationship (Kelley and Walker, 1959), but recent authors generally agree that the most realistic model for this trade-off is a convex curve (Choi and Kwak, 2012; Shr and Chen, 2003). This trade-off can be extended by considering the scope of the project, thus forming the well-known concept of the iron triangle (Marques et al., 2011). Again the nature of the trade-offs between scope and cost or time are generally assumed to be of convex nature (El-Rayes and Kandil, 2005).

The scope concept is used in a broad sense, encompassing not only the amount of work performed, but also the quality of this work and any other area of performance which is valued by the owner (Bower et al., 2002; Rose and Manley, 2011). Hence, it can also be considered to encompass concepts such as occupational health and safety, environmental impact and coordination which have received increasing amounts of attention in recent years (Gangwar and Goodrum, 2005; Meng and Gallagher, 2012; Tang et al., 2008). Nevertheless since it is possible to incentivise these components separately, a disaggregation of the scope concept could be interesting for future research.

As stated above, modelling the owner-contractor relationship requires a fourth dimension to be added to the trade-off: contractor effort. Though the nomenclature differs between authors, several examples of this dimension can be found in the literature, as shown by the *"E"-Dimension* column in table 2.1.

Table 2.1: Overview of literature on incentive contracting and associated trade-offs.

Author(s)	Dimensions[1] C	D	S	E	Contract Nature[2]	Cost Contract Types[3]	$A+B$[4]	Validation[5]
Meinhart and Delionback (1968)	I	I	I	-	L	FPI/TCC, CPIF	-	-
McCall (1970)	I	-	-	-	L	FFP, FPI/TCC, CPFF	-	-
Cukierman and Shiffer (1976)	-	I	-	-	L	-	-	-
Hiller and Tollison (1978)	I	-	-	-	L	FPI/TCC, CPFF, CPPF	-	CS
Weitzman (1980)	I	-	-	-	L	FFP, CPIF, CPFF	-	TE
Stukhart (1984)	I	I	I	-	L	FFP, GMP, FPI/TCC, CPFF, CPPF	-	-
Herten and Peeters (1986)	I	I	I	-	P/L	FFP, FPI/TCC, CPIF, CPFF, CPPF	-	CS, Sur
McAfee and McMillan (1986)	I	-	-	-	L	FFP, CPIF, CPFF, CPPF	-	CS
Ryan et al. (1986)	I	-	-	-	L	FFP, FPI/TCC, CPFF	-	-
William and Ashley (1987)	I	I	I	-	-	FFP, CPFF, CPPF	-	Sur
Abu-Hijleh and Ibbs (1989)	I	I	I	T	P/L	FFP, GMP, CPIF, CPFF, CPPF	-	CS
Veld and Peeters (1989)	I	I	I	-	P/L	FFP, FPI/TCC, CPIF, CPFF, CPPF	-	Sur
Rosenfeld and Geltner (1991)	I	-	-	-	L	FFP, GMP, FPI/TCC, CPIF, CPFF, CPPF	-	TE
Chapman and Ward (1994)	I	-	-	T	L	FFP, CPFF, CPPF	-	-
Herbsman (1995)	-	I	-	-	L	-	x	Sur
Jaraiedi et al. (1995)	I	I	I	-	L	FPI/TCC, CPIF	x	-
Ward and Chapman (1995)	I	-	-	-	P/L	FFP, FPI/TCC, CPIF, CPFF	-	-
Jaafari (1996)	I	I	T	-	P/L	FFP, FPI/TCC	-	CS, Sim
Al-Subhi Al-Harbi (1998)	I	-	-	-	L	FFP, FPI/TCC, CPIF, CPFF, CPPF	-	-
Arditi and Yasamis (1998)	I	I	T	T	L	FFP, FPI/TCC, CPIF, CPFF	x	Sur
Khang and Myint (1999)	T	T	T	-	-	-	-	CS
Berends (2000)	I	-	-	-	P/L	FFP, FPI/TCC, CPIF, CPFF	-	CS
Paquin et al. (2000)	-	-	T	-	-	-	-	TE
Perry and Barnes (2000)	I	-	-	-	L	FPI/TCC	-	TE
Boukendour and Bah (2001)	I	-	-	-	P/L	GMP	-	-
Dayanand and Padman (2001)	-	I	-	-	-	-	-	TE
El-Rayes (2001a)	-	I	-	T	L	-	x	TE
Bower et al. (2002)	I	I	I	-	P/L	FFP, CPIF	-	CS
Broome and Perry (2002)	I	T	T	-	P/L	FPI/TCC, CPIF, CPFF	-	CS
Bubshait (2003)	I	I	I	-	L	FPI/TCC, CPIF, CPFF, CPPF	-	Sur
Shr and Chen (2003)	T	I	-	-	L, N	-	x	CS
Shr and Chen (2004)	-	I	-	-	L	-	x	CS
Shr et al. (2004)	T	I	-	-	L	-	x	CS
Turner (2004)	I	-	-	-	L	FFP, GMP, FPI/TCC, CPIF, CPFF, CPPF	-	-
Bayiz and Corbett (2005)	-	I	-	T	L	FFP	-	TE
El-Rayes and Kandil (2005)	T	T	T	T	-	-	x	TE
Gangwar and Goodrum (2005)	-	-	T	-	-	-	-	Sur
Kandil and El-Rayes (2006)	I	I	-	T	-	-	-	-
Pollack-johnson and Liberatore (2006)	T	T	T	-	-	-	-	TE
Tang et al. (2006)	I	I	I	-	-	-	-	Sur
Tareghian and Taheri (2006)	T	T	T	-	-	-	-	TE
Osei-Bryson and Ngwenyama (2006)	I	-	T	-	L	FFP, CPIF	-	-
Afshar et al. (2007)	T	T	T	-	-	-	-	TE
Lee and Thomas (2007)	-	I	T	T	L	-	x	CS
Rosandich (2007)	I	I	I	-	L	FPI/TCC	-	TE, Sim
Sillars (2007)	T	I	-	T	L	-	x	CS
Chapman and Ward (2008)	T	T	-	-	L	FPI/TCC, CPFF	-	CS
Rose (2008)	I	I	I	-	P/L	FFP, CPIF	-	CS
Stenbeck (2008)	T	-	I	-	L	-	-	CS
Tang et al. (2008)	I	I	I	-	L	-	-	CS, Sur
Iranmanesh et al. (2008)	T	T	T	-	-	-	-	-
Rahimi and Iranmanesh (2008)	T	T	T	-	-	-	-	-
Ghodsi et al. (2009)	T	T	T	-	-	-	-	TE
Ramón and Cristóbal (2009)	T	T	T	-	-	-	-	CS
Bajari and Lewis (2009)	T	I	-	-	L	-	x	Sur
Anvuur and Kumaraswamy (2010)	I	I	I	-	L	GMP	-	CS
Chan et al. (2010b)	I	-	-	-	P/L	FFP, FPI/TCC	-	CS, Sur
Chan et al. (2010a)	I	-	-	-	L	GMP, FPI/TCC	-	Sur
Choi et al. (2010)	T	I	-	-	L	-	-	CS
Love et al. (2010)	I	I	I	-	P/L	-	-	Sur
Mihm (2010)	I	-	-	-	L	FPI/TCC	-	-
Rose and Manley (2010)	I	I	I	-	L	CPIF	-	CS
Shahsavari Pour et al. (2010)	T	T	T	-	-	-	-	TE
Zhang and Xing (2010)	T	T	T	-	-	-	-	CS
Pour et al. (2010)	T	T	T	-	-	-	-	-
Chan et al. (2011a)	I	T	T	-	L	FFP, GMP, FPI/TCC	-	Sur
Chan et al. (2011b)	I	-	-	-	L	GMP, FPI/TCC	-	Sur
Rose and Manley (2011)	I	I	I	-	L	-	-	CS
Chan et al. (2012)	I	-	-	-	L	FFP, GMP, FPI/TCC	-	Sur
Choi and Kwak (2012)	T	I	-	T	L	-	-	CS
Mackley (2012)	T	I	-	-	P/L	-	-	CS
Meng and Gallagher (2012)	I	I	I	-	L	FFP, FPI/TCC, CPIF, CPFF	-	CS, Sur
Keren and Cohen (2012)	T	T	T	-	-	-	-	TE
Zhang et al. (2012)	T	T	T	-	-	-	-	TE
Lippman et al. (2013)	I	-	-	T	L	FFP, CPIF, CPFF, CPPF	-	-
Tavana et al. (2013)	T	T	T	-	-	-	-	-
Cohen and Iluz (2014)	T	T	T	-	-	-	-	Sur
Hu and He (2014)	T	T	T	-	-	-	-	CS
Chen et al. (2015)	T	I	-	-	P	-	-	-
Tang et al. (2015a)	T	I	-	-	L	-	x	-
Gupta et al. (2015)	T	I	-	-	P/L	-	x	-

[1] Indicates if the authors cover incentive contracting (I) or the trade-offs (T) associated with the cost (C), duration (D), scope (S) and effort (E) dimensions.
[2] Incentive contracts can be linear (L), piecewise linear (P) or non-linear (N).
[3] Indicates which of the traditional cost contract types are treated, these include: firm fixed price (FFP), guaranteed maximum price (GMP), fixed price incentive/target cost contract (TCC), cost plus incentive fee ($CPIF$) and cost plus fixed fee ($CPFF$) contracts.
[4] Shows if the publication deals with $A+B$ contracting, a frequently used bidding technique which is often used in combination with time incentives.
[5] The type of empiric or experimental validation used in the paper, a distinction is made between the following categories: case study (CS), theoretical example (TE), survey (Sur) and simulation (Sim).

The simplest example of contractor effort is managerial attention from the contractor (Abu-Hijleh and Ibbs, 1989; Arditi and Yasamis, 1998), which can improve project performance, at a cost to the contractor. Another example is the proactive reduction of uncertainty through the development of contingency plans and by taking out insurance policies (Chapman and Ward, 1994). For contracts where the personnel costs are not included in the project cost dimension, but assumed to be included in the basic fee of the contractor, the allocation of additional staff can also be considered to be contractor effort (El-Rayes, 2001a; El-Rayes and Kandil, 2005). Another very specific example is presented by Lee and Thomas (2007), who considered a road refurbishment project where the contractor hired an on-site towing service to prevent delays due to uncleared accidents. Bayiz and Corbett (2005) also use a similar dimension which they define as an unobservable element to the project owner, which comes at a certain cost to the contractor. Moreover, they also assume the trade-off relationship to be convex, something which also returns in the axioms used to construct the trade-off model in section 3.4.1.

2.3 Incentives Literature

Incentive contracts for projects can be categorised along two dimensions. The first dimension being the trade-offs which are covered by the specific contract. An incentive contract can be linked to any of the three trade-off dimensions which are perceived by the owner: cost (C), duration (D) and scope (S). Naturally, an incentive contract can include multiple dimensions. Moreover, an important decision in incentive contract design can be finding the right balance between these different dimensions (Abu-Hijleh and Ibbs, 1989).

The second dimension is the nature of the equations used to calculate the incentive amount earned, based on the value of the relevant outcome dimension. These equations can be linear (L), piecewise linear (P), or non-linear (N). The *Dimensions* and *Contract Nature* columns of table 2.1 show how the existing literature on incentive contracts can be categorised along these two dimensions.

2.3.1 Literature on Cost Incentive Contracts.

Cost incentives are the most researched incentive category, as can easily be seen from table 2.1. The most basic form of these cost incentives is a simple linear equation. The properties of such a contract can be described using two simple equations, representing the total amount paid by the project owner (P) and the yield for the subcontractor (Y). The latter does not include the overheads of the subcontractor, but solely represents the cash inflow (s)he receives from the project. These equations can be expressed as follows (based on the notation used by Perry and Barnes (2000)):

$$P = C + F + s \cdot (C^t - C) \tag{2.1}$$

$$Y = F + s \cdot (C^t - C) \tag{2.2}$$

In equations 2.1 and 2.2, C represents the actual direct cost incurred (excluding the fee for the contractor), and C^t represents the a priori cost target set by the project owner. The contractor's fee is represented by F. Note that in this case the fee of the contractor is assumed to be fixed (independent from the actual cost incurred), for some contracts this fee is expressed as a fraction of the actual costs incurred. The contractor's share of the cost savings or overrun is expressed by $s \in [0, 1]$.

As shown in table 2.1, cost contracts can frequently be categorised as one of six basic contract types (Weitzman, 1980): Firm Fixed Price (FFP), Guaranteed Maximum Price (GMP), Fixed Price Incentive (FPI) (or Target Cost (TCC)), Cost Plus Incentive Fee ($CPIF$), Cost Plus Fixed Fee ($CPFF$) and Cost Plus Percentage Fee ($CPPF$) contracts. This set of cost contract types forms a risk-transfer spectrum, where the FFP contract represents the situation where all the risk is carried by the contractor, and the $CPPF$ is the situation where all the risk is carried by the owner (Al-Subhi Al-Harbi, 1998). Variations on these archetypes are also often used to manage cost performance in production environments (Wang et al., 2015). These contract types all use a single sharing ratio, sometimes extended with a safeguard preventing excessive disincentives to be allocated to contractors (Turner, 2004). The basic properties of these contract types can be summarised as follows:

Firm Fixed Price (FFP): The contractor and the client agree on a fixed price for the contract. Regardless of the actual costs incurred during the project this is the amount which will be paid. This contract effectively transfers all the risk from the client to the contractor, the latter will of course charge a risk premium for this service. The equations expressing the owner's payment and contractor's yield can be written as:

$$P = F \tag{2.3}$$

$$Y = F - C \tag{2.4}$$

Often the contractor will be reluctant to agree to this type of contract unless his financial situation is stable enough to cope with potential losses and he has a good control over the costs. This contract also reduces the project owner's willingness to invest ongoing effort to reduce the cost, since this no longer affects his profits.

Guaranteed Maximum Price (GMP): This contract sets an upper limit on the payment of the client to the contractor. If this cost is exceeded the contractor bears all the remaining costs. For outcomes where the final costs are below the threshold a sharing ratio is usually proposed to encourage cost saving by the subcontractor. Note that in case this sharing ratio would not be present there would be no incentive for the subcontractor to spend less than the agreed upon maximum amount. Notably, if the fee paid to the contractor is a percentage of the total incurred costs - as is regularly the case - there is actually an incentive for the subcontractor to match these costs as closely as possible to the guaranteed upon maximum. The equations expressing the owner's payment and contractor's yield can be written as (where MP represents the maximum price agreed upon in the contract):

$$P = \min(MP, C + F + s(C^t - C)) \tag{2.5}$$

$$Y = \min(MP, C + F + s(C^t - C)) - C \tag{2.6}$$

These equations make it clear that the FFP is nothing more than the standard incentive contracting equations 2.1 and 2.2, with added guarantees for the project owner.

Effectively this contract improves upon the FFP contract because the client is now also motivated to cooperate with the contractor to decrease the costs. Nevertheless the contractor still bears the downside risk in this contract. Moreover, the upside potential of the contract is reduced in this contract, since the subcontractor has to share a fraction of the cost savings with the project owner.

Another important remark when comparing this contract to the firm fixed price is that the administrative effort is greatly increased. Whereas the FFP contract did not need any open accounting principles or measurement, objective cost measurement will be important in this type of contract. This is a downside which is encountered in the majority of incentive contract structures.

Fixed Price Incentive (FPI) / Target Cost Contract (TCC): The client and the contractor agree upon a certain target cost as well as a sharing percentage for the cost over- or under-runs. The payment for the contractor then consists of the redeemable costs plus his basic fee plus the incentive or disincentive amount. The equations are simply the basic equations which were presented at the start of this section.

$$P = C + F + s \cdot (C^t - C) \tag{2.7}$$

$$Y = F + s \cdot (C^t - C) \tag{2.8}$$

This type of contract is often used when the client and contractor have cooperated on the cost estimate of the project and potential deviations are not the fault of either party. This type of contract provides both parties with a substantial incentive to keep the costs down.

Cost Plus Incentive Fee (CPIF): This contract sets a target cost and a sharing ratio of the cost savings when the costs incurred are below their target cost. No disincentive is included when the costs exceed this amount and the client carries the risk of potential cost overruns. This contract is the mirror image of the GMP where the contractor carried the risk of costs overruns. The equations expressing the payment and yield are now the following:

$$P = \max(C + F + s \cdot (C^t - C), C + F) \tag{2.9}$$

$$Y = \max(C + F + s \cdot (C^t - C), C + F) - C \tag{2.10}$$

When the contractor has a relatively weak financial position compared to the magnitude of the project this type of contract is often used to avoid bankruptcy risk for the contractor. The contractor will of course have to lower his fees to accommodate for the client bearing the majority of the cost risk. Variations on this contract type can also include a small disincentive never exceeding a minimal profit margin for the subcontractor, such as the scheme proposed by Love et al. (2010).

Cost Plus Fixed Fee (CPFF): This contract does not include any incentive structure, the project owner pays the direct costs associated with the contract, plus a fixed fee. Effectively even more risk is transferred to the owner this way, no longer including any incentive for the contractor to limit the costs. The equations expressing the payment and yield are now the following:

$$P = C + F \tag{2.11}$$

$$Y = F \tag{2.12}$$

An advantage of this type of contract is that the fee of the contractor will be substantially lower since there is no longer a need to include a risk premium.

Cost Plus Percentage Fee (CPPF): This contract is identical to the previous except that the fee is no longer fixed but a percentage of the actual cost. The equations representing the pay-offs now become as follows, with f representing the fraction of the actual costs being paid as a fee to the subcontractor.

$$P = C + f \cdot C \tag{2.13}$$

$$Y = f \cdot C \tag{2.14}$$

When contracts of this form are used it is very important for the project owner to retain substantial control during the execution of the project since the subcontractor has an implicit incentive to spend as much as possible. Hence, the rationality of employing this contract form can rightfully be called in to question, nevertheless literature shows that this type of contract is still in common use (see table 2.1).

Piecewise linear equations are often used in practice in order to improve upon the incentive schemes presented above (Abu-Hijleh and Ibbs, 1989; Berends, 2000; Boukendour and Bah, 2001; Bower et al., 2002; Broome and Perry, 2002; Herten and Peeters, 1986; Jaafari, 1996; Love et al., 2010; McCall, 1970; Rose, 2008; Veld and Peeters, 1989; Ward and Chapman, 1995). The main goal of these incentive schemes is to provide a better match to the specific risks of a project. Several practical examples of such incentive schemes are given by Broome and Perry (2002). The basic principle of such contracts is the use of multiple sharing ratios for different ranges of cost outcomes. By doing so, a better alignment of motivations is obtained, while also giving the possibility to cap the upside or downside risk for either party. The basic principle of such a piecewise linear contract is illustrated by equation 2.15.

$$I^C(C) = \begin{cases} s_1^C \cdot (C^t - B_2^C) + s_2^C \cdot (B_2^C - C) & \text{if } C > B_2^C \\ s_1^C \cdot (C^t - C) & \text{if } B_1^C \leq C \leq B_2^C \\ s_1^C \cdot (C^t - B_1^C) + s_0^C \cdot (B_1^C - C) & \text{if } C < B_1^C \end{cases} \tag{2.15}$$

In this equation I^C is the cost incentive amount paid to the contractor which is a

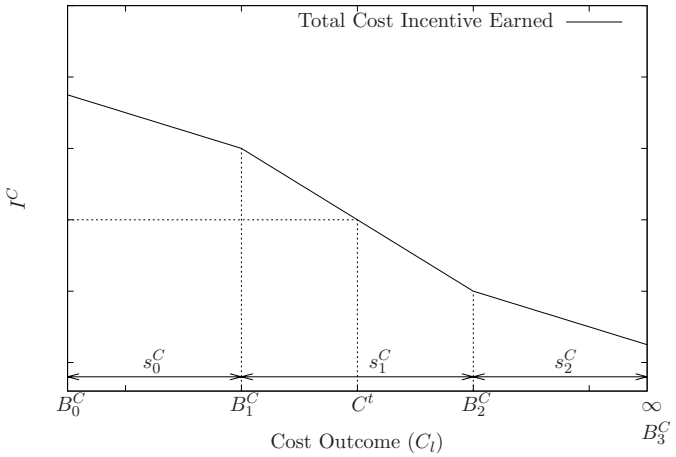


Figure 2.1: Visualisation of a piecewise linear incentive contract.

function of the final cost of the project C, C^t is the target cost specified in the contract, B_r^C is the lower bound of the r-th cost region specified in the contract, and s_r^C is the sharing ratio for the r-th region.

Figure 2.1 visualises this piecewise linear incentive contract. For this specific example, the contractor carries a lower fraction of extreme upside and downside risk when the costs are significantly higher or lower than expected. Such a contract can be used to take into account that the contractor may not be held accountable if the original estimate of the cost turned out to be unrealistic.

2.3.2 Literature on Duration Incentive Contracts

Since no open accounting standards are required to agree upon the outcomes of the time dimension, the duration incentive is arguably the easiest incentive to implement (Abu-Hijleh and Ibbs, 1989). A very fruitful area of research related to time incentive contracting is the research done on the $A + B$ contracting method employed by the US government in highway contracting (see column "$A + B$" in table 2.1). This is a bidding method where the contractors have to submit a bid for both the expected cost and the expected duration of a project, where after the road user cost (i.e. the economic damage of closing the road for a day) is used to evaluate the total cost of all offers. The implementation of such contracts is then often accompanied by a per diem time incentive. Moreover, this type of bidding allows the owner to get a better perspective on the nature of the trade-offs from the perspective of the contractor (Choi et al., 2010), thus aiding the owner in estimating the possible reaction of a contractor to a given incentive contract more accurately.

Piecewise linear contracts for the duration dimension are discussed less frequently than piecewise linear contracts for the cost trade-off dimensions. Nevertheless piecewise linear contracts for time are used (Love et al., 2010) and even non-linear contracts using a quadratic formula have been discussed in the literature (Shr and Chen, 2003).

2.3.3 Literature on Scope Incentive Contracts

Scope incentives are the least commonly used of all incentive contracts, however in cases where they are used, they are often considered the most influential incentive dimension of the contract (Tang et al., 2008). Table 2.1 shows that scope incentives are rarely used when cost and time incentives are not already in place. This is most likely due to the added complexity of the valuation of the performance along this dimension. Nevertheless, several authors have proposed methods to facilitate the evaluation of scope performance. Most of these methods use key performance indicators (Rose, 2008) or balanced scorecard techniques (Tang et al., 2008). Along the same lines, Pollack-johnson and Liberatore (2006) use the analytic hierarchy method to link these techniques to the work breakdown structure of a typical project environment.

Although they are the least commonly used incentive category, the adoption of scope incentives is not novel, Herten and Peeters (1986) discuss an interesting example of a scope-based incentive contract from 1908 when the Wright brothers were awarded an incentive based on the airspeed attained by the plane.

2.3.4 Multi-dimensional contracts

One of the key challenges of the project owner is the integration of incentive clauses related to the cost, duration and scope dimensions into a single multi-dimensional contract. Creating a correct balance between the various incentive dimensions is paramount to the quality of the contract. This balance it attained only when the relative strength of the incentives succeed in aligning the motivations of the contractor with those of the project owner (Bower et al., 2002). The use of an unbalanced contract can result in unfavourable gaming by the contractor, who sacrifices the owner's return on investment in favour of his own. One such example is discussed by Stukhart (1984), where a contractor who was unable to meet the cost target decided to over-invest in speeding up the project in order to earn a substantial time incentive. In spite of this risk, the use of contracts containing multiple incentive clauses has increased in recent years (Meng and Gallagher, 2012).

Literature on these multi-dimensional incentive contracts is mainly limited to specific cases, as well as surveys investigating the use and perceived effectiveness of these contract. Hence, one of the key research opportunities is the design of more prescriptive models that

can aid the project owner in his contract design process.

2.4 Conclusion

The literature review on incentive contracts for projects presented in this chapter clearly shows that there is substantial academic and professional interest in this subject. A key research opportunity in this respect is the design of more quantitative and prescriptive models which can guide project owners and contractors in their interactions with such incentive contracts. Existing literature on trade-offs in projects offers interesting opportunities for cross-fertilisation, enabling the elements underlying the contract structures to be quantified.

3

The Owner's Perspective on Incentive Contract Design

The aim of this chapter is to provide a quantitative framework for incentive contract design in projects, which can be used by the project owner to select the most adequate contract for any given project environment. This quantitative framework consists of three components: a trade-off model describing the nature of the project, an evaluation model describing the valuation of the different outcomes of the project for both the owner and contractor, and a contract model which is capable of representing the majority of (incentivised) contractual agreements used in practice.

3.1 Introduction

The project owner and the contractor executing the project are two separated economic actors, each with their own set of potentially conflicting objectives (Müller and Turner, 2005; Turner, 2004). Hence, when the owner expedites work to a contractor, a relationship must be established. The nature of this relationship can be plotted on a spectrum between an explicitly negotiated contract and an alliance in which both parties are formally unified into a single economic actor for the duration of the project. For projects where complexity is limited, an explicit contract which specifies the deliverables can suffice (Van Weele and van der Puil, 2013). For complex projects on the other hand, it may be more favourable to unify both actors in an alliance structure, effectively forming a single economic entity (Walker et al., 2002). Although valuable arguments can be made in favour of such alliance structures (Rose and Manley, 2011), the implementation of such a structure is often highly complex (Bresnen and Marshall, 2000) and expensive (MacCormack and Mishra, 2015). Hence, the inclusion of incentive clauses, which form the middle ground in the relational spectrum between explicit contracts and alliances, can provide a more workable alternative (Bower et al., 2002). Performance related pay in general (Dayanand and Padman, 2001; Ederer and Manso, 2013; Zhang et al., 2015), and the design of incentivised agreements for projects in particular (see table 2.1) have been widely studied in academic literature over the last decades. Notwithstanding these recent advances, little guidance is available for project owners on how to identify the best contract for a specific project environment.

The aim of this chapter is to provide a quantitative framework for incentive contract design in projects, which can be used by the project owner to select the most adequate contract for any given project environment. This quantitative framework consists of three components: a trade-off model describing the nature of the project, an evaluation model describing the valuation of the different outcomes of the project for both the owner and contractor, and a contract model which is capable of representing the majority of (incentivised) contractual agreements used in practice.

Using these models, computational experiments have been carried out to investigate the impact of different project environments on the performance of contract types. These experiments take an economical rather than psychological perspective on the problem, and therefore assume that the contractor is a risk neutral profit-maximising actor. This risk-neutrality can be assumed since economic actors rather than individuals are considered (Gibbons, 2005). The desirability of different types of contracts is judged by taking into account both the expected profit of the owner, as well as the degree to which the motivations of the owner and contractor are aligned.

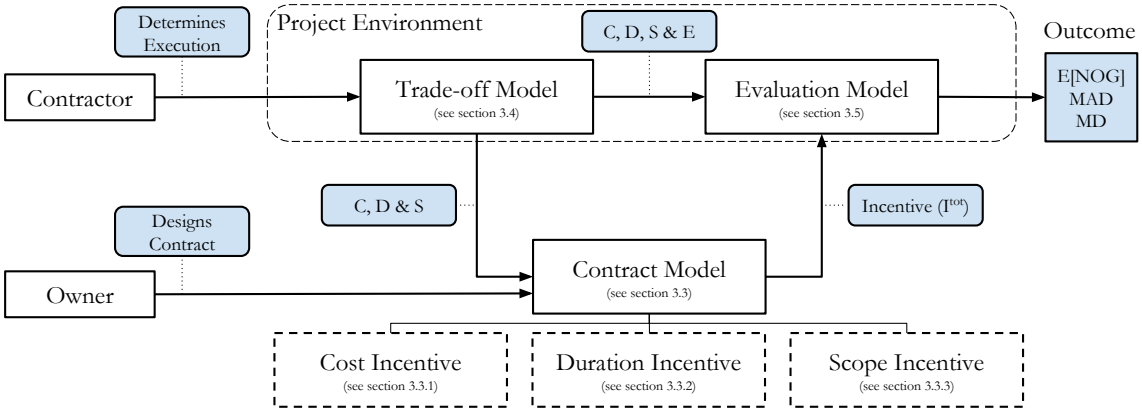


Figure 3.1: Visualisation of the modelling approach, and the interrelations between models.

3.2 Modelling the Problem

The contract design problem faced by the contractor is quantified using a combination of three models: the contract model, trade-off model, and the evaluation model (see figure 3.1). The underlying assumptions for each of these models are based on existing literature.

The *contract model* represents the structure of the incentive contract and is based on contracting literature. This model was designed to be capable of representing the majority of incentive contracts used in practice, thus allowing for a comprehensive search of the solution space.

The *trade-off model* model describes the nature of the project on which the incentive contract is to be used. The foundations for this second model originate from project management literature, from which six basic axioms have been derived. These six axioms fully describe the relationship between the four discernible trade-off dimensions of the contractor: direct cost, duration, scope and contractor effort. Because the contractor is assumed to be a risk-neutral profit-maximiser it suffices to define the trade-offs deterministically. Nevertheless, relaxing this assumption and working with a stochastic definition of the project is an interesting topic for future research[1].

The goal of the *evaluation model* is simply the valuation of a project and contract combination. The model itself uses financial valuations, rather than utility theory, since the latter is often much harder to apply in practical situations (Broome and Perry, 2002). This model quantifies the evaluations of both parties, as well as their alignment in a way which allows for an objective evaluation of the adequacy of a contract.

In any project these three building blocks can be defined: there is always a contract,

[1] An interesting approach to stochasticity and incentives has been presented by Sommer and Loch (2009). Future research could potentially combine this approach with the models presented in this chapter.

there are always different ways of executing a project[2], and both the contractor and owner always associate certain costs and profits with different outcomes. For this paper the models have been designed in order to cover as much practical cases as possible, but naturally it may be necessary to extend these models for highly specific cases.

The details of these three models are further explained in sections 3.3, 3.4 and 3.5 respectively. For an overview of the introduced notation, the reader is referred to appendix 3.A.

3.3 Contract Model

Any incentive contract can be split up into its three components: cost, duration and scope. Hence, the model presented here also consists of three distinct components which can be used either jointly or separately for cases where one or more dimensions are irrelevant to the owner. For each dimension, the owner can opt to implement a linear, piecewise linear or non-linear contract. For the duration dimension in particular, a lump sum incentive attached to a specific deadline can also be included.

3.3.1 Cost Incentive Model

Since the outcome of this dimension is already expressed as a monetary amount, setting cost targets and evaluating the performance is relatively straightforward. The basic principle of a cost incentive can be illustrated using a simple linear incentive contract (Perry and Barnes, 2000). In such a contract the cost incentive (I^C) paid to the contractor is determined by multiplying the difference between the cost target (C^t) and the direct cost incurred (C) by a sharing ratio ($s^C \in [0,1]$), as given by equation 3.1. When the cost incurred is lower than the target cost, the difference is positive and a positive incentive (I^C) is awarded, the inverse also being true.

$$I^C = s^C(C^t - C) \tag{3.1}$$

All of the traditional linear cost contract types (see section 2.3.1) can be expressed as variations of this equation. This is done by varying the sharing ratio s^C, and in some cases capping the risk exposure of either owner or contractor for extreme outcome scenarios (e.g. when the cost exceeds a certain threshold the disincentive no longer increases and the owner carries the remainder of the downside risk, avoiding potential bankruptcy of the contractor).

[2]Should this not be the case, it would be irrelevant to implement any kind of incentive contract.

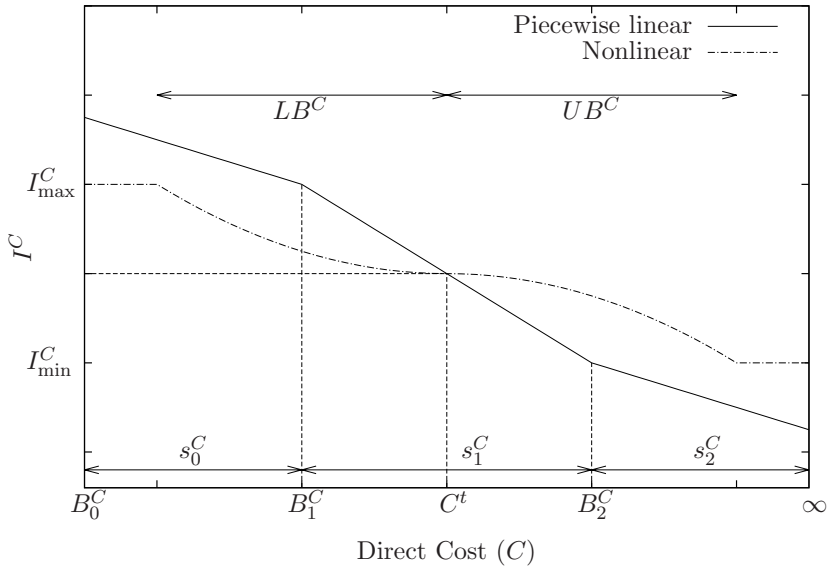


Figure 3.2: Piecewise linear and non-linear clauses.

An implementation of such a piecewise linear contract is visualised in figure 3.2, where the horizontal axis represents the possible performance outcomes for the contractor in terms of direct costs (C), and the vertical axis represents the incentive earned by the contractor (I^C). The piecewise linear contract in figure 3.2 defines three regions, numbered $r = \{0, 1, 2\}$. Each region r is defined by its lower bound B^C_r. For each of these regions a sharing ratio is defined as s^C_r. The contract also contains a target for the direct cost (C^t), which can be situated in any of the available regions. Equation 3.2 shows a mathematical expression for the 3-piecewise linear contract shown in figure 3.2, analogous expressions can of course be created for contracts with more dimensions and/or targets lying in other regions.

$$I^C = \begin{cases} s^C_1 \cdot (C^t - B^C_2) + s^C_2 \cdot (B^C_2 - C) & \text{if } C > B^C_2 \\ s^C_1 \cdot (C^t - C) & \text{if } B^C_1 \leq C \leq B^C_2 \\ s^C_1 \cdot (C^t - B^C_1) + s^C_0 \cdot (B^C_1 - C) & \text{if } C < B^C_1 \end{cases} \tag{3.2}$$

An alternative to the piecewise linear approach is the quadratic contract form as presented in Shr and Chen (2003). Other non-linear forms can of course also be used, but since their use is infrequent, only the quadratic form is included in this chapter. This quadratic form requires the owner to specify upper (UB^C) and lower (LB^C) bounds, which outline the region over which he wishes to spread his (dis)incentive. Note that these express the *distance from the target cost, rather than an absolute cost.* The magnitude of the incentive itself is defined using two parameters: $I^C_{\max} \in [0, +\infty[$ signifies the maximal possible

incentive, and $I^C_{\min} \in]-\infty, 0]$ which specifies the biggest disincentive amount. The target cost (C^t) has to be specified as well. For this contract type, the cost incentive amount can be calculated as follows:

$$I^C = \begin{cases} \min\left(I^C_{\max}; I^C_{\max} \cdot \left(\frac{C^t - C}{LB^C}\right)^2\right) & \text{if } C \leq C^t \\ \max\left(I^C_{\min}; I^C_{\min} \cdot \left(\frac{C - C^t}{UB^C}\right)^2\right) & \text{if } C > C^t \end{cases} \tag{3.3}$$

This type of incentive contract is visualised by the dotted line in figure 3.2. An owner wishing to use such a contract is left to determine the direct cost target (C^t), the maximal incentive and disincentive amounts ($I^C_{\max}$, $I^C_{\min}$), and the range wherein the incentive is to be distributed (LB^C, UB^C).

3.3.2 Duration Incentive Model

Duration incentives are also frequently used in project management. Since the finishing time of a project is easy to observe in an objective manner, it is also one of the easiest to implement. However, it does require the owner to attach a monetary valuation to the timeliness of project completion. An example of such a valuation is the road user cost used by the U.S. government when evaluating different bids for road refurbishment (El-Rayes and Kandil, 2005; Sillars, 2007).

Once this valuation has been determined, the owner has to decide on the fraction of this valuation to be awarded to the contractor as an incentive. Naturally the value transferred to the contractor should never exceed the gain of the owner.

Similarly to the cost incentive contracts of section 3.3.1, this can be done using a piecewise linear contract (equation 3.2), or a non-linear contract (equation 3.3). Equation 3.4 shows how the piecewise linear cost contract can be adjusted to a piecewise linear duration contract with similar assumptions. The difference between these two contracts is that rather than using a sharing ratio $s^C_r \in [0, 1]$, a valuation parameter v^D_r is used. This parameter defines the monetary amount the contractor earns per unit of time saved in a specific region, if the region is below the target duration specified in the contract. If the region is situated above the contract's target duration, the parameter signifies the disincentive amount. The non-linear contract for the duration dimension is identical to the non-linear contract for the cost dimension, and will not be repeated here.

$$I^D = \begin{cases} v^D_1 \cdot (D^t - B^D_2) + v^D_2 \cdot (B^D_2 - D) & \text{if } D > B^D_2 \\ v^D_1 \cdot (D^t - D) & \text{if } B^D_1 \leq D \leq B^D_2 \\ v^D_1 \cdot (D^t - B^D_1) + v^D_0 \cdot (B^D_1 - D) & \text{if } D < B^D_1 \end{cases} \tag{3.4}$$

The concept of time in a project-environment is inherently related to specific project deadlines. Oftentimes, projects have to be completed prior to a certain date to ensure their value to the project owner. (e.g. facilities which have to be constructed for Olympic games.) Since delivering a project prior to an exact date can be so important to the owner, incentive contracts are of course adjusted to represent this.

Ensuring that the contractor's valuation of this date is aligned with the owner's disproportionately large valuation (compared to the dates prior and after this deadline), can be done using a single lump sum incentive amount. When the contractor delivers the project prior to the agreed upon date, the agreed incentive amount is wholly awarded. In case the contractor fails to deliver on time, he does not receive anything. This very simple principle can be expressed as follows:

$$I^D = \begin{cases} I_{ls}^D & \text{if } D \leq D_{ls}^t \\ 0 & \text{if } D > D_{ls}^t \end{cases} \tag{3.5}$$

Where D_{ls}^t is the target date associated with the lump sum payment, and I_{ls}^D is the amount of the lump-sum payment. Naturally, such lump-sum incentive amounts can be combined with other duration-related incentive provisions such as piecewise linear or non-linear incentive contracts, since the presence of a deadline does not mean that additional time savings or delays have no value to the project owner.

3.3.3 Scope Incentive Model

The implementation of a scope incentive requires the adoption of a measurement method. Because scope envelops a large number of concepts, its measurement often relies on subjective estimates (De Wit, 1988). However, such subjective estimates are of little use as a basis for awarding incentive amounts, since the estimates of the owner and contractor will most likely be skewed in opposite directions. Stukhart (1984) has proposed the use of an external party to provide a more objective measurement in such cases, but even so there is potential for disputes between both parties. Moreover, the cost of hiring a third party also decreases the project profitability. The use of key performance indicators has been advocated by Rose (2008) as a method of making the performance measurement more objective. Whereas this method does indeed succeed in removing the subjectivity of scope performance, it does not include a formal method for aggregating the scope performance over the complete project. A solution for this issue is a top-down approach which defines a hierarchical structure for the complete project, defining the relative importance of various components until a degree of detail where objective evaluations are possible is reached (Love et al., 2010; Pollack-johnson and Liberatore, 2006; Song and AbouRizk, 2005). By

using such techniques, the scope performance of the complete project can be measured on a ratio scale. Hence, a value of 0 will always represent no work performed and comparative calculations can be made using different values of the scope (i.e. a scope of 2 is twice as valuable to the owner as a scope of 1).

By taking this approach, the analysis of the scope dimension can happen in a way which is largely analogous to the preceding two dimensions. Assuming that the contract specifies a minimally acceptable scope as well as a scope target, similar equations as those constructed for the cost incentive in section 3.3.1 can be constructed. The piecewise linear incentive contract can be adjusted for application on the scope dimension as follows:

$$I^S = \begin{cases} v_1^S \cdot (B_2^S - S^t) + v_2^S \cdot (S - B_2^S) & \text{if } S > B_2^S \\ v_1^S \cdot (S - S^t) & \text{if } B_1^S \leq S \leq B_2^S \\ v_1^S \cdot (B_1^S - S^t) + v_0^S \cdot (S - B_1^S) & \text{if } S < B_1^S \end{cases} \tag{3.6}$$

The key difference between the piecewise linear equation for the duration dimension (equation 3.4) is that the statements between brackets are inverted. This is logical since a larger scope value is more valuable than a smaller value. Again the valuation is done using the v_r^S parameters. These parameters represent the monetary valuation of a unit of scope within a certain region r.

$$I^S = \begin{cases} \min\left(I_{\max}^S; I_{\max}^S \cdot \left(\frac{S - S^t}{UB^S}\right)^2\right) & \text{if } S > S^t \\ \max\left(I_{\min}^S; I_{\min}^S \cdot \left(\frac{S^t - S}{LB^S}\right)^2\right) & \text{if } S \leq S^t \end{cases} \tag{3.7}$$

The adjustment of the non-linear contract for the scope dimensions is shown by equation 3.7. The sole difference between this equation and the expression for the cost dimension (equation 3.3) is that the positive and negative senses are inverted.

3.4 Trade-off Dynamics Model

Contractors subjected to an incentive contract face a four-dimensional trade-off decision. These dimensions are: *direct cost, duration, scope and contractor effort.* The first three of these dimensions are directly observed by the project owner and can potentially be used as a basis for an incentive provision. The fourth dimension represents the possibility for the contractor to enhance the project outcome using his own means.

3.4.1 Axioms

A number of axioms which are an extension of those proposed by Ghodsi et al. (2009) are used as a basis for the generation of realistic problem environments for the contractor. These axioms are assumed to be valid on the level of the (aggregated) project as well as on the level of individual activities. For sake of simplicity, the relationship between the different dimensions is considered on a two-by-two basis. A conceptual representation of these axioms is given by figure 3.3, and a visual representation of their implications is given by figure 3.4. The axioms can be summarised as follows:

1. The *direct cost* is a *non-increasing convex* function of the *duration* of the project. (The contractor will first use the cheapest methods for time reduction.) Or in mathematical terms: $\frac{\partial C}{\partial D} \leq 0$ and $\frac{\partial^2 C}{\partial D^2} \geq 0$.

2. The *direct cost* is an *non-decreasing convex* function of the *scope* of the project. (Increasing the scope becomes progressively more expensive as the low hanging fruit is picked first.) Or in mathematical terms: $\frac{\partial C}{\partial S} \geq 0$ and $\frac{\partial^2 C}{\partial S^2} \geq 0$.

3. The *direct cost* is a *non-increasing convex* function of the *contractor effort* of the project. In mathematical notation this becomes: $\frac{\partial C}{\partial E} \leq 0$ and $\frac{\partial^2 C}{\partial E^2} \geq 0$. This effort can be viewed as a way to influence the other relationships, i.e. making a decrease in duration or an increase in scope less expensive in terms of direct costs. Note that the cost in this case signifies the cost to the owner, not the contractor. The contractor's costs naturally increase when the effort is increased.

4. *Shortening the duration* of a *high-scope project* causes a *cost increase* which is *greater than* or equal to the cost increase in a low-scope project. Or in mathematical terms: $\forall S_1 > S_2 : \frac{\partial C(S_1)}{\partial D} \leq \frac{\partial C(S_2)}{\partial D} \leq 0$. Similarly, *increasing scope* will be *at least as costly* for a *shorter duration* project: $\forall D_1 < D_2 : \frac{\partial C(D_1)}{\partial S} \geq \frac{\partial C(D_2)}{\partial S} \geq 0$.

5. *Decreasing the duration* of a project will be *less than or equally costly* when the *effort invested* in the project *is greater*. This can be noted as: $\forall E_1 < E_2 : \frac{\partial C(E_1)}{\partial D} \leq \frac{\partial C(E_2)}{\partial D} \leq 0$. Analogously, a *smaller project duration* will - ceteris paribus - result in a *steeper slope* of the relation between invested *effort and project cost*: $\forall D_1 < D_2 : \frac{\partial C(D_1)}{\partial E} \leq \frac{\partial C(D_2)}{\partial E} \leq 0$.

6. *Increasing the scope* of a project will be *relatively cheap* when the *effort invested* in the project is *greater*, and vice versa. In mathematical terms this can be expressed as: $\forall S_1 > S_2 : \frac{\partial C(S_1)}{\partial E} \leq \frac{\partial C(S_2)}{\partial E} \leq 0$. Analogously, a *change in effort level* will have a *greater impact* on the project cost when the *effort invested is smaller*, or in mathematical notation: $\forall E_1 < E_2 : \frac{\partial C(E_1)}{\partial S} \geq \frac{\partial C(E_2)}{\partial S} \geq 0$.

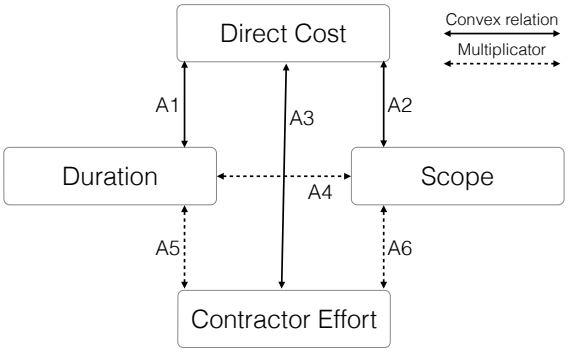


Figure 3.3: Schematic overview of the components of the contractor's trade-off and the associated axioms.

Note that the effect of the multiplicators presented in axioms 4 to 6 is represented visually in figure 3.4 by the change in slopes of the upwardly translated functions, and not the upward translation itself, which is a consequence of the first three axioms.

In order to quantify these relationships the *direct cost* is assumed to be the *dependent variable*, and the three other dimensions are considered to be *independent variables*. This is merely a design choice which is made to improve the comprehensibility of the model. This design choice is visualised by image 3.4, where the first three axioms represent *direct relationships* to the costs, and the latter three axioms are shown as *multiplicators*. An important note being that although the cost is the dependent variable in this model, it is still possible for the contractor to make trade-offs between the independent variables by making choices which inversely impact the cost of the project.

3.4.2 Mathematical Representation

The relationships expressed above will now be quantified. The complete model of the relationships is built in *three stages*: In the first stage, the basic relationships are *approximated using linear relationships* (axioms 1-3). Secondly, the *interaction effects* are modelled, using the *multiplicators* which represent axioms 4 to 6. Thirdly, the simplified linear relationships are transformed to more realistic *convex relationships*.

All the trade-offs in this model are considered to be discrete, meaning that the functions they create are piecewise linear functions which connect a discrete number of points. These discrete points are indexed with subscript i, j and k for duration, scope and effort respectively, assuming that: $i, j, k = 0, 1, ..., n$. These discrete points can be seen on the left panel of figure 3.5, counting from $i = 0$ for the rightmost point up to $i = n$ for the leftmost point of the trade-off. The following analysis assumes that there is an equal number of points $(n + 1)$ for each of the independent dimensions. This is not a prerequisite for the

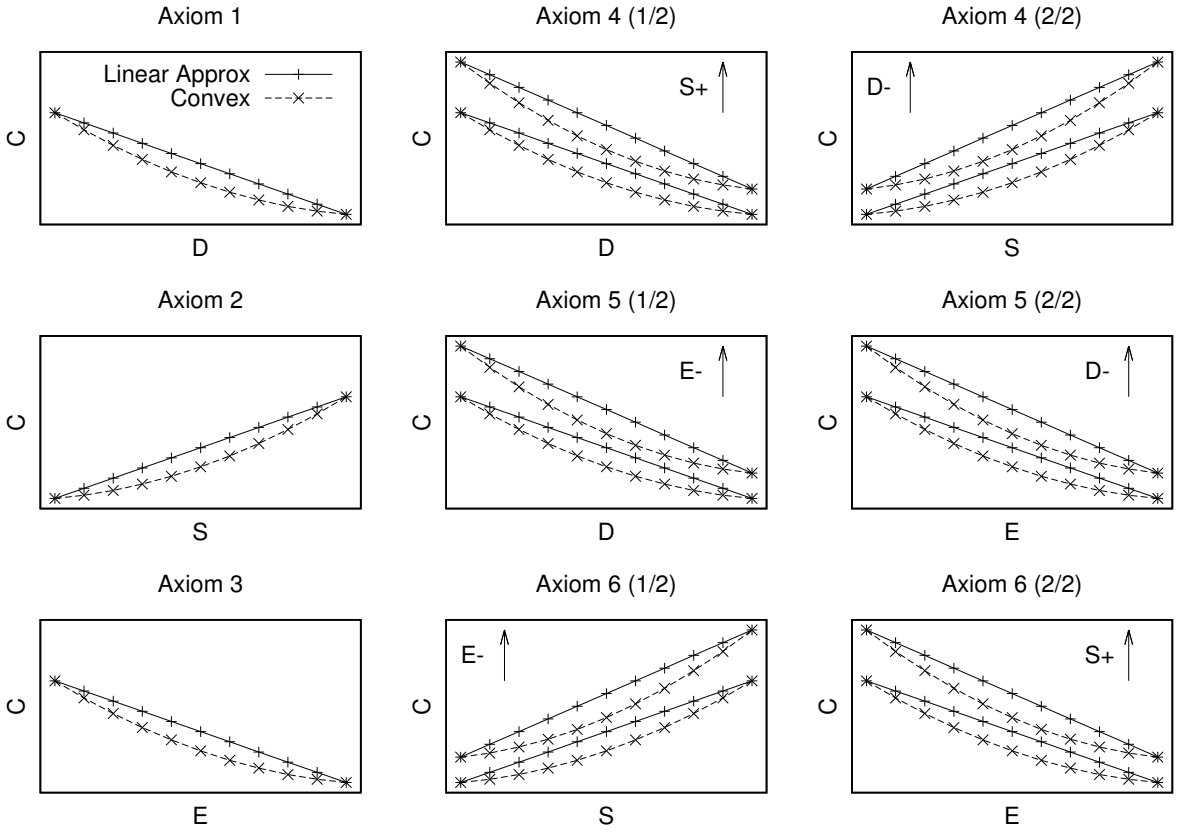


Figure 3.4: The axioms visualised.

analysis to be valid but is simply a choice to simplify the notation which is introduced.

Rather than modelling the direct cost (C), duration (D), scope (S) and effort (E) directly, the model defines Δ-variables which represent the *distance of these values from their lowest cost option*, which is indexed 0 (see the left panel of figure 3.5):

$$\Delta D_i = D_0 - D_i \tag{3.8}$$

$$\Delta S_j = S_i - S_0 \tag{3.9}$$

$$\Delta E_k = E_0 - E_k \tag{3.10}$$

Figure 3.5 illustrates this notation. The left panel shows the traditional relationship between the duration (D) on the horizontal axis and the cost (C) on the vertical axis. For each of the possible discrete duration steps, an associated distance can be defined to the

lowest cost option, as illustrated by ΔD_4. Associated with this ΔD_i-value, a cost value can be defined as ΔC_4^D, which is in effect the cost increase due to the decreased duration of the project. The right panel of figure 3.5 shows that by using these delta values on the axes, an increasing convex relationship can be defined.

By defining all the variables in this manner the formulae are simplified, since every dimension now has an increasing convex relationship to the cost dimension. Moreover, each of these relations starts at the origin (e.g. $(\Delta D_0, \Delta C_0) = (0, 0)$). This also simplifies the definition of the *convexity magnitude* (see section 3.4.2.3).

3.4.2.1 Linear relationships

The first expression is a basic linear relationship between the cost (as a dependent) and the three other dimensions. This relationship is in effect the sum of the impacts of the duration, scope and effort levels respectively. This is expressed by equation 3.11, where ΔC_i^D represents the impact of selecting duration mode i on the costs, and ΔC_j^S and ΔC_k^E are similar metrics for scope and effort respectively. This is also illustrated on the right panel of figure 3.5, where the vertical axis shows ΔC_i^D graphically.

In order to define this relationship, the lowest possible direct cost has to be given as a parameter ($C_{\min} = C|\Delta D_i = \Delta S_j = \Delta E_k = 0$). Also for each of the independent dimensions, the slope of the curve has to be defined: S_D, S_S and S_E. This slope represents both the slope of the linear approximation as well as the average slope of the convex curve, since the start and end points of the linear approximation and the convex curve are made to coincide. Due to the Δ transformation of the independent variables, these slopes can all be represented as positive numbers, hence the absolute values in equation 3.12.

$$C_{ijk} = C_{\min} + \Delta C_{ijk} = C_{\min} + \Delta C_i^D + \Delta C_j^S + \Delta C_k^E \tag{3.11}$$

$$C_{ijk} = C_{\min} + |S_D|\,\Delta D_i + |S_S|\,\Delta S_j + |S_E|\,\Delta E_k \tag{3.12}$$

Equation 3.12 is in effect a simplified linear representation of the first three axioms. Note that the independent dimensions are indexed separately, since these dimensions can naturally be at different positions. (i.e. The contractor can select the positions of these dimensions separately, the effect of which will be reflected in the total cost of the project.)

3.4.2.2 Interaction Effects

Axioms 4 to 6 are then added to the linear expression through the definition of of the *slope multiplicators*. These are defined in the following form: m_a^b is the maximal impact

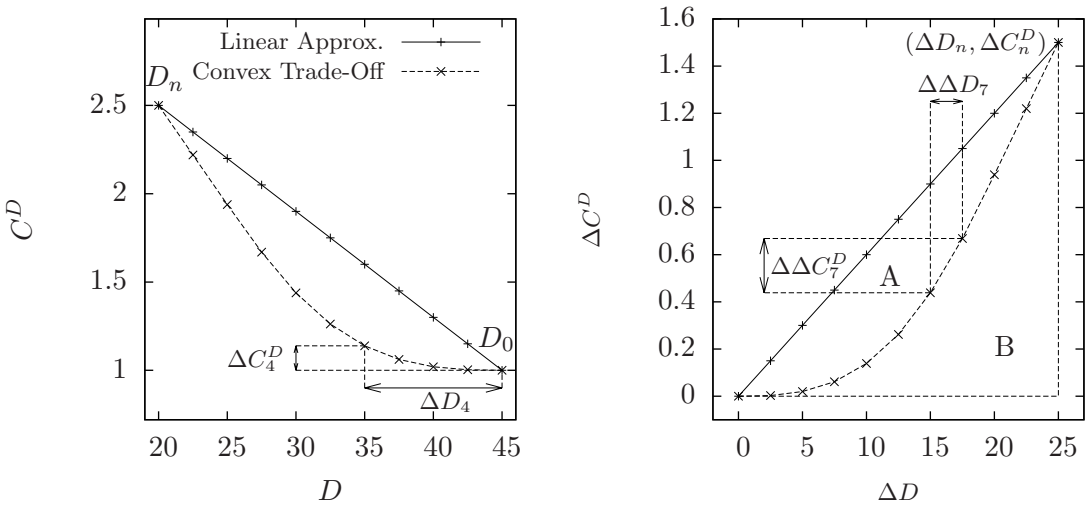


Figure 3.5: Illustration of the mathematical description, using the relationship between duration and cost as an example.

of dimension b on the slope of the relationship between the direct cost and dimension a, expressed as a fraction added to the original slope of the curve. For example $m_D^S = 0.2$ indicates that the slope of the duration curve will be 20% steeper when ΔS is at its maximal value.

$$\begin{aligned} C_{ijk} =& C_{\min} + |S_D|\,\Delta D_i \left(m_D^S \frac{\Delta S_j}{\Delta S_n} + 1\right)\left(m_D^E \frac{\Delta E_k}{\Delta E_n} + 1\right) + \\ & |S_S|\,\Delta S_j \left(m_S^D \frac{\Delta D_i}{\Delta D_n} + 1\right)\left(m_S^E \frac{\Delta E_k}{\Delta E_n} + 1\right) + \\ & |S_E|\,\Delta E_k \left(m_E^D \frac{\Delta D_i}{\Delta D_n} + 1\right)\left(m_E^S \frac{\Delta S_j}{\Delta S_n} + 1\right) \end{aligned} \tag{3.13}$$

The effect of adding these multiplicators is shown in figure 3.4, where the upwardly translated curves have a steeper slope than the original curves.

3.4.2.3 Convexity

The third and final step is the conversion of these linear relationships into more realistic convex functions. To express the degree of convexity of a relation quantitatively, the *convexity magnitude* (CM) metric is introduced in this chapter. The logic behind this metric can be explained by looking at the right panel of figure 3.5 where a triangle is formed by the simplified linear relationship and the projection towards the horizontal axis.

Two zones are defined within this triangle such that zone A is the the surface between the simplified linear relationship and the convex curve and zone B is the area below the convex curve. Hence, the total surface of the triangular surface is equal to $A+B$. The CM is now defined as the fraction of the triangle consisting of zone A. Assuming a *uniform step size* ($\Delta\Delta D_i = \frac{\Delta D_n}{n}, \forall i = 1, ..., n$), for the duration dimension this metric can be calculated as follows (see appendix 3.B for a full derivation):

$$\begin{aligned} CM^D &= \frac{A}{A+B} \\ &= 1 - \frac{2 \cdot B}{\Delta D_n \cdot \Delta C_n^D} \end{aligned} \tag{3.14}$$

$$B = \frac{\Delta D_n}{n} \sum_{i=1}^{n} \left[\Delta C_{i-1}^D + \frac{\Delta C_i^D - \Delta C_{i-1}^D}{2} \right] \tag{3.15}$$

In the formula above $\Delta D_n / \Delta C^D$ can of course be replaced by $\Delta S_n / \Delta C^S$ or $\Delta E_n / \Delta C^E$ to model the degree of convexity of the scope (CM^S) and effort (CM^E) respectively. Logically the value of this metric lies in the interval $[0, 1[$, where a value of 0 represents a perfectly linear relation and higher values represent different gradations of convexity. The actual upper bound on the convexity depends on the number of discrete points n: $CM^D_{\max} = \frac{n-1}{n}$. Although this upper bound is attainable, trade-off curves approaching this bound will often have non-increasing regions and very steep angles.

3.4.2.4 Specific Project Cases

The trade-off points for a specific scenario can either be defined explicitly when full information on all feasible execution methods is available, or fitted using partial information in combination with the aforementioned axioms. Hence, although it is assumed that the majority of projects will comply with the aforementioned assumptions, this is not a prerequisite for the application of the model.

3.5 Contract Evaluation Model

The preceding two models enable a quantitative representation of contract and project trade-off structures respectively. The contract evaluation model presented in this section focuses on how appropriate the use of a certain contract is for a given project.

From the perspective of the owner, the optimal contract has three properties. First of all, it maximises the gain the owner can expect from the project. Secondly, it guarantees that the evaluation of the contractor is well aligned with that of the owner. Thirdly, the

contract should guarantee an adequate return for the contractor. An adequate contractor return has two facets, on the one hand the absolute incentive amount has to be substantial enough to be relevant to the contractor, on the other hand, the range between the highest and lowest incentive amount has to be large enough such that the contractor is motivated to locate the optimum pay-off.

The model presented in this section uses a single index $l = 1, ..., m$ which represents the m different possible outcome scenarios, in no particular order. Such an outcome scenario is simply an attainable point of the trade-off model (i.e. a combination of cost, duration, scope and effort), for which the associated incentive amount has been calculated. When combined with a trade-off model as presented in section 3.4 the value of m is equal to $(n+1)^3$.

3.5.1 Maximisation of Expected Owner Gain

First and foremost, the goal of the project owner is to maximise his expected return. The contract evaluation model presented here calculates this amount by assuming that the contractor is a risk-neutral profit-maximiser (Gibbons, 2005). Given that the choice of a trade-off point is controlled by the contractor, the expected profit of the project owner is the profit for the trade-off point where the contractor's profits are maximised. Hence, the first step in calculating the expected net owner gain ($E[NOG]$) is the maximisation of the contractor's profit function.

When evaluating the discrete set of scenarios from the perspective of the contractor, two elements have to be considered for every possible project outcome scenario l. First of all, the cost of effort invested by the contractor (E_l^{cost}) and secondly, the total incentive amount earned (I_l^{tot}). This total incentive amount is simply the sum of the cost incentive (I_l^C), duration incentive (I_l^D) and the scope incentive (I_l^S) amounts (which are defined by the contract model, as presented in section 3.3). Using these two elements, the net contractor gain (NCG_l) can be calculated for every scenario:

$$NCG_l = I_l^{tot} - E_l^{cost} = I_l^C + I_l^D + I_l^S - E_l^{cost} \tag{3.16}$$

However, it must not be neglected that there is an opportunity cost associated with investing effort in a project, since the same effort could also earn a certain return elsewhere. Hence, the net contractor gain (NCG_l) has to be corrected for this opportunity cost using the return on investment (ROI^E) this effort could earn in other projects or activities. The adjusted NCG can be calculated as:

$$NCG_l^{adj} = I_l^{tot} - E_l^{cost}(1 + ROI^E) \tag{3.17}$$

This adjustment decreases the valuation of the scenarios with the opportunity cost of the effort investment. By doing this, the different NCG_l^{adj} values can be directly compared. In the case where there is no ulterior investment possibility for the contractor's effort, ROI^E can be set to zero.

The owner's net gain from a specific scenario (NOG_l) is equal to the value derived from the direct cost (C_l), duration (D_l) and scope (S_l) dimensions diminished by the total incentive amount awarded to the contractor (I_l^{tot}).

$$NOG_l = OV(C_l, D_l, S_l) - I_l^{tot} \tag{3.18}$$

Where OV is a function expressing the financial value of a certain outcome of the project for different values of direct cost, duration and scope. This function can be further detailed as a simple sum of the valuation of each of these three dimensions:

$$OV(C_l, D_l, S_l) = OV^C(C_l) + OV^D(D_l) + OV^S(S_l) \tag{3.19}$$

As for the cost component, the owner's valuation (OV^C) is inversely proportional to the direct cost outcome:

$$OV^C(C_l) = -C_l \tag{3.20}$$

The time valuation is slightly more complex, containing both the possibility of a linear value per time unit such as a road user cost, and the possibility of a lump sum value representing an important deadline.

$$OV^D(D_l) = \begin{cases} \mu_D \cdot (-D_l) & \text{if } D_l \geq DL \\ \mu_D \cdot (-D_l) + \mu_{ls} & \text{if } D_l < DL \end{cases} \tag{3.21}$$

In equation 3.21, μ_D represents the monetary valuation of an unit of time by the owner (e.g. 1000 Euro/day). This is similar to the road user cost which is often defined in $A + B$ contracting methods for roadwork manufacturing (Sillars, 2007). Analogously, the parameter μ_{ls} is used to represent the monetary value to the owner of satisfying a certain deadline DL. Note that the parameter DL is different from the D_{ls}^t parameter introduced in section 3.3.2, the former signifies a deadline relevant to the owner, whereas the latter is a deadline clause included in an incentive contract to which the contractor is subjected. These two dates are not necessarily the same, since the owner can choose to insert additional buffer in the contract by setting the contractor's deadline prior to his own (i.e. $DL \geq D_{ls}^t$ for rationally designed contracts).

A similar construction is created for the valuation of scope, as seen in equation 3.22.

Based on the premise that scope can be measured on a ratio scale (see section 3.3.3), it suffices to introduce μ_S as the monetary value of an unit of scope to the owner.

$$OV^S(S_l) = \mu_S \cdot S_l \tag{3.22}$$

Using the formulae above, the play-off functions for both the owner and the contractor have been fully defined as NOG_l and NCG_l^{adj} respectively. As stated above, the optimal contract design has three evaluation criteria: a maximal return for the owner, an accurate alignment of the evaluation of the contractor and the owner, and an adequate play-off for the contractor. The first of these criteria can be expressed as follows:

$$E\left[NOG\right] = \max\left[NOG_l | \left(NCG_l^{adj} = \max_l\left[NCG^{adj}\right]\right)\right] \tag{3.23}$$

Equation 3.23 effectively states that the expected gain of the owner is equal to the gain he receives when the contractor selects the scenario which maximises the contractor's own return. In the case where there are multiple scenarios in which the contractor's profits are maximised, the maximal net owner gain is used. The reasoning behind this being that given the opportunity to do so, the contractor will chose to maximise the owner's profits if this does not affect his own return.

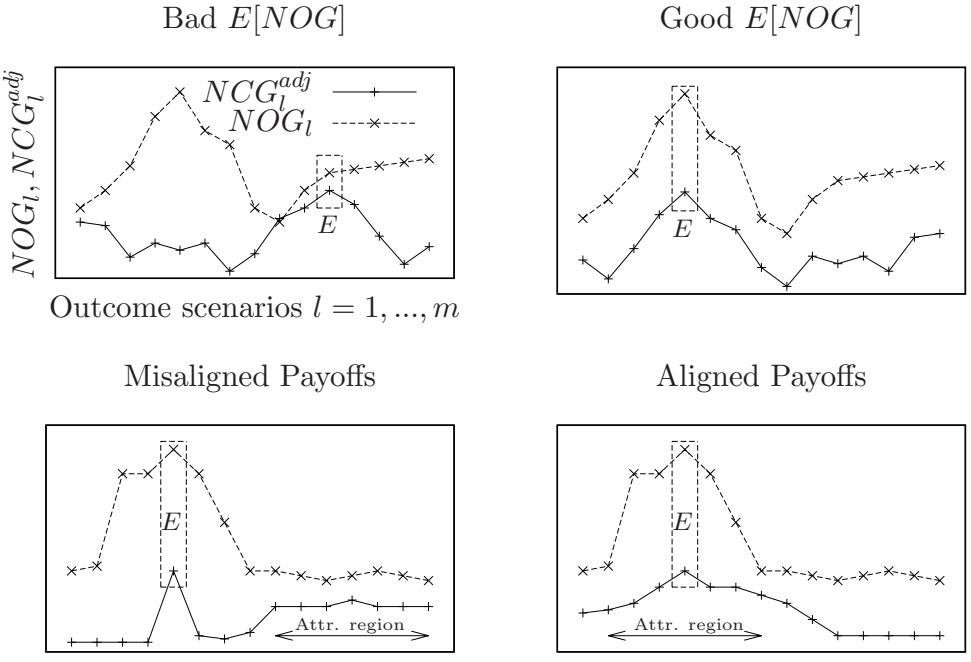


Figure 3.6: Illustration of the optimisation objectives

The top two panels of figure 3.6 illustrate the expected owner profit ($E\left[NOG\right]$) max-

imisation objective. On the horizontal axis the all the potential outcome scenarios (i.e. the trade-off points which can be chosen by the contractor) are listed in no particular order. The vertical axis represents the profit for the owner and contractor for the respective scenarios. As was formerly stated, the contractor is a profit maximiser and will therefore always select the scenario which maximises his profits ($NCG_l^{adj} = \max_l \left(NCG_l^{adj}\right)$). Hence, the owner profit (NOG_l) associated with this scenario is defined as the expected net owner gain ($E\left[NOG\right]$). This expected outcome is indicated by the rectangle and the letter E in figure 3.6. Assuming the owner is confronted with a choice between the contract represented by the upper left panel and the contract represented by the upper right panel, the owner will prefer the right contract, since this is likely to result in an outcome which yields a much greater net owner gain (NOG). The situation in the top left pane is in fact equivalent to rewarding a certain strategy, while hoping a different strategy will be followed (Kerr, 1975).

Note that the example presented in figure 3.6 is simplified in that a change in the contract structure is likely to change the owner's pay-off function as well as the function of the contractor. This is one of the reasons for the complex nature of incentive contract design.

3.5.2 Evaluation Alignment

The two bottom panels of figure 3.6 present an intuitive example of the importance of evaluation alignment. Again the horizontal axis lists all possible outcome scenarios in no particular order, and the vertical axis represents the contractor (NCG_l^{adj}) and owner (NOG_l) gains for the respective scenarios. When comparing the two bottom panels, it is clear that both contracts have an identical expected net owner gain $E\left[NOG\right]$. Hence, if this was the only evaluation criterion used, the owner would be indifferent between the contract resulting in the bottom left and the contract resulting in the bottom right scenario. However, when looking at the most attractive regions for the contractor, excluding the contractor's optimal point, a case can be made for the contract represented in the bottom right. Namely, it is clear that for the bottom left contract the *attractive region* coincides with the least attractive points for the contractor. Hence, should external influence cause the optimal point to be unattainable, it is likely that the outcome will gravitate towards such an unattractive point. The opposite is true for the contract on the bottom right, where even if the expected point should not be obtained for whichever reason, the other points still result in relatively attractive outcomes for the owner.

The theoretical example in the bottom panels of figure 3.6 can be linked to the concept of balanced contract design in project management practice (Abu-Hijleh and Ibbs, 1989). For example, assume the only time-related incentive of a contract is a lump-sum provision

awarded in case a certain deadline is met, but the contract also includes a substantial cost incentive to encourage cost savings. Should it at some point during the execution of this project become apparent to the contractor that it will be impossible to deliver the project before said deadline, it is highly likely that the contractor will attempt to maximise his cost incentive and completely neglect the time dimension. This would of course have a dramatic effect on the value of the project to the contract owner, who of course attaches a substantial value to the timely delivery of the project. The concept of pay-off alignment is a way of formalising this issue.

In order to quantify the alignment of the evaluation of the owner and contractor, both values are expressed in relative terms, meaning that they are rescaled in the range $[0, 1]$. This results in the following relative expressions for contractor and owner respectively:

$$RNCG_l^{adj} = \frac{NCG_l^{adj} - NCG_{\min}^{adj}}{NCG_{\max}^{adj} - NCG_{\min}^{adj}} \tag{3.24}$$

$$RNOG_l = \frac{NOG_l - NOG_{\min}}{NOG_{\max} - NOG_{\min}} \tag{3.25}$$

Using these relative expressions, a perfect evaluation alignment of both parties can be defined as the situation where the following is true: $RNCG_l^{adj} = RNOG_l \ \forall l = 1, ..., m$. Naturally this optimal situation can rarely be attained in practice. Hence the need to develop metrics to measure the degree of deviation from this optimal situation. The following two metrics are used in order to express the degree of alignment (i.e. the deviation from this optimal situation):

$$MD = \frac{1}{m} \sum_{l=1}^{m} \left[RNCG_l^{adj} - RNOG_l \right] \tag{3.26}$$

$$MAD = \frac{1}{m} \sum_{l=1}^{m} \left| RNCG_l^{adj} - RNOG_l \right| \tag{3.27}$$

Equation 3.26 measures the mean deviation (MD) of the relative gains of both parties. The value of this metric lies within the range $[-1, 1]$, negative values indicating that there are more scenarios where the owner gains are larger than those of the contractor and positive values indicating that there are more scenarios where the inverse is true. Hence, the value of this metric should be as close to zero as possible.

The major flaw in equation 3.26 is of course that opposite deviations cancel each other out. Hence, the mean absolute deviation (MAD) is presented in equation 3.27. The MAD always has a value in the range $[0, 1]$, and gives an accurate reading of the overall deviations, regardless of their sign. By using the MAD to measure the average alignment

in combination with the MD to measure possible skewness, the overall alignment of the evaluations can be quantified.

3.5.3 Constraints Enforcing Sensible Contract Design

The preceding two optimisation criteria can be extended by adding constraints enforcing the contracts to adhere to certain rules for good incentive contract design. Such guidelines have been regularly proposed in the literature and often include advice on minimal sharing ratios (Perry and Barnes, 2000), and a bias towards positive incentives rather than disincentives (Rose and Manley, 2010). The remainder of this section presents two constraints which can be used to enforce certain principles when evaluating contracts.

The first constraint guarantees that the maximal incentive earned by the contractor is larger than or equal to a certain minimal value. If this would not be enforced, the optimal solution could potentially gravitate towards contracts which impose large disincentives on contractors. Several authors have already indicated that such contracts are undesirable because they undermine the contractor's motivation (Abu-Hijleh and Ibbs, 1989; Rose and Manley, 2010).

The level at which this constraint is set is of course strongly project-specific, and several authors have quoted different values depending on the industry in which they operated. Abu-Hijleh and Ibbs (1989) examined projects in industrial construction and found incentive amounts up to 2.62% of total project cost. A study by Veld and Peeters (1989) found that this percentage was approximately 0.7% in the aerospace industry. A much higher fraction is apparently used in road construction in the US, where an incentive cap of approximately 5% of the total project cost is recommended (Tang et al., 2008).

By defining the maximal incentive which can be earned by the contractor as $I^{tot}_{\max}$ and the lower bound for this value as $LB_{I^{tot}_{\max}}$, this constraint can be formulated as:

$$I^{tot}_{\max} \geq LB_{I^{tot}_{\max}} \tag{3.28}$$

The second constraint is introduced to guarantee that there is sufficiently variability in the pay-off of the contractor. Insufficient variability will impede the contractor's motivation to locate the optimal pay-off region. Hence, $R_{I^{tot}}$ is defined as the minimal range required for adequate contractor motivation. The difference between the maximal ($I^{tot}_{\max}$) and minimal ($I^{tot}_{\min}$) incentive awarded to the contractor should then be at least as great as this minimal range.

$$I^{tot}_{\max} - I^{tot}_{\min} \geq R_{I^{tot}} \tag{3.29}$$

To illustrate the workings of the different models, as well as the interactions between

them, section 3.6 presents a simplified fictitious example.

3.6 Fictitious Example

A retail store owner wishes to renovate his store interior and has employed a contractor to refurbish the entire store. For his services, the contractor receives a fixed fee of 10,000 Euro, plus all the direct material costs (C). The quality (scope) of the work performed by the contractor will be judged upon completion and scored on a ratio scale in the range $[1, 2]$, where 1 indicates the lowest quality and 2 indicates the highest quality.

Specific targets are specified with respect to the outcome dimensions of the project: the total cost of materials is expected to be 6,000 Euro (C^t), the project should take approximately 10 days to complete (D^t), and the quality of the project is expected to be at least 1.5 (S^t) on the aforementioned ratio scale. In order to elicit superior performance from the contractor, incentives are also included in the contract. The owner has decided that the impact of possible performance fluctuations are to be shared equitably between both parties. For the cost dimension this means that the sharing ratio s^C (see equation 3.1) is set to 0.5. With respect to the duration of the project, the owner estimates that he loses 1,000 Euro in income for each day the store has to remain closed for the refurbishment, hence the per diem (dis)incentive amount (v^D, see equation 3.4) is set at 500 Euro. The quality of the work performed has an impact on the time until a new renovation will be needed; the highest grade performance will last approximately 2 years longer than the lowest grade performance. The owner has calculated that he can save 2,500 Euro by being able to postpone the next renovation by two years. Hence, the valuation parameter for the scope incentive (v^S, see equation 3.6) is set to 1,250 Euro per unit of scope.

During the execution of the project, the contractor is able to allocate resources in the way which optimises the return (s)he receives from the project. This implies that the contractor selects a point on the cost, duration, scope and effort trade-off spectrum (see section 3.4). The contractor can for example choose to decrease the duration (D) by using fast-drying paints. However, these paints are either more expensive, causing a rise in the direct cost paid by the owner (C) or of inferior quality when compared to traditional paints, negatively influencing the quality (scope, S) of the result. The effort the contractor invests in the project is reflected by the number of labourers (s)he assigns to the project. Increasing the number of labourers can have a positive influence on the cost, duration or scope of a project, depending on the specific instructions given to personnel. Since the labour cost is covered by the contract's fixed fee rather than its variable cost component, increasing the number of labourers increases the contractor's cost with 500 Euro per labourer. Moreover, the contractor also takes into account an opportunity cost

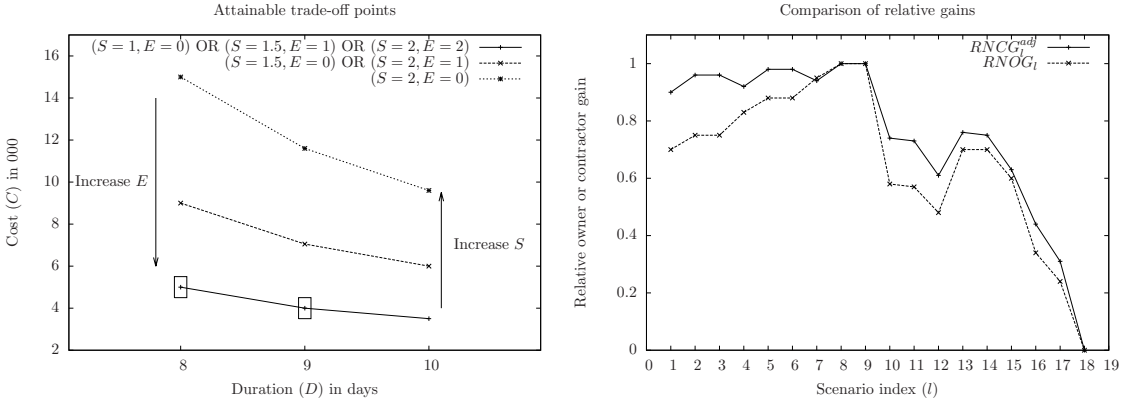


Figure 3.7: Trade-off points and relative gains for the sample project.

Scenario (l)	Cost (C)	Duration (D)	Scope (S)	Effort (E)	Effort Cost (E_l^{cost})	I^C	I^D	I^S	I^{tot}	NCG_l^{adj}	NOG_l	$RNCG_l^{adj}$	$RNOG_l$
1	3,500	10	1	0	0	1,250	0	-625	625	625	-11,000	0.90	0.70
2	4,000	9	1	0	0	1,000	500	-625	875	875	-10,500	0.96	0.75
3	5,000	8	1	0	0	500	1,000	-625	875	875	-10,500	0.96	0.75
4	3,500	10	1.5	1	500	1,250	0	0	1,250	700	-9,750	0.92	0.83
5	4,000	9	1.5	1	500	1,000	500	0	1,500	950	-9,250	0.98	0.88
6	5,000	8	1.5	1	500	500	1,000	0	1,500	950	-9,250	0.98	0.88
7	3,500	10	2	2	1,000	1,250	0	625	1,875	775	-8,500	0.94	0.95
8	4,000	9	2	2	1,000	1,000	500	625	2,125	1,025	-8,000	1.00	1.00
9	5,000	8	2	2	1,000	500	1,000	625	2,125	1,025	-8,000	1.00	1.00
10	6,000	10	1	0	0	0	0	-625	-625	-625	-13,500	0.58	0.45
11	7,050	9	1	0	0	-525	500	-625	-650	-650	-13,550	0.57	0.45
12	9,000	8	1	0	0	-1,500	1,000	-625	-1,125	-1,125	-14,500	0.45	0.35
13	6,000	10	1.5	1	500	0	0	0	0	-550	-12,250	0.60	0.58
14	7,050	9	1.5	1	500	-525	500	0	-25	-575	-12,300	0.59	0.57
15	9,000	8	1.5	1	500	-1,500	1,000	0	-500	-1,050	-13,250	0.47	0.48
16	9,600	10	2	0	0	-1,800	0	625	-1,175	-1,175	-14,600	0.44	0.34
17	11,600	9	2	0	0	-2,800	500	625	-1,675	-1,675	-15,600	0.31	0.24
18	15,000	8	2	0	0	-4,500	1,000	625	-2,875	-2,875	-18,000	0.00	0.00

Table 3.1: Scenarios and associated results for the sample project.

of 10% (ROI^E) since the labourers could otherwise be assigned to other projects.

The left panel of figure 3.7 visualises the 18 attainable trade-off points in this theoretical scenario. Specifically, the graph shows the relationship between project duration and cost for various levels of effort and scope. Note that multiple lines have been made to coincide to improve the readability of the example. Increasing the effort invested in the project causes a downward translation of the cost-duration trade-off curve, whereas an increase in the scope of the project has the opposite effect, as indicated on the graph itself. These trade-off combinations are also listed in table 3.1, together with the incentive amounts associated with each scenario. Using the total incentive amount (I^{tot}) together with the formulae from section 3.5.1, both the absolute and relative gains for the owner (NOG_l & $RNOG_l$) and the contractor (NCG_l^{adj} & $RNCG_l^{adj}$) have also been calculated.

Given this set of pay-offs, the contractor will choose either scenario 8 or 9, both of which yield the highest contractor gain (NCG_l^{adj}). Incidentally, these scenarios also result

in the highest values for the owner gain (NOG_l). However, the fact that these two values coincide does not mean that this is the best contract for this scenario, other contracts may result in a higher expected gain for the contractor, regardless of the fact that that the highest gains for both parties coincide or not.

The right panel of figure 3.7 compares the relative gains for both parties in each of the available scenarios l. Due to the equitable sharing policy implemented by the owner, it is clear that the fluctuations of both actors are largely similar. This is also reflected in the MAD and MD values which equal 0.094 and 0.093 respectively. The fact that the magnitude of the MD indicator is similar to that of the MAD indicator indicates that there is considerable skew, meaning that deviations are largely biased in one direction. This can also be seen on the right panel of figure 3.7, which shows that the relative owner gain ($RNOG_l$) is consistently higher than the relative contractor ($RNCG_l^{adj}$) gain.

3.7 Evaluating the Performance of Contract Sets

One of the key objectives of this research is to evaluate the performance of different types of contracts. However, due to the presence of multiple objectives, comparing performance is not trivial. In order to evaluate how different contract types perform, all Pareto-efficient contracts of a specific type are considered jointly. These *local* Pareto frontiers are then compared to a *global* Pareto frontier, which can include all contracts regardless of their type. This section discusses the methods used to make this comparison.

A qualitative interpretation of such a Pareto frontier is the set of viable options created by a specific type of contract. Ideally, a project owner could limit the complexity of her/his decision process by using only a limited number of contract types, while still having a sufficiently broad choice of options compared to the globally best performing solutions (i.e. the Pareto frontier that is created when all contract types are allowed).

As was stated in section 3.5, the performance of a contract is measured on two key dimensions: expected owner gain ($E[NOG]$), and the alignment of the relative pay-offs of the contractor and the owner, as measured by the MD and MAD metrics. Since the problem at hand has multiple objectives, the performance has to be evaluated accordingly. This is done by constructing a global three dimensional ($E[NOG], MAD, MD$) Pareto frontier, containing all the contracts for a problem environment which are not dominated. The performance of different contract sets (i.e. the contracts belonging to a specific contract group, can then be compared to this global Pareto frontier. When making such comparisons, three aspects are of importance (Okabe et al., 2003; Sarker and Coello, 2002): (i) the distance to the global Pareto front, (ii) the spread of the solutions found, and (iii) the number of elements found.

(i) Distance to the global Pareto front. Frequently used metrics to measure the distance between the global Pareto front and approximations thereof include the error rate (ER) , the generational distance (GD), the inverted generational distance (IGD) as well as the Hausdorff distance (d_H) (Sarker and Coello, 2002; Van Veldhuizen, 1999a,b). However, each of the aforementioned metrics displays certain shortcomings: the error rate fails to measure the precise distance to the global Pareto front, both the GD and IGD tend to zero as the number of elements on the tested or global frontier is increased, and the Hausdorff distance only measures the worst case scenario. Hence, the averaged Hausdorff distance (Δ_p), which was introduced by Schutze et al. (2012) to mitigate these shortcomings, is used as a metric for the distance from the tested set to the best known Pareto front, this metric is calculated as follows:

$$\Delta_p(X,Y) = \max\left[\left(\frac{1}{N}\sum_{i=1}^{N} dist(x_i,Y)^p\right)^{\frac{1}{p}}, \left(\frac{1}{M}\sum_{i=1}^{M} dist(y_i,X)^p\right)^{\frac{1}{p}}\right] \tag{3.30}$$

Where X and Y are the two sets between which the distance is to be measured (i.e. the global Pareto front and the tested front). These sets contain N and M points respectively, which are represented as x_i and y_i. The function $dist$ measures the distance from a point to the closest point in the other set, using simple euclidean distances. Finally, p is equal to the number of dimensions in the Pareto front (i.e. $p = 3$ in this study). The coordinate system used to calculate this metric in the experiments below has been standardised to the range $[0, 1]$ for each of the dimensions measured, to avoid one of the dimensions having a more substantial impact than the others merely due its measurement scale. For more information regarding this metric, the reader is referred to Schutze et al. (2012).

The calculation of the Δ_p metric is illustrated by figure 3.8. To allow for a simple visual representation, two rather than three evaluation dimensions are considered in this example. By visual inspection alone, it is hard to determine which of these two sets is closest to the global Pareto front. The Δ_p metric can be used to compare both of these sets ($S1$ and $S2$) to the global frontier (G). The first step is to calculate the distance from each point on $S1$ to the global Pareto front, which is simply the euclidean distance to the closest point on the global front. This is illustrated for the first set ($S1$) in the top right panel of figure 3.8. By taking the power mean of all these distances, the first of the two expressions within the square brackets of formula 3.30 is found. The calculation of the second expression is simply the inverse of the first and is done by taking the power mean of the distances from every point on the global front to the first set of points ($S1$). This is illustrated for the first set $S1$ by the bottom left panel of figure 3.8. By taking the maximum of these two expressions, the "worst case"-averaged distance is acquired.

When calculating the result for this small example, it appears that the second set ($S2$) is somewhat closer to the global Pareto front with a Δ_p value of 0.48, compared to a Δ_p value of 0.57 for the first set ($S1$)

(ii) Spread of solutions. To determine the quality of the spread of the tested Pareto front, a very simple metric is proposed which compares the distance between the extreme outcomes for each dimension of the tested front to the distance between the extreme outcomes for the global Pareto front. For $E[NOG]$ this can be calculated as follows:

$$spr(E[NOG]) = \frac{E[NOG]^{test}_{\max} - E[NOG]^{test}_{\min}}{E[NOG]^{glob}_{\max} - E[NOG]^{glob}_{\min}} \tag{3.31}$$

In equation 3.31, $E[NOG]^{test}_{\max}$ and $E[NOG]^{test}_{\min}$ represent the highest and lowest values for the $E[NOG]$ metric in the Pareto front which is compared to the global front. Similarly, $E[NOG]^{glob}_{\max}$ and $E[NOG]^{glob}_{\min}$ represent the highest and lowest values for $E[NOG]$ which can be observed on the global Pareto front. Note that since the global Pareto front will always include the best known solutions the following is always true: $E[NOG]^{glob}_{\min} \leq E[NOG]^{test}_{\min} \leq E[NOG]^{test}_{\max} \leq E[NOG]^{glob}_{\max}$. Hence, the spread will always be a value in the range $[0, 1]$. The graphical interpretation of this metric is illustrated in the bottom right panel of figure 3.8. For this example, it is clear that the spread of the second set ($S2$) with regard to the $E[NOG]$ metric is superior to the spread of the first set ($S1$). Since the spread for the MD and MAD dimensions is calculated analogously, the respective equations for these dimensions will not be repeated here.

(iii) Number of elements. The number of elements found on the tested Pareto front will not be considered as a metric in this chapter, since this number will of course be strongly dependent on the number of contracts that have been tested. This number is of course an aspect of the experimental set-up, rather than a property of the contract structure itself.

3.8 Computational Experiments

The formal models presented in the preceding sections make it possible to carry out computational experiments using carefully constructed datasets. In order to structure the analysis, the following two concepts are defined:

Contract type: A collection of contracts which adhere to certain design principles with respect to the dimensions which are incentivised, the formulae used to calculate the incentive and the owner and contractor accountability in the case of extremely positive or negative outcomes.

Environment: The project setting in which a contract operates, which is formed by combining a trade-off model (see section 3.4) and an evaluation model (see section 3.5).

The goal of these experiments is to compare the performance of different contract types, as well as the impact of differences in the environments in which contracts operate. Differences in environments can of course be due to differences in the trade-off model as well as differences in the evaluation model.

The computational experiments provide answers to four key research questions: (1) How do different contract types perform in practice? (2) How do environmental factors (i.e. the properties of the project trade-off and the evaluation models) influence the performance of these different contract types? (3) How substantial are the added benefits of multi-dimensional contract forms when compared to uni-dimensional contract forms? (4) How are these multi-dimensional contract forms influenced by environmental factors?

The answers to the former two research questions are provided by the uni-dimensional experiment design (section 3.8.2), whereas the latter two research questions are answered through the multi-dimensional experiment design (section 3.8.3). Before going into further detail on these two experimental designs, more information on the datasets is provided in section 3.8.1.

3.8.1 Datasets

In accordance with the methodology which was introduced when modelling the problem, three separate datasets have been created. Each of these datasets corresponds to one of the components used to model the problem: the contract, the trade-off and the evaluation model (see section 3.2). All these datasets are freely available and can be downloaded from the following website:

`www.projectmanagement.ugent.be/?q=research/contracting`.

3.8.1.1 Contract Models

The goal of the contract model dataset is to present a comprehensive cross section of contracting techniques used in practice. By testing this cross section of contract structures on a wide range of possible environments, the effectiveness of these techniques can be determined. As was mentioned in section 3.7, the contract dataset consists of several contract (sub)sets, grouping contracts belonging to a specific type. An overview of this is given in table 3.2.

Contract types discussed in the literature are usually defined based on which party is accountable in case of extremely positive or negative outcomes (see section 2.2). This

Base type	Accountability Downside	Accountability Upside	Code	# Contracts tested Cost	# Contracts tested Duration	# Contracts tested Scope	In Literature
Linear	Contractor	Contractor	LCC	7	168	9	Firm Fixed Price
	Contractor	Shared	LCS	56	1,344	72	Guaranteed Maximum Price
	Shared	Shared	LSS	63	1,512	63	Fixed Price Incentive
	Owner	Shared	LOS	56	1,344	72	Cost Plus Incentive Fee
	Owner	Owner	LOO	7	168	9	Cost Plus Fixed Fee
3-Piecewise	Contractor	Contractor	3PCC	336	8,064	432	-
	Contractor	Shared	3PCS	1,176	36,288	3,888	-
	Owner	Shared	3POS	1,512	36,288	1,944	-
	Owner	Owner	3POO	336	8,064	432	-
4-Piecewise	Contractor	Contractor	4PCC	980	0	1,260	-
	Contractor	Shared	4PCS	2,940	0	2,520	-
	Owner	Shared	4POS	4,200	0	3,780	-
	Owner	Owner	4POO	1,260	0	1,260	-
non-linear	Shared	Shared	N	28,672	1,102,248	59,049	-
			TOTAL	41,601	1,195,488	74,790	

Table 3.2: Contract dataset

allocation of accountability for the different contract types is shown by the accountability columns in table 3.2. Accountability can either be shared or allocated to the contractor/owner. Allocating the downside accountability to the contractor means that all overruns beyond a certain point will be paid in full by the contractor. Where this specific point lies depends on the other parameters set in the contract. Contrarily, when the upside risk is allocated to the contractor, (s)he can fully benefit from all the gains of the performance (e.g. cost savings) beyond a certain point. Alternatively, the risk can also be shared between the two parties according to a certain ratio. This accountability in the extrema is translated into a shortened contract code which will be used from now on to identify the different contract types (see the *code* column in table 3.2). The final column of table 3.2 also mentions the nomenclature used in existing literature for these specific types of contracts, insofar as this nomenclature exists.

Practically, allocating the accountability to the owner or contractor is equivalent to setting the sharing ratio for the associated region to 0 or 1 respectively. For the simple linear contracts, this means that the sharing ratio above and/or below the target is set to 0 or 1. For three- and four-piecewise linear contracts, the upside and downside accountability refer to the first and final segment.

For each of these types, the parameters which are not fixed due to the nature of the contract type are varied, resulting in a varying number of available instances per contract type, as listed in table 3.2. These parameters include the respective targets (C^t, D^t, S^t), the bounds for the various regions (B_r^C, B_r^D, B_r^S), the sharing ratios (s_r^C), the region valuation parameters (v_r^D, v_r^S), the non-linear incentive amounts and bounds ($I^C_{\max}$, $I^D_{\max}$, $I^S_{\max}$, $I^C_{\min}$, $I^D_{\min}$, $I^S_{\min}$, UB^C, UB^D, UB^S, LB^C, LB^D, LB^S), and lump sum duration incentive targets and incentive amounts (D^t_{ls}, I^D_{ls}). Each of these parameters is varied across the complete relevant range, with a step size equal to $\frac{1}{8}$ of the total size of the range, including or excluding the bounds of the range as is relevant for practical implementation. This results

in a total of 41,601 cost contract options, 1,195,488 duration contract options, and 74,790 scope contract options. Note that the number of duration contract options is substantially greater due to the possibility of including lump-sum incentives related to a specific deadline. Readers who desire more information on these datasets are referred to the on-line resources accompanying this research.

3.8.1.2 Trade-off Models

In order to create instances of the trade-off model, the independent parameters (D, S, E) are spread across a range of size 1: $D \in [1, 2]$, $S \in [1, 2]$ and $E \in [0, 1]$ (i.e. The shortest possible duration for the project is defined as 1, and the longest possible duration for the project is defined as 2). For each of these dimensions, 11 equidistant discrete options were created within this range ($n = 10$, see section 3.4). This results in a a total of 1,331 $(=(n+1)^3)$ trade-off points for each instance. The actual difference between these instances is represented by the variation in the costs associated with each of these trade-off points, which depends on the parameter settings used for the specific instance. These settings are summarised in the first part of table 3.3.

The generation procedure for the trade-off problems requires a value for each of the parameters listed in the first part of table 3.3. The generation procedure starts by simply selecting the values for the three first parameter types: the slopes ($|S_D|$, $|S_S|$, $|S_E|$), the multiplicators (m_D^S, ..., m_E^S), and the minimal cost value ($C_{\min}$). Given these parameter values, a linear approximation of the relation between each of the independent dimensions (duration, scope and effort) can be made, for every possible value of the other two independent dimensions (e.g. the relation between duration and cost given a certain scope and effort level). These linear approximations are then used in combination with the value for the convexity magnitude metrics (CM_D, CM_S, CM_E) to determine a convex curve which represents the impact of each separate dimension on the cost. Practically, this convex curve is created using a linear programming (LP) model which uses the linear approximation as well as the relevant CM metric as an input. The workings of this model are detailed in 3.C. This means that a total of $3 \cdot (n+1)^2$ LP models are solved in order to model the impact of the individual dimensions. The implementation of this model was done using Gurobi 5.6 optimisation software, which solved each of the instances in under a minute on a 2.5 GHz Intel Core i5 machine. The result for the individual dimensions is then combined by summing the cost increases (ΔC) for the separate dimensions, resulting in $(n+1)^3$ costs (C_{ijk}) corresponding to every possible duration-scope-effort combination. Using this methodology and the parameters listed in table 3.3, a total of 380 trade-off models have been created.

Model	Parameter	Symbol	Values	# Instances	Total
Trade-Off	Basic slopes	$\lvert S_D\rvert$, $\lvert S_S\rvert$, $\lvert S_E\rvert$	$\{0.5, 0.75, 1, 1.25, 1.5\}$	125	380
	Multiplicators	m_D^S, ..., m_E^S	$\{0, 0.1, 0.2, 0.3, 0.4\}$	125	
	Minimal cost	$C_{\min} = C_{000}$	$\Delta C_{nnn} \cdot \left\{\frac{1}{0.10}, \frac{1}{0.20}, \frac{1}{0.30}, \frac{1}{0.40}, \frac{1}{0.50}\right\}$	5	
	Convexity	CM_D, CM_S, CM_E	$\{0, 0.0625, 0.125, 0.1875, 0.25\}$	125	
Evaluation	Max Effort cost	E_m^{cost}	$\Delta C_{nnn} \cdot \{0.25, 0.375, 0.5, 0.625, 0.75\}$	5	25
	Effort ROI	ROI^E	$\{0, 0.05, 0.1, 0.15, 0.2\}$	5	
	Time value	μ_D	$\Delta C_{nnn} \cdot \{0.5, 0.75, 1, 1.25, 1.5\}$	5	
	Deadline value	μ_{ls}	$\Delta C_{nnn} \cdot \{0, 0.05, 0.1, 0.15, 0.2\}$	5	
	Scope valuation	μ_S	$\Delta C_{nnn} \cdot \{0.5, 0.75, 1, 1.25, 1.5\}$	5	

Table 3.3: Environment dataset

3.8.1.3 Project Evaluation Models

The generation of evaluation model instances requires the selection of the parameters which are shown in the bottom half of table 3.3. The majority of these parameters are expressed relative to the total cost spread of the trade-off problem used, expressed as the difference between the lowest attainable cost (C_{000}) and the highest possible cost (C_{nnn}): $\Delta C_{nnn} = C_{nnn} - C_{000}$. This is done to ensure a consistent variation of the different parameters over different trade-off models. As shown in table 3.3, a total of 25 different evaluation model instances have been created, allowing for a total of 9,500 ($25 \cdot 380$) different environments on which contract performance can be tested.

3.8.2 Uni-Dimensional Experiments

The aim of the uni-dimensional experiment is to investigate the relative performance of different contract types for the different dimensions, as well as the impact of environmental factors on the performance of these contracts. To do this, the 25 evaluation model instances are combined with the 380 trade-off model instances (see table 3.3), resulting in 9,500 problem environments on which the different contract type instances (see table 3.2) can be tested. The nature of this experiment is uni-dimensional, which signifies that only one type of incentive is tested per environment (i.e. an environment is never subjected to both a cost and scope incentive simultaneously).

Due to the difference in the number of parameter settings between the contracts, the time needed to evaluate a single contract differs depending on the contract type. On a 2.26 GHz processor the evaluation of a single environment (i.e. a combination of a trade-off and an evaluation instance) takes 27, 793 and 135 seconds on average for the cost, duration and scope contract sets respectively. Hence, the complete experiment takes approximately 2,500 single-core computation hours.

For every environment, the performance of the different contract types is judged by creating a number of efficient frontiers containing only the contracts of a certain types. These *local* efficient frontiers are then compared to the *global* efficient frontier, which

includes contracts of all available types. The comparison itself is made by calculating the averaged Hausdorff distance (Δ_p) between the local and global frontiers as well as the spread of the local frontiers.

This process is repeated for different types of sensible contract design strategies (see section 3.5.3). Whenever a specific contract does not satisfy the criteria for sensible contract design, it is no longer included in any of the local or global efficient frontiers. Four different sensible contract design strategies have been used. For the first type no restrictions whatsoever are included, all contracts are considered in the efficient frontiers. The second type imposes the restriction that the maximal incentive which can be earned by the contractor cannot be negative. The third type requires that the maximal incentive which can be earned by the contractor is at least 2% of the average project cost[3]. The fourth criterion extends the third by also requiring that the range between the highest and lowest incentive amounts which can be earned is at least 4% of the average project cost. These constraints are similar to the values observed in practice by other authors (Abu-Hijleh and Ibbs, 1989; Tang et al., 2008; Veld and Peeters, 1989), see section 3.5.3.

The results of the experiments will be presented using visual representations rather than raw data tables to facilitate interpretation. Readers who would like the raw data behind these graphs can download them on-line from the following website:
`www.projectmanagement.ugent.be/?q=research/contracting`.

[3]measured as the average over all possible outcomes of the project.

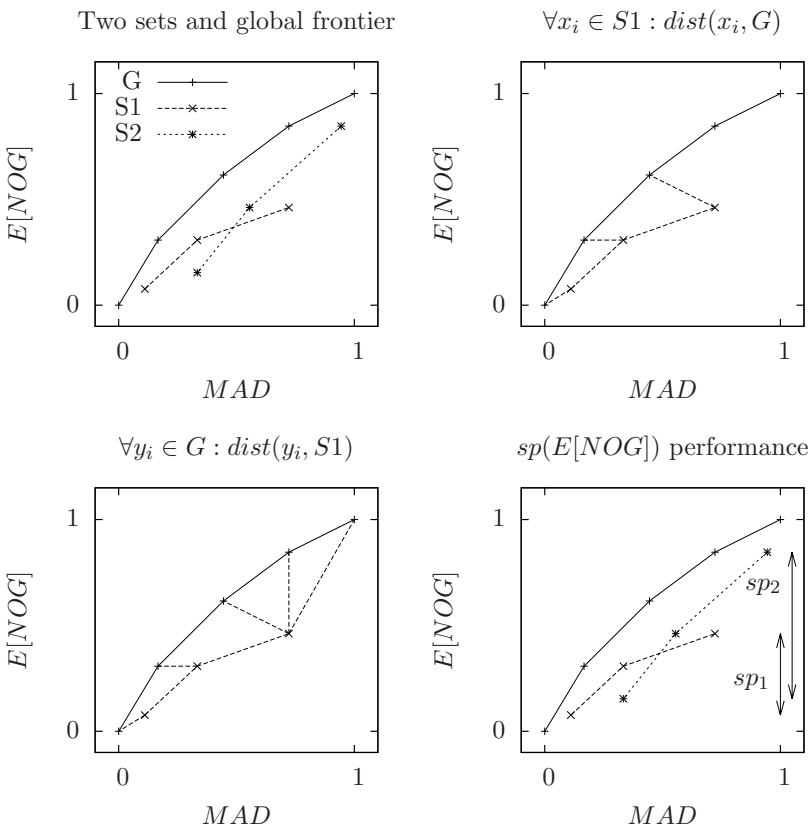


Figure 3.8: Graphical illustration of the metrics used to measure the performance of certain contract sets. The top left panel shows the Pareto frontier of two contract sets $S1$ and $S2$, as well as the global Pareto frontier G, which is constructed using all the best known contracts regardless of their type. The top right and bottom left panels illustrate components of the averaged Hausdorff distance (Δ_p) metric used to measure the distance to the global front. The bottom right panel illustrates the measurement of the $E[NOG]$ spread associated with the respective sets.

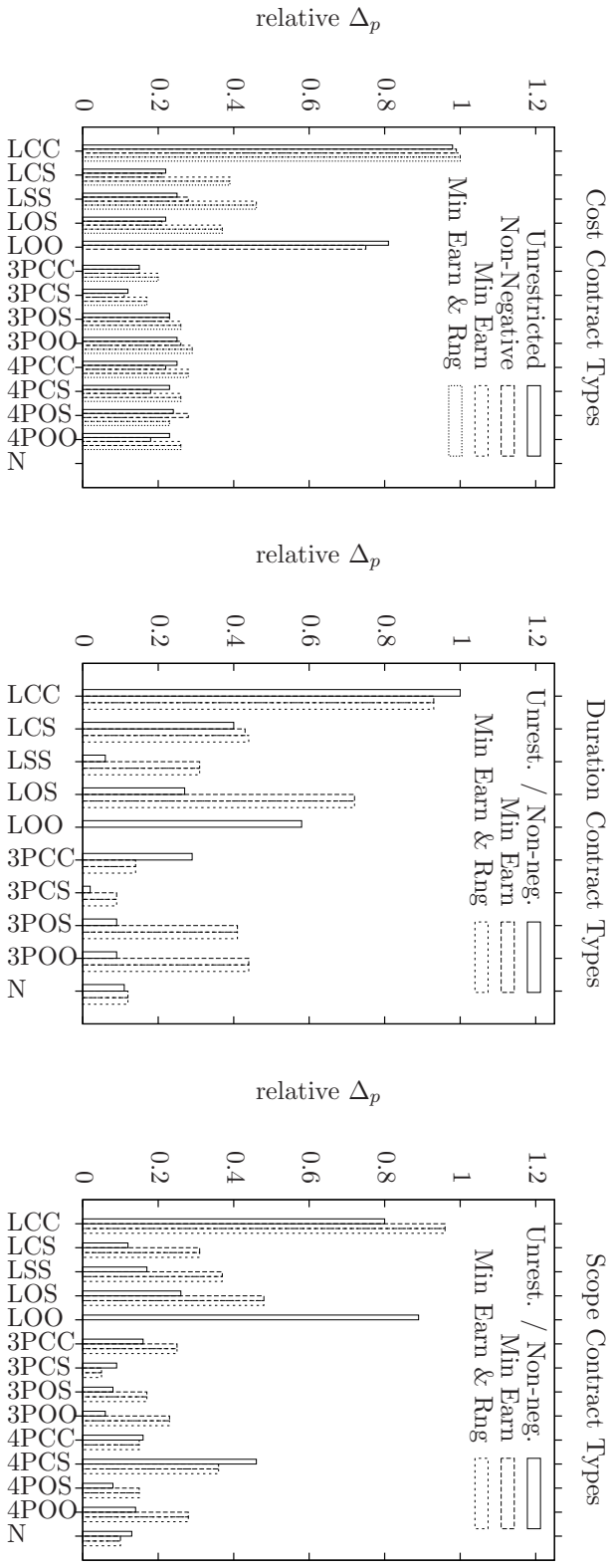


Figure 3.9: Distance to the Pareto front for uni-dimensional experiments.

3.8.2.1 Performance of Contract Types

The first step in analysing the results of the uni-dimensional experiment is measuring the (relative) performance of the contract types. Figure 3.9 visualises the the averaged Hausdorff distance Δ_p performance measure, which represents the proximity to the global efficient frontier. To facilitate comparing results the averaged Hausdorff distance was standardised within the range $[0, 1]$. The second performance measure - the spread of the solutions for the different evaluation dimensions - is shown in figure 3.10. Naturally these results were also standardised within the range $[0, 1]$, based on the best and worst values for each specific environment.

For the majority of contract types the fraction of contracts which did not satisfy imposed restrictions for sensible contract design (see section 3.5.3) remained limited: in less than 1.5% of cases violations were observed. Only the *LOO* contract, which transfers all risk to the owner, was unable to satisfy any of the imposed constraints, as can be seen from the results in figures 3.9 and 3.10

Non-linear and piecewise linear contracts outperform linear types: *Non-linear and piecewise linear contract types generally outperform linear contract types in terms of proximity to the efficient frontier, whilst showing average to good performance in terms of spread for all metrics.* Figure 3.9 shows that the best performance in terms of proximity to the efficient frontier (Δ_p) is always observed to be either a non-linear (N) or piecewise linear contract type. It can also be noted that the best performing piecewise linear contracts always share the upside potential, rather than allocating it completely to either party. The spread of these best performing types is also consistently good, as shown by figure 3.10.

Avoid linear contracts which transfer all risk to a single party: *Linear contracts which transfer all the risk to either the contractor*[4] *(LCC) or the owner*[5] *(LOO) perform worse in terms both proximity and spread when compared to other linear contract types.* Moreover, due to the complete absence of value transfer from the owner to the contractor when using the *LOO* contract type, the minimum incentive and range conditions for sensible contract design cannot be satisfied. The relevance of these observations is validated by the continued extensive use of this type of contract, as can be seen from table 2.1.

Preferred types are robust to changing restrictions: *Adding restrictions to enforce sensible contract design does not change the preferred contract type for any of the contract dimensions.* For other contract types imposing additional constraints can have a significant impact on the relative performance. Simple linear contracts seem particularly

[4]Often dubbed a Firm Fixed Price (FFP) contract in traditional literature.
[5]Often described as a cost plus fixed fee ($CPFF$) contract in contracting literature

sensitive to the imposition of constraints for sensible contract design.

Skewness persists within contract types: *The alignment skewness as measured by the MD persists within a contract type, as indicated by the consistently low values for the standardised spread for this metric ($spr(MD)$).* Figure 3.10 shows that the average spread for the MD metric is low in both absolute and relative terms for all types of contracts. Hence, it is likely that other contract types will have to be considered in order to remove an undesired skew from a contract.

Based on these observations, project owners are advised to use piecewise linear or non-linear contracts, rather than the traditional linear contract archetypes. Especially linear contracts which transfer all risk to the contractor or the owner, such as the traditional firm fixed price (FFP) or cost plus fixed fee ($CPFF$) contract types, should be avoided. Moreover, when an undesirable skew of the alignment (as measured by the MD indicator) is observed, it may be beneficial to include other contract types in the analysis.

Figure 3.10: The average spread obtained by different contract types for the cost, duration and scope dimensions. These are the outcomes for the situation where both a minimal earning and a minimal range for the contractor is required.

3.8.2.2 Impact of Environmental Factors on Contract Type Performance

The properties of the environment in which the contract operates can also have an important influence on its effectiveness. Hence, the experiment design uses a large set of parameter settings for both trade-off and evaluation models. The impact of these environmental factors on the average Hausdorff distance Δ_p is measured by the slope of the linear regression line which estimates Δ_p as a function of the respective environment parameter value: $\Delta_p \sim$ parameter).

The key considerations with respect to the influence of environment parameters are the following: Firstly, which of the environmental parameters has the largest average impact on contract performance? And secondly, which contract types are influenced most by changes in the environment in which they operate (i.e. the robustness of different contract types)? Note that impact in this case can mean either a significantly positive or a significantly negative slope of the linear regression line.

Figure 3.11 provides an answer to the former question by investigating the impact of different environment parameters. An answer to the latter question is provided by figure 3.12 which plots the average absolute value of the scope coefficients for the various contract types.

Strong variation of impact across environment parameters: *A comparison of the impact of different environment parameters shows that there is a clear distinction between parameters with a high and low impact on contract type performance.* These high-impact environment parameters can be identified in figure 3.11 by their large positive or negative deviation from zero.

The results show that in situations where increasing the effort has a larger impact on the other trade-offs (i.e. a high value for the m_D^E or m_S^E multiplicators), the proximity to the efficient frontier worsens for all dimensions. Contrarily, an increase in the convexity of the relations between the cost and any of its drivers (duration, scope and effort) has a generally positive influence on the proximity of contract sets of different types to the efficient frontier, again for all dimensions.

The impact of environmental parameters on different contract dimensions (cost, duration and scope) can also be substantially different. An example of this are the convexity parameters (CM_D, CM_S, CM_E), which have a substantially larger impact on cost incentive contracts. Moreover, several other parameters show opposite influences on contract types of different dimensions. An example of this is the cost of effort (E_m^{cost}): an increase of this parameter has a negative impact on cost incentive performance, but a positive impact on the performance of scope contracts.

After analysing which environment parameters have the largest influence on contract types in general, the inverse can also be tested: which specific contract types are most

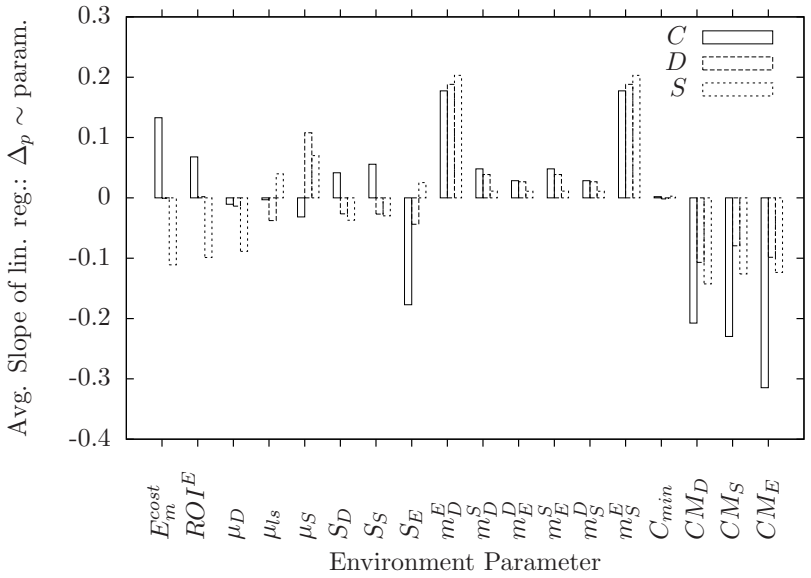


Figure 3.11: The average impact of variation of environment parameters on contract performance, measured by the slope coefficient of the linear regression. These are the outcomes for the situation where both a minimal earning and a minimal range for the contractor's earnings is required.

susceptible to changes in environment parameters in general? This impact is expressed in relative terms for the various contract types in figure 3.12. The vertical axis on these graphs represents the absolute value of the regression slopes as they were used in figure 3.11.

Duration and scope contract types are more volatile when faced with changes in the environment: *Variations in the environment in which a contract type is operating have a greater average impact on duration and scope contract types than on cost contract types.* Moreover, the relative impact of changes in the environment differs little between different cost contract types, whereas more significant fluctuations can be observed between duration and scope contract types.

Preferred types are also robust to changing environments: *The preferred contract types (see section 3.8.2.1) in general, and the non-linear contract types in particular show a remarkable robustness to changes in the environment.* Hence, the superior performance of these types is likely to be valid across most types of projects. Contrarily, several contract types show an above average volatility when compared to the other contract types. Especially the LCS and $4PCS$ contract types for the duration and scope dimension respectively, seem to be greatly influenced by the nature of the environment. Other contract types which show above average volatility in the scope dimension include

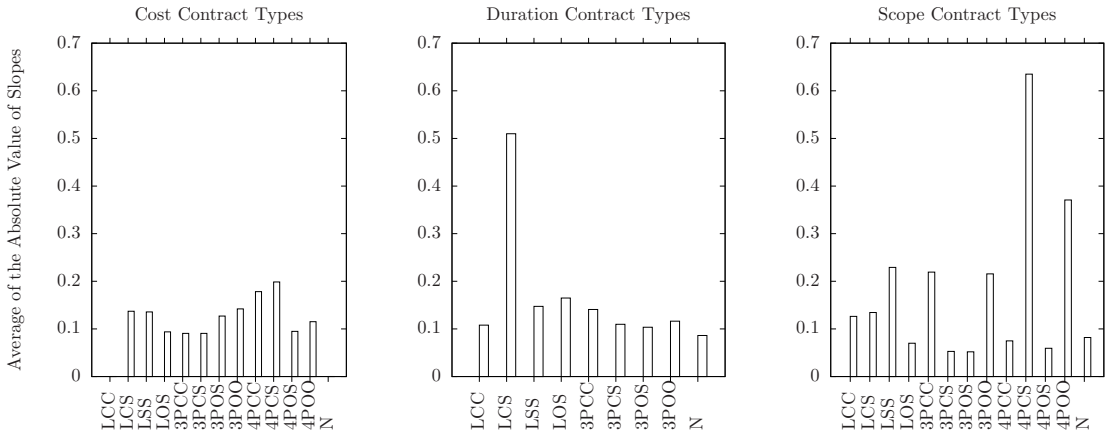


Figure 3.12: The average impact of variations in the environment for the different contract types. These are the outcomes for the situation where both a minimal earning and a minimal range for the contractor's earnings is required.

the LSS, $3PPC$ and $4POO$ types.

Based on these observations, project owners are advised to pay extra attention to the nature of the environment in which they are operating when designing contracts for duration or scope dimensions. Even more so when they opt for contract types different from the ones which were identified as most effective in section 3.8.2.1. Naturally, the environment parameters which were identified as being most critical should have priority over the other parameters when analysing the impact of the environment on contract type performance.

3.8.3 Multi-Dimensional Experiments

The goal of the multi-dimensional experiments is to determine how the performance of multi-dimensional contracts (i.e. contracts which combine cost, duration and scope incentives) compares to that of uni-dimensional contracts. Several authors have highlighted the risk of contracts which focus on a single incentive dimension (Abu-Hijleh and Ibbs, 1989; Broome and Perry, 2002; Meng and Gallagher, 2012), yet no extensive computational experiments have been carried out to test these hypotheses.

Due to the exponential increase in computational complexity when testing all possible contract combinations[6], these experiments were carried out using only a subset of the datasets which were employed for the uni-dimensional experiments (see section 3.8.2). The contract set used for this experiment is limited to the linear contract types which are listed

[6]Testing all possible combinations would take approximately 90,000 years on a single core.

in table 3.2, resulting in a total of 189 cost contracts, 4,536 duration contracts, and 225 scope contracts. This means that a total of 193 million ($189 \cdot 4,536 \cdot 225$) multi-dimensional contract combinations are tested for each environment.

The time needed to evaluate the performance of all these combinations for a single environment is approximately 40 hours using a 2.26 GHz processor. Because of this, the number of environments on which the tests are carried out is also a subset of the environments used in the uni-dimensional experiment. For both the trade-off and evaluation datasets, one instance was initialised using the average values for all parameters, plus for each parameter of the data set a high and low value instance with all other parameters still at the average value was initialised. The values used are simply the average, maximal and minimal values which were mentioned in table 3.3. The number of environments was further reduced by always combining the trade-off and evaluation models with the default instance of the evaluation and trade-off model respectively. This results in a total of 38 different project environments, based on 27 trade-off models and 11 evaluation models. Hence, a total of approximately 1,520 single-core computation hours are needed to carry out this experiment.

The performance of contracts is evaluated by categorising them into different types, and then comparing the Pareto front formed by contracts of a certain type to the global Pareto front, which is constructed by taking all possible contracts into consideration. Similar to the uni-dimensional experiments, this is done for four different strategies regarding sensible contract design: the unrestricted case, the non-negative case, the minimum earnings case and the minimal earnings and range case (see section 3.8.2). The results of this experiment will first be analysed in the context of the relative performance of the different types of contracts in section 3.8.3.1. Next, the impact of variations in the environment is discussed in section 3.8.3.2.

3.8.3.1 Performance of Contract Types

The relative performance of different combinations of contract types is visualised in figure 3.13. To analyse the performance of these multidimensional contracts, an number of new contract types are defined, as can be seen from the horizontal axes on the figure. These contract types can be interpreted as follows: A first set of contract types focuses on the number of dimensions used in a contract, as well as the nature of these dimensions. The n-dim nomenclature simply indicates that the contract set contains incentives affecting n contract dimensions out of a maximum of three (cost C, duration D and scope S). When this name is followed by dimension names, this indicates that the contracts belonging to this type have incentive clauses exclusively for these specific types. For example the 2-dim, C&S contract type includes contracts which have both a cost and scope incentive, but no

duration incentive clause.

The second set of contract types refers to the inclusion of the simple contract types as listed in table 3.3 into a multi-dimensional contract. The nomenclature is similar to the one used in section 3.8.2. For example, the $\exists C - LCC$ contract type represents all contracts containing an LCC contract for the cost dimension, regardless of any other incentives which may or may not be included. This enables a comparison of the average effectiveness of different types of contract clauses.

More dimensions yields better performance: *Increasing the number of dimensions of a contract has a positive effect on all metrics for contract set performance.* Looking at the performance of the contract types which only specify the number of dimensions, it is clear that the performance both in terms of proximity and spread increases substantially when moving from a 1-dim contract to a 2-dim contract, and from a 2-dim contract to a 3-dim contract (see figure 3.13). Similarly, when looking at the types which also specify certain contract dimensions to be used, an improvement can always be seen when adding a contract dimension (e.g. moving from a "1-dim, C" contract to a "2-dim, C&S" contract).

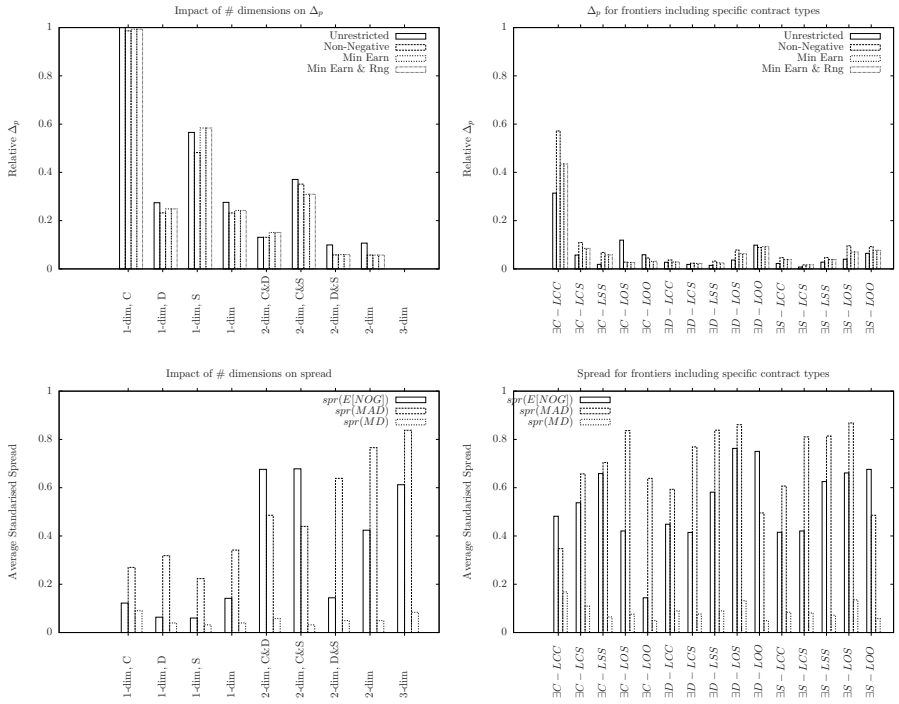


Figure 3.13: Average distance to the global Pareto front for multi-dimensional contract types (top panels), and the average spread obtained using different contract types for the situation where both minimal earnings and range of the contractor's earnings is required (bottom panels).

Duration > Scope > Cost: *In case a 3-dimensional contract cannot be implemented, the owner should opt to incentivise the duration and scope dimensions in a bi-dimensional set-up or the duration dimension in a uni-dimensional set-up, preferring the former over the latter.* The top left panel of of figure 3.13 indicates that the performance in terms of proximity to the global frontier of a one and two-dimensional contract is strongly dependent on the chosen incentive dimensions. Specifically, the owner should prefer to attach incentives to the duration and scope dimensions - in that order. If the right incentive dimensions are chosen, the average performance gap between contracts including more or less incentives can be greatly reduced, although it still remains present.

Avoid contract components which transfer all risk to a single party: *Multi-dimensional contracts which include components which transfer all risk to one of the two parties (LCC or LOO) result in suboptimal performance.* The increased distance to the global frontier when using such contracts (see the right panels of figure 3.13) corroborates the undesirable behaviour of these contract types which was observed in the uni-dimensional experiments.

3.8.3.2 Impact of Environmental Factors on Performance

Figure 3.14 shows how variations in the environment impact the performance of the tested multidimensional contracts. This performance is measured as the slope of the linear regression line which estimates the Δ_p as a function of the respective environment parameter.

Environmental robustness increases with # dimensions: *The robustness to changes in the environment of contract types increases as the number of incentivised dimensions in the contract increases* (see the left panel of figure 3.14). This is true for all but one case: when moving from a one dimensional cost contract (1-dim, C) to a two-dimensional cost and scope contract (2-dim, C&S), there is a slight increase in the average sensitivity to changes in the environment. This is due to the fact that incentivising the scope appears to be highly sensitive to changes in the problem environment, as was already observed in section 3.8.2.2.

Preferred multidimensional types are relatively robust to environmental changes: *The most desirable multidimensional contract types for each dimension identified in section 3.8.3.1 are not heavily influenced by changes in the environment.* This confirms that these managerial guidelines are valid regardless of the type of project the owner is facing. Figure 3.14 also shows that unidimensional scope contracts are relatively unstable with respect to changes in the environment parameters. Project owners should therefore be cautious and verify the adequacy of such contracts in the specific environment where they will be implemented.

Including certain components increases the environmental sensitivity: *The*

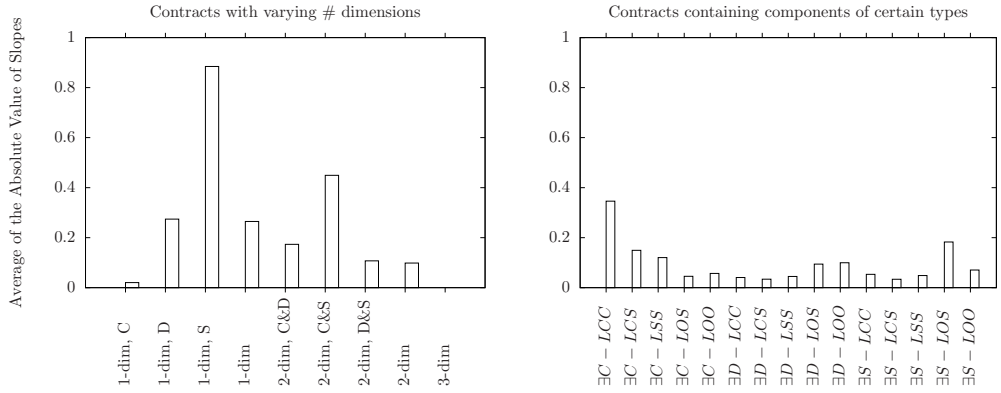


Figure 3.14: The average impact of variations in the environments for contracts of different compositions. These are the outcomes for the situation where minimal earning and range contract design rules are enforced.

right pane of figure 3.14 reveals that using certain components in multi-dimensional contracts can negatively influence the robustness of the contract type. The greatest peak can be observed for the $\exists C - LCC$ contract type, which indicates that a traditional firm fixed price contract is used for the cost dimension. Again confirming preceding observations of the risks involved when implementing this type of contract.

Based on these observations, a contract owner wishing to construct a contract structure which is robust to changes in the environment in which it operates is advised to use as many dimensions as possible in the contract. In case it is impossible to construct a three-dimensional contract, the combination of scope and duration is on average most robust for a two-dimensional contract. The best uni-dimensional alternative from a robustness perspective is the introduction of a cost incentive, however given that a uni-dimensional duration contract is only slightly more sensitive and yields better performance on average (see section 3.8.3.1), using uni-dimensional duration contracts is preferable.

3.9 Conclusions

This chapter discussed incentive contract design for projects from the perspective of the project owner. Based on axioms derived from an extensive literature review, a model consisting of three subcomponents has been constructed. The model serves as a framework which can be used by project owners to improve the design of incentivised contracts. This model was also used to conduct computational experiments, from which a set of managerial contract design guidelines have been derived (see summary in table 3.4).

The results of these experiments indicate that multidimensional contracts with piece-

	Experiment	
	Uni-Dimensional	**Multi-Dimensional**
Contract type	Non-linear and Piecewise contract outperform linear contracts. Avoid transfer of all risk to one party. Preferred types are robust to imposed restrictions. Skewness persists within contract types.	More dimensions improve performance. Duration > Scope > Cost. Avoid contract components transferring risk to a single party.
Environment	Strong variation of impact across environment parameters. Duration and scope contracts are more volatile. Preferred types are more robust.	More incentivised dimensions improves robustness. Preferred multidimensional types are more robust. Including certain components increases sensitivity.

Table 3.4: Summary of experimental results

wise linear and/or non-linear incentive clauses are superior on all performance metrics. Moreover, good performing contract structures are generally more robust to changes in the environment in which they operate, and can therefore be applied in structurally different project environments.

Linear contracts which transfer all the risk to one of the two parties, such as the traditional firm fixed price (FFP) and cost plus fixed fee (CPFF) contracts should be avoided in both uni-dimensional and multi-dimensional incentive contracts. These contracts produce inferior results in terms of both solution quality and diversity, whilst also being less robust to changes in the environment in which they operate, when compared to other contract alternatives.

Furthermore, when incentivising all dimensions is impossible from a practical point of view, the owner's first aim should be to implement a duration incentive, then a scope incentive in that order. This opposes the traditional focus on cost incentives in both literature and practice.

Several extensions of the problem can be considered relevant for future research. A first extension would be to implement more advanced optimisation techniques to optimise the contract parameters, rather than the full factorial approach taken in this study. Secondly, the realism of the model could also be improved by lifting the assumption of full information as well as the deterministic nature of the trade-offs. These conditions could be tested within a game-theoretical setting which analyses the optimal behaviour of both owner and contractor in both single and repeated games. A final area for future research would be the implementation of this model in real life case studies.

3.A Overview of Notation

Table 3.5 presents an overview of all the notation introduced in this chapter.

Model	Variable	Description
Contract Model	C, D, S	The observed cost, duration and scope at the end of the project.
	I^C, I^D, I^S	Cost, duration and scope incentive awarded.
	C^t, D^t, S^t	Targets for cost, duration and scope as specified in the contract.
	s^C_r	Sharing ratio for (piecewise) linear cost incentive contract in region r.
	v^D_r, v^S_r	Region valuation parameter for piecewise linear duration/scope incentive.
	B^C_r, B^D_r, B^S_r	The lower bound of cost, duration or scope region r in a (piecewise) linear contract.
	UB^C, UB^D, UB^S	Upper bound of the region over which a cost, duration or scope incentive is spread using a non-linear contract.
	LB^C, LB^D, LB^S	Lower bound of the region over which a cost, duration or scope incentive is spread using a non-linear contract.
	$I^C_{\max}, I^D_{\max}, I^S_{\max}$	Greatest incentive amount which can be awarded in a non-linear cost, duration or scope contract.
	$I^C_{\min}, I^D_{\min}, I^S_{\min}$	Greatest disincentive amount which can be awarded in a non-linear cost, duration or scope contract.
	D^t_{ls}	Target date associated with a lump-sum duration incentive amount.
	I^D_{ls}	Incentive amount for a lump-sum duration incentive clause.
Trade-Off	D_i, S_j, E_k	The attainable values for duration, scope and effort as defined by a trade-off model.
	$\Delta D_i, \Delta S_j, \Delta E_k$	Attainable trade-off points for duration, scope and effort, defined as a distance to their lowest cost option.
	$\Delta C^D_i, \Delta C^S_j, \Delta C^E_k$	The impact on the cost of selecting duration, scope and effort mode i, j and k respectively.
	$C_{\min}$	The lowest attainable cost for the project.
	S_D, S_S, S_E	The slope of the relationship between the cost (dependent) and duration, scope and effort (independents).
	m^b_a	Slope multiplicator, max impact of dimension b on the slope of the relationship between dimension a and the cost.
	CM^D, CM^S, CM^E	The convexity magnitude for the relation between C (dependent) and D, S and E (independent) respectively.
Contract Evaluation	$E[NOG]$	Expected Net Owner Gain
	E^{cost}_l	Cost of effort
	I^{tot}_l	Total incentive awarded in scenario l
	NCG_l	Net contractor gain for scenario l
	ROI^E	Average ROI for effort investments by the contractor.
	NCG^{adj}_l	Net contractor gain for scenario l taking opportunity cost into account
	NOG_l	The value the owner derives from scenario l
	C_l, D_l, S_l, E_l	The cost, duration, scope an effort associated with scenario l.
	μ_D, μ_{ls}, μ_S	The monetary valuation of time, deadline and scope of the owner
	$RNCG^{adj}_l$	Relative net contractor gain: NCG^{adj}_l rescaled to $[0, 1]$
	$RNOG_l$	Relative net owner gain: NOG_l rescaled to $[0, 1]$
	MD	Mean deviation, a measure for the pay-off alignment
	MAD	Mean absolute deviation, a measure for the pay-off alignment
	$I^{tot}_{\max}$	The maximal attainable incentive amount for the contractor
	$LB_{I^{tot}_{\max}}$	A lower bound for the maximal incentive amount which can be earned by the contractor
	$R_{I^{tot}}$	The range of the incentive earnings by the contractor
	DL	External deadline relevant to the project owner.

Table 3.5: Overview of model parameters

3.B Derivation of the CM Metric

The basic definition of the CM metric is as follows:

$$CM = \frac{A}{A + B} \tag{3.32}$$

The variable A can be eliminated by calculating the surface of the triangle as follows:

$$\begin{aligned} A + B &= \frac{\Delta D_n \cdot \Delta C_n}{2} \\ A &= \frac{\Delta D_n \cdot \Delta C_n}{2} - B \end{aligned} \tag{3.33}$$

And rewriting CM as:

$$\begin{aligned} CM &= \frac{A}{A+B} \\ &= \frac{\frac{\Delta D_n \cdot \Delta C_n}{2} - B}{\frac{\Delta D_n \cdot \Delta C_n}{2} - B + B} \\ &= 1 - \frac{2 \cdot B}{\Delta D_n \cdot \Delta C_n} \end{aligned} \tag{3.34}$$

The value of B can be calculated as:

$$\begin{aligned} B = \sum_{i=1}^{n} [(\Delta D_i - \Delta D_{i-1})\Delta C_{i-1} + \\ \frac{1}{2}(\Delta D_i - \Delta D_{i-1})(\Delta C_i - \Delta C_{i-1})\Big] \end{aligned} \tag{3.35}$$

In case of generated convex curves it can be assumed that the step size of the duration are always identical and therefore all equal to $\frac{\Delta D_n}{n}$. This means the formula can be simplified to:

$$B = \frac{\Delta D_n}{n} \sum_{i=1}^{n} \left[\Delta C_{i-1} + \frac{\Delta C_i - \Delta C_{i-1}}{2}\right] \tag{3.36}$$

3.C Linear Model for Trade-off Model Generation

The objective of this model is to create the smoothest possible convex curve which starts and ends at the start and end points of the linear approximation curve respectively. The model uses the variable $\Delta\Delta C_i$ to signify the change in cost between two consecutive points on the trade off curve: $\Delta C_i = \Delta C_{i-1} + \Delta\Delta C_i$. The graphical interpretation of these variables is illustrated in the right panel of figure 3.5. Hence, this variable is indexed $i = 1, ..., n$. Since the curve is non-decreasing and convex it is known that each of these variables is positive (see equation 3.38) and at least as large as the preceding variable (see equation 3.39).

The smoothness of the curve can be measured by the difference between two consecutive $\Delta\Delta C_i$ variables, which indicate the change in cost caused by a change of an independent variable (duration, scope or effort). To avoid having linear segments where the scope is not increased, the minimal difference between these $\Delta\Delta C_i$ variables is maximised. Doing this guarantees that the continual slope increase over the consecutive discrete steps is as large as possible. This minimal difference is represented by the variable $\Delta\Delta\Delta C_{\min}$ in the

model. The input parameters required for the linear model are the greatest potential cost increase (ΔC_n), as can be determined by the linear approximation of the curve, and the desired convexity of the curve (CM).

$$\max \Delta\Delta\Delta C_{\min} \tag{3.37}$$

s.t.

$$0 \leq \Delta\Delta C_1 \quad i = 1, ..., n \tag{3.38}$$

$$\Delta\Delta C_{i-1} \leq \Delta\Delta C_i \quad i = 2, ..., n \tag{3.39}$$

$$\sum_{i=1}^{n} \Delta\Delta C_i = \Delta C_n \tag{3.40}$$

$$\Delta\Delta\Delta C_{\min} \leq \Delta\Delta C_1 \tag{3.41}$$

$$\Delta\Delta\Delta C_{\min} \leq \Delta\Delta C_i - \Delta\Delta C_{i-1} \quad i = 2, ..., n \tag{3.42}$$

$$CM = 1 - \frac{2}{n \cdot \Delta C_n} \sum_{i=1}^{n} \left[\sum_{j=1}^{i-1} \Delta\Delta C_j + \frac{1}{2} \Delta\Delta C_i \right] \tag{3.43}$$

Equation 3.38 stipulates that the ΔC_i curve should always be an increasing function (see axioms 1-3). The convexity of the trade-off curve is guaranteed by equation 3.39 which states that the consecutive steps have to be of non-decreasing magnitude. Equation 3.40 ensures that the final point of the convex trade-off curve coincides with the linear approximation on which this curve is based, the latter is defined as the parameter ΔC_n. The value of the smallest difference between the consecutive steps is assigned by equations 3.41 and 3.42. Finally, equation 3.43 guarantees that the resulting curve has the desired convexity magnitude specified by the parameter CM.

4

A Parallel Multi-Objective Scatter Search for Optimising the Owner's Incentive Contract Design

Where chapter 3 used a brute-force enumeration approach to search for an optimal contract structure, the main objective of this research is the design of a meta-heuristic optimisation procedure which can be used by the owner of a project in order to maximise her/his expected profits as well as the degree to which the motivations of both parties are aligned.

4.1 Introduction

A novel technique to optimise the structure of the incentivised contractual agreement between the project owner and the contractor who executes the project is presented in this chapter. The goal of this technique is to optimise the contract from the perspective of the contractor, maximising the expected value of the owner and the alignment of the motivation of both actors, preventing conflicts of interest which can potentially result in suboptimal results for the project owner (Müller and Turner, 2005; Turner, 2004).

Because the delegation of a complete project to a contractor is a highly complex undertaking, an explicit contract which contains all the deliverables as well as the remuneration for the contractor in sufficient detail is impractical (Van Weele and van der Puil, 2013). The best alternative for such a high-complexity environment is an alliance between the two economic actors, effectively forming a single actor for the duration of the project (Rose and Manley, 2011; Walker et al., 2002). However, the fugitive nature which is inherent to a project often causes the substantial investment required by both parties to create such a partnership to be disproportionate to the expected returns of the project. Because of this, contracts that use incentive clauses are often used as a viable middle ground between an explicit contract and a partnership of the two actors (Bower et al., 2002).

Investigating, and by extension optimising, such contract structures requires a systematic view on the relationship between the *owner* of the project, and the *contractor* who is responsible for executing the project. To do this, the framework proposed in chapter 3 is used in this research. This framework consists of three components, the first of which is the *contract model*, which describes the nature of the incentive clauses and targets which are set in a specific agreement. These incentive clauses are linked to the three outcome dimensions which are traditionally used in project management literature: cost, duration and scope (Marques et al., 2011). The second component of this framework is the *trade-off model*, which is used to represent the nature of the project itself as a set of discrete trade-off points. Specifically, these trade-offs strike a balance between the three traditional outcome dimensions, as well as the effort investment made by the contractor who executes the project. The third and final component of the framework is the *evaluation model*, which is used to calculate the value of a specific outcome of the project, given the use of a specific contract. The dynamic of the complete framework considers the owner of the project to be in full control of the contract model, effectively deciding which contract structure will be used to govern the project. Similarly, the contractor controls the choice of a trade-off point, representing her/his control over the manner in which the project is executed. The precise dynamic of this model will be discussed in further detail in section 4.3.

Where chapter 3 used a brute-force enumeration approach to search for the optimal contract structure, the main objective of this research is the design of a meta-heuristic optimisation procedure which can be used by the owner of a project in order to maximise her/his expected profits as well as the degree to which the motivations of both parties are aligned. As such, this research departs from existing literature (see section 4.2) because of its proactive and prescriptive nature, whereas the majority of existing research has focussed on the types of contracts used in practice, as well as their perceived effectiveness.

To achieve this goal, a multiobjective scatter search is designed and tested on a set of problem instances. This type of heuristic is highly suitable to deal with the complex solution space created by the wide range of possible contract structures. The reason for this being the inherent mechanisms of the scatter search procedure which conserve diversity in the solution population. The experiments show that the performance of this heuristic significantly exceeds that of a brute force approach which attempts to scan a broad area of solution space (see section 4.7).

The remainder of this chapter is structured as follows: section 4.2 gives an overview of the existing literature on incentive contracting, as well as multi-objective optimisation techniques. Next, the modelling approach which is used to describe the problem is discussed in greater detail in section 4.3. The dynamic of this model is further illustrated using a small-scale example in section 4.4. Next, the parallel multi-objective scatter search ($MOSS$) is described in sections 4.5 and 4.6. Finally, the computational experiments are discussed in section 4.7 prior to formulating a general conclusion of the research in section 4.8.

4.2 Literature on Multi-Objective Optimisation Methods

The first step in optimising any situation is the formulation of the objective function. Traditionally a single objective is chosen, such as the maximisation of profits or the minimisation of costs. However, the contract design problem studied in this research requires multiple objectives to be optimised. Specifically, the owner is not only interested in maximising his expected profits, but also interested in aligning the objectives of the contractor with his own objectives as much as possible. Hence, in order to present an adequate solution to these (potentially) conflicting objectives, a Pareto frontier containing non-dominated solution is required.

One of the traditional methods of creating such a set of solutions is a weighted-sum approach which scalarizes a set of objectives into a single objective (Burke and Kendall, 2005). By iteratively varying the weights which are allocated to the separate objectives and re-running the solution algorithm a varied set of solutions can be created. Nevertheless,

such techniques are often incapable of guaranteeing an evenly distributed spread across the true efficient frontier (Deb, 2001).

Multi-objective population based meta-heuristics are an alternative solution technique which does not require the aggregation of the different dimensions into a single objective. These techniques are capable of searching in multiple dimensions simultaneously. Research interest in these techniques has steadily increased during the last two decades (Zhou et al., 2011). Moreover, these solution techniques have proven their effectiveness within a project management context (Meier et al., 2016). However, the complex nature of the contract design problem, which requires both the nature of the contract as well as the contract's parameters to be determined, decreases the effectiveness of traditional evolutionary algorithms which rely on relatively simple chromosome encodings.

The scatter search meta-heuristic has been proposed as a technique which uses a similar population-based approach but uses a more systemic way to search the solution space, increasing its effectiveness for complex problems (Glover et al., 2000). Scatter search algorithms have already proven their efficiency for other operational project management challenges (Ranjbar et al., 2009; Vandenheede et al., 2015). Moreover, several authors have already shown that multi-objective adaptations of the scatter search procedure are capable of exceeding or matching the current state of art for multi objective optimisation (Bajestani et al., 2009; Beausoleil, 2006; Ebrahim et al., 2009; Nebro et al., 2008; Rahimi-Vahed et al., 2007; Tavakkoli-Moghaddam et al., 2010). Based on these promising results, a dedicated multi-objective scatter search ($MOSS$) has been designed for incentive contract optimisation. The details of this procedure are outlined in section 4.5. Moreover, the performance of this heuristic was improved further by taking advantage of recent advances in parallel computing, allowing increased calculation speed as well as the parallel creation of multiple populations (see section 4.6).

4.3 Problem Framework

This section describes the framework which is used to analyse the contract optimisation problem faced by the contractor. An overview of this framework is given by figure 3.1, which shows how the different models contained within this framework interact. This framework has already been discussed in detail in chapter 3, but is summarised here for the convenience of the reader.

The framework considers two separate actors: the owner and the contractor. The former controls the properties of the contractual arrangement which governs the relationship, and the latter is in full control over the execution of the project, which is represented as a discrete set of trade-off points (cost (C), duration (D), scope (S) and effort (E)).

The interaction between these two actors can be viewed as a sequential two-player game where the owner makes the first move by defining a contract, and the contractor makes the second move by determining the manner in which the project will be executed by choosing one of the trade-off points. Specifically, it is assumed that the contractor is a profit-maximising actor who will always select the option that maximises his expected outcome. Hence, the owner can determine the performance of a contract using backwards induction in order to calculate his expected profit (expected net owner gain $E[NOG]$). However, the owner is not solely interested in this expected value, the degree to which the contractor's objectives are aligned with his own objectives for the other possible outcome scenario's is also important. This alignment is one of the main reasons why incentive contracts are adopted, since they serve as a safeguard against undesired contractor behaviour. The methodology used to calculate these outcomes is captured by the evaluation model.

Sections 4.3.1, 4.3.2 and 4.3.3 provide a short summary on each of the models contained in the framework. The notation used for the equations used in this model is summarised in appendix 4.B.

4.3.1 The Contract Model

The contract model is a representation of the quantitative targets and incentives which are contained in an agreement between the owner and contractor. The model can be used to describe linear, piecewise linear and quadratic non-linear incentive clauses linked to the cost, duration and scope dimension of the project. Hence, the total incentive awarded (I^{tot}) based on a contract is simply the sum of the incentive awarded for the cost (I^C), duration (I^D) and scope (I^S) of the project.

Within the dynamic of the framework, the owner of the project decides the structure (archetype) of the incentive clauses as well as the parameters such as the targets and sharing ratios for each of the three outcome dimensions. Once this is done, the contract model uses these parameters set by the owner to compute a total incentive amount for any given input combination of cost, duration and scope.

Generally speaking, this means that the creation of a contract within this framework requires the owner to take two decisions. Firstly, the structure (archetype) of the contract has to be chosen for each of the outcome dimensions. Such archetypes can be linear with a given number of segments or quadratic, and for the time dimension a lump-sum linked to a specific deadline. Secondly, the owner has to determine the parameters of the contract archetypes which have been selected for each outcome dimension (cost, duration and scope). Especially the combination of the decision on the global structure with the setting of the parameters greatly increases the size of the solution space. The $MOSS$ heuristic presented in this research takes advantage of this problem structure by using the

structure (i.e. the combination of archetypes for the different dimensions) as a measure for the contract diversity in the solution population (see section 4.5).

To summarise: the **contract** used by the project owner consists of three distinct **contract clauses**, and each of these clauses is linked to a specific outcome dimension: cost, duration or scope. The total incentive awarded is a combination of the incentive amount for the separate outcome dimensions, as calculated by the individual clauses in the contract. These **contract clauses** belong to a specific **archetype** that defines the nature of the equation which is used to calculate the incentive amount. An example of such an archetype is the piecewise linear cost incentive with three segments shown in equation 3.2. Depending on the selected **archetype** a set of **parameters** has to be defined for each **contract clause**. For the 3-piecewise linear contract in equation 3.2 this includes the target (C^t), bounds (B_r^C) and the sharing ratios (s_r^C).

4.3.2 The Trade-off Model

In this chapter, the subcontractor's decision regarding the way in which the project is executed is modelled using a set of discrete trade-off points. Specifically, a subset of the datasets containing such trade-off points made available by created in chapter 3 are used to conduct the experiments described in section 4.7. The points in this dataset were specifically constructed to adhere to a set of six axioms which describe the relation between the four trade-off dimensions (see section 3.4.1).

All instances in the dataset have a total of 1331 modes from which the contractor can choose. This number is reached by defining 11 discrete modes for the duration, scope and effort dimensions and viewing the cost as the dependent variable. Hence, there are 11^3 possible combinations of duration, scope and effort - each associated with a specific cost.

The diversity of this dataset is guaranteed by varying a set of parameters that define the topological structure of the project. These parameters are listed in the top part of table 4.1. Specifically, the slopes of the duration, scope and effort dimensions, the multiplicators which define the interaction effects between two dimensions, the minimal project cost, as well as parameters which express how convex the relationship between the different dimensions and the project cost is. The subset which is used contains one instance with the default parameters (indicated in **bold** in the table), plus one instance for each of the possible variations on the parameters for a total of 17 sets of trade-off points. For an in-depth look at the interpretation of these parameters, the reader is referred to chapter 3.

Model	Parameter	Symbol	Values
Trade-Off	Basic slopes	$\|S_D\|$, $\|S_S\|$, $\|S_E\|$	$\{0.5, 0.75, \mathbf{1}, 1.25, 1.5\}$
	Multiplicators	m_D^S, ..., m_E^S	$\{0, 0.1, \mathbf{0.2}, 0.3, 0.4\}$
	Minimal cost	$C_{\min}$	$\max(\Delta C) \cdot \left\{\frac{1}{0.10}, \frac{1}{0.20}, \frac{\mathbf{1}}{\mathbf{0.30}}, \frac{1}{0.40}, \frac{1}{0.50}\right\}$
	Convexity	CM_D, CM_S, CM_E	$\{0, 0.0625, \mathbf{0.125}, 0.1875, 0.25\}$
Evaluation	Max Effort cost	E_m^{cost}	$\max(\Delta C) \cdot \{0.25, 0.375, \mathbf{0.5}, 0.625, 0.75\}$
	Effort ROI	ROI^E	$\{0, 0.05, \mathbf{0.1}, 0.15, 0.2\}$
	Time value	μ_D	$\max(\Delta C) \cdot \{0.5, 0.75, \mathbf{1}, 1.25, 1.5\}$
	Deadline value	μ_{ls}	$\max(\Delta C) \cdot \{0, 0.05, \mathbf{0.1}, 0.15, 0.2\}$
	Scope valuation	μ_S	$\max(\Delta C) \cdot \{0.5, 0.75, \mathbf{1}, 1.25, 1.5\}$

Table 4.1: Dataset

4.3.3 The Evaluation Model

The evaluation model is used to calculate the impact of using a specific contract for a certain project (the latter being represented by a discrete set of trade-off points, see section 4.3.2). This is achieved by first calculating the contractor's monetary valuation for each of the trade-off points, including the incentive awarded according to the contract as well as the cost of their effort investment. Similarly, a value for each of the trade-off points can be calculated for the owner. This value is based on the outcome for each of the three outcome dimensions perceived by the owner: cost, duration and time. The value of the latter to dimensions also being translated into monetary terms. Similarly, the incentives which are being paid in each of these scenario's are also taken into account in the owner's valuation. Given the pay-off of both actors for each method of executing the project, the expected outcome for both parties is given by the option which maximises the profit of the contractor. Moreover, the alignment of the pay-offs over the various execution modes can also be evaluated. The following paragraphs provide more information on how these variables are calculated and how contract performance is evaluated.

This procedure can be formulated as a mixed-integer multi-objective optimisation problem using expressions 4.2 to 4.11. This optimisation problem aims to maximise the expected net owner gain ($E[NOG]$), while at the same time minimising the mean absolute deviation (MAD) and the mean deviation (MD), two metrics which represent the degree to which the pay-offs of the owner and the contractor are aligned. The sole decision variable available to the owner is the total incentive amount (I_l^{tot}) associated with a project execution mode l. Nevertheless, it is only possible for the owner to manipulate these variables indirectly by deciding the structure and the parameters of the contractual agreement (see section 4.3.1).

Equation 4.3 determines the value of the net owner gain (NOG_l) for each possible execution mode l of the project. This value is equal to the owner value (OV) function

which translates the cost, duration and scope of the project into a monetary valuation for the owner, minus the total incentive (I_l^{tot}) which has been awarded by the owner for a given outcome l. The owner valuation function is simply the sum of the owner value for each of these three dimensions: $OV(C_l, D_l, S_l) = OV^C(C_l) + OV^D(D_l) + OV^S(S_l)$. The monetary valuation of the cost dimension to the owner is simply the inverse of the incurred costs: $OV^C(C_l) = -C_l$. However, for the other two outcome dimensions some parameters have to be defined in order to convert time and scope units into monetary amounts. For the duration dimension, this is done by specifying a monetary value per unit of time (μ_D), as well as a monetary value (μ_{ls}) linked to a specific deadline (DL). The owner value function of the project duration can then be defined as:

$$OV^D(D_l) = \begin{cases} \mu_D \cdot (-D_l) & \text{if } D_l \geq DL \\ \mu_D \cdot (-D_l) + \mu_{ls} & \text{if } D_l < DL \end{cases} \tag{4.1}$$

The valuation of the scope dimensions is done in a similar manner, but no specific lump sum value is used for this dimension. Given that μ_S is the monetary value of an unit of scope to the owner, the owner valuation of the scope dimension simply becomes: $OV^S(S_l) = \mu_S \cdot S_l$.

The net contractor gain (NCG_l^{adj}) is easier to calculate, as is shown by equation 4.4. This value is simply equal to the total incentive amount earned (I_l^{tot}), minus the cost of the additional effort invested by the contractor (E_l^{cost}), multiplied by the average return on investment the contractor attains in his other projects (ROI^E). This is adjustment is the reason why net contractor gain is denoted as NCG^{adj}.

Equation 4.5 continues by defining that the expected net owner gain is equal to the owner gain in the scenario where the contractor's gains are the highest. Should multiple scenario's yield an equal gain for the contractor, the contractor is assumed to select the scenario which yields the highest benefit to the owner. The reason for this being that the contractor has no motivation to decrease the owner's profits, and will always benefit from having satisfied clients when competing for future business.

Equations 4.6 until 4.9 define the owner's and contractor's gain in relative terms ($RNOG_l$ and $RNCG_l^{adj}$) prior to using these relative quantities to calculate the mean deviation (MD) and the mean absolute deviation (MAD) of the pay-offs. These deviations are used to express how good the motivations of the owner and contractor have been aligned by the incentive agreement.

Finally, two constraints are imposed in order for the solutions to be considered valid. Equation 4.10 ensures that the contractor's optimal gain ($\max_l(I_l^{tot})$) exceeds a predefined lower bound ($LB_{I_{\max}^{tot}}$). Various different values for this parameter can be observed in the literature (Abu-Hijleh and Ibbs, 1989; Tang et al., 2008; Veld and Peeters, 1989), ranging

from 0.7% to 5% of the total project cost. Based on these observations, the value of this parameters was set at 5% of the minimal project cost. Similarly, equation 4.11 ensures that the minimal and maximal outcome values for the contractor are at least as broad as a predefined range. This is done to ensure that the contractor is in fact motivated to locate the optimum trade-off point for the contract, should there be no significant variation in his outcomes, it is unlikely that any effort would be spent in locating the optimum. For the experiments conducted in this research, the lower limit ($R_{I^{tot}}$) for the range of contractor outcomes was set to 7.5% of the minimal project cost. This value is similar to the amount of variation which can be observed in contracts 3 in the literature (the minimal return for the contractor).

$$\max\left(E[NOG], \frac{1}{MAD}, \frac{1}{|MD|}\right) \tag{4.2}$$

Subject to:

$$NOG_l = OV(C_l, D_l, S_l) - I_l^{tot} \quad \forall l \in L \tag{4.3}$$

$$NCG_l^{adj} = I_l^{tot} - E_l^{cost}(1 + ROI^E) \quad \forall l \in L \tag{4.4}$$

$$E[NOG] = \max\left[NOG_l | NCG_l^{adj} = \max_l[NCG_l^{adj}]\right] \tag{4.5}$$

$$RNOG_l = \frac{NOG_l - NOG_{\min}}{NOG_{\max} - NOG_{\min}} \quad \forall l \in L \tag{4.6}$$

$$RNCG_l^{adj} = \frac{NCG_l^{adj} - NCG_{\min}^{adj}}{NCG_{\max}^{adj} - NCG_{\min}^{adj}} \quad \forall l \in L \tag{4.7}$$

$$MD = \frac{1}{NL}\sum_{l=1}^{NL}\left[RNCG_l^{adj} - RNOG_l\right] \tag{4.8}$$

$$MAD = \frac{1}{NL}\sum_{l=1}^{NL}\left|RNCG_l^{adj} - RNOG_l\right| \tag{4.9}$$

$$\max_l(I_l^{tot}) \geq LB_{I_{\max}^{tot}} \tag{4.10}$$

$$\max_l(I_l^{tot}) - \min_l(I_l^{tot}) \geq R_{I^{tot}} \tag{4.11}$$

Similarly to the trade-off model (see section 4.3.2) a number of parameters define the nature of the evaluation model. These parameters which also appear in the equations that define the optimisation model are listed in the lower part of table 4.1. These values have also been derived from the dataset created in chapter 3. Again, the values of the basic scenario are indicated in **bold**, and one instance is included for each of the variations of the parameters, resulting in a total of 21 different sets of parameters for the evaluation model.

4.4 Case Study

This section contains a small example which illustrates the methodology in this research. This example is based on a real project.

4.4.1 Introduction

The Belgian Sea Electricity consortium has ordered the installation of twenty-four wind turbines on the Thornton Bank in the North Sea, just off the Belgian coastline. The actual work is to be carried out by a contractor, who went through the selection procedure. Now that the precise scope of the work to be performed is known, the project owner (Sea Electricity) has to decide on the structure of the contractual agreement. The expected budget for this contract is 150 million Euro.

4.4.2 Project Properties

Prior to the design of the incentive structure for the project, experts from both parties have agreed on some basic properties of the project. For the duration of the project both parties agreed that a reasonable execution of the project is unlikely to exceed a total of 550 days ($D \leq 550$). However, the use of progressively more expensive equipment could possibly reduce this duration down to a total of 400 days ($D \geq 400$). Ceteris paribus - assuming that the other dimensions are fixed at their minimum cost level - the approximate cost of this reduction would be 40 million Euro.

One of the key factors that determines the profitability of offshore wind turbines is the uptime of the turbines. Hence, it was deemed sensible by the owner to specify an incentive based on the reliability of the turbines during their first year of operations. The minimal required uptime percentage was set at 60%, below this level the contractor could face serious legal consequences. Nevertheless, it is possible for the contractor to deliver the project with a higher expected uptime level by performing additional quality checks during the installation process. However, an uptime percentage higher than 90% is deemed

unattainable in the first year of operations. Evaluating the contractor's performance based on the perceived uptime was deemed to be both impractical since the effects could only be measured one year after completion, as well as possible unfair since a large amount of randomness affects the actually observed downtime. Hence, it was agreed that the incentive would be based on the amount of quality checks that had been performed during the installation process. This amount approximately doubled when the contractor aimed for the best rather than the lowest acceptable uptime percentage. Within the framework presented in section 4.3 this is easiest to quantify as a scope ranging from a value of 1 for the lowest acceptable number of checks ($S \geq 1$), and a scope value 2 for the greatest number of quality checks ($S \leq 2$). Ceteris paribus, delivering the project at maximal scope is estimated to be approximately 10 million Euro more expensive than delivering at the minimally required scope ($\Delta C_n^S = 10^7$).

The contractor can also influence the project trade-offs by allocating more of his staff and resources to this project. Practically, this means moving these production factors from other projects to this project. This is expressed on a scale from 0 to 1, where 0 is the minimally required staff and 1 signifies the largest useful staff allocation ($E \leq 1, E \geq 0$). Ceteris paribus, the maximal impact of this additional effort is estimated at 10 million Euro.

Using this basic information, as well as information on the interaction effects, a set of 216 execution modes was created for the project by assuming that each of the dependent dimensions could be set at one of six discrete options ($6^3 = 216$). A subset of these points is visualised by figure 4.1, the left panel shows the attainable trade-off points when the invested effort by the contractor is minimal ($E = 0$). The right panel shows the attainable points when the contractor invests the highest possible amount of effort ($E = 1$).

4.4.3 Project Evaluation

Naturally, the outcome of the project has an impact on the owner's profits. For every day the wind park is operational, the owner get a gross income of 230,400 Euro (24 Turbines $\cdot 5 \frac{MW}{Turbine} \cdot 80\%$ uptime $\cdot 24 \frac{hours}{day} \cdot 100 \frac{euro}{MWh} = 230,400 \frac{euro}{day} \Rightarrow \mu_D = 230,400$). Moreover, the owner has signed a contract with the energy distributor, promising that the project would be completed in 475 days time ($DL = 475$). If this deadline is not met, the project owner has to pay a penalty of 2.5 million Euro to the distributor ($\mu_{ls} = 2.5 \cdot 10^6$).

The attained scope also has a significant impact on the energy production of the wind farm in the first year. In case the highest attainable scope is reached, an uptime of 90% can be assumed. However, in case only the lowest acceptable scope level is achieved, this uptime will only be approximately 60%. From a financial perspective this comes down to a 31.54 million Euro difference in gross income ($\mu_S = 31.54 \cdot 10^6$).

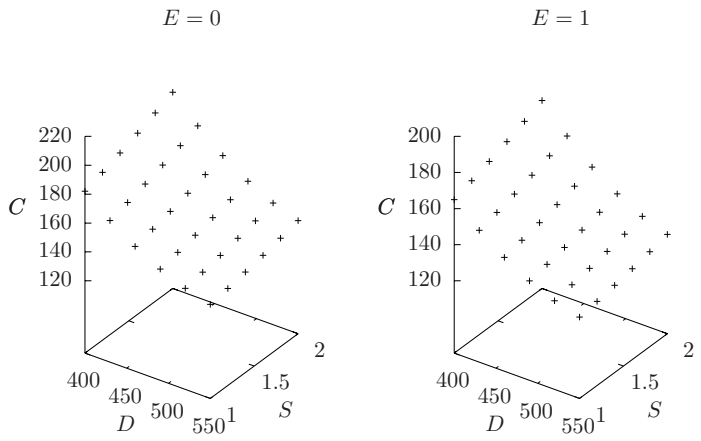


Figure 4.1: Subset of attainable trade-off points in the sample project.

As stated above, the contractor can choose to allocate more of his own resources to the project. To achieve the maximal benefit, the contractor will have to invest 5 million Euro of his own resources ($E_l^{cost} = 5 \cdot 10^6$). Note that in case he decides to invest these resources in another project in his portfolio, he can expect to earn a return on investment of 10% ($ROI^E = 0.1$).

4.4.4 Contract Options

The owner now has the task of designing the most adequate incentive scheme to maximise his own profits, as well as the alignment of his pay-off and that of the contractor. In this case, it is assumed that the choice has already been narrowed down to the contract components presented in table 4.2. This table contains two potential contract components for every incentive dimension, which means that a total of eight (2^3) integrated contracts can be created through combining these different options by selecting one component for each dimension.

4.4.5 Analysis

Using the methodology described above, the contract design problem of the owner in this situation can be analysed. Assuming that the focus of the owner lies on having both the maximal expected net owner gain $E[NOG]$ as well as a minimal mean absolute deviation MAD (to enable a simpler visualisation on a two-dimensional plane, the MD metric is not considered in this example). The different scenarios can be plotted on a two dimensional

ID	Type			C^t, D^t, S^t	Linear Param.		Non-linear Param.				Lump-Sum	
	L	P	N		B_r^C, B_r^D, B_r^S	s_r^C, s_r^D, s_r^S	$I_{\max}$	$I_{\min}$	LB	UB	D_{ls}^t	I_{ls}^D
C1		x		140	{0,140}	{0.5, 0.75}	-	-	-	-	-	-
C2		x		140	{0,140}	{0.75, 0.25}	-	-	-	-	-	-
D1			x	450	-	-	10	-5	25	50	450	0.5
D2			x	450	-	-	5	-10	60	100	460	1
S1		x		1.1	{0, 1.1}	{15.8, 23.7}	-	-	-	-	-	-
S2		x		1.1	{0, 1.1, 1.3}	{15.8, 23.7, 15}	-	-	-	-	-	-

Table 4.2: Sample contract components

representation (see figure 4.2).

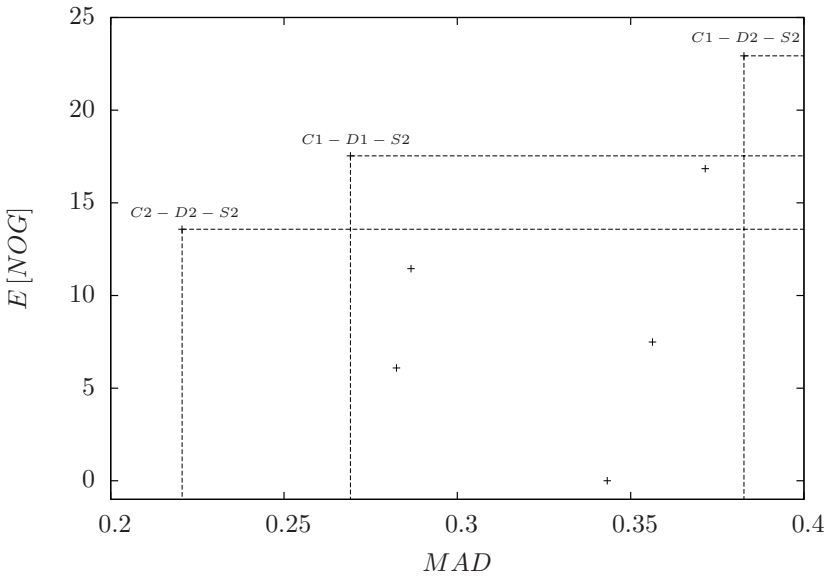


Figure 4.2: Analysis of the example project.

Due to the small number of options considered, it is easy to determine which contracts are undesirable. Looking at figure 4.2, it is clear that contracts which are closer to the top and left of the graph are more desirable. Moreover, a number of dominated contracts, which have both a lower expected net owner gain and a higher level of deviations can be distinguished. This is illustrated by the dotted lines, any contract which is to the bottom right of a dotted line is considered to be an undesirable option. Hence, a Pareto frontier containing three contracts is found. Which of these contracts is ultimately selected depends on the preferences of the owner. A more risk prone owner will be more likely to select the contract if the top right corner ($C1-D2-S2$), whereas a risk averse owner will be more likely to trade in some of his gain for a better pay-off alignment and therefore a

Category	Parameter	Ref	Value		
			PC	*PPc*	*PPv*
Population size	Good set size	4.5.1	200	80	{20,24,...,80}
	Diverse set size	4.5.1	100	50	{34,38,...,50}
Contract	Max # Regions	4.5.1	8	8	8
Intensification	# Moves 1^{st} run	4.5.2	50	25	25
	# Moves per it	4.5.2	50	50	{10,14,...,50}
	Inflow	4.5.2	true / false	true / false	true / false
Stopping criterion	# Generations	4.5.3	100	10	{3,5,7,9}
Recombination	P(Good Parent)	4.5.4	0.5	0.5	{0.25,0.30,...,0.75}
	P(2 Parents)	4.5.4	0.5	0.5	{0.25,0.30,...,0.75}
	# Offspring	4.5.4	100	50	50

Table 4.3: Parameters for the MOSS heuristic

lower risk in case of unforeseen circumstances.

This simple example was presented to illustrate the contract design decision faced by the owner of the project. The following section presents the multi-objective scatter search procedure which was used to facilitate the owner's contract design decision.

4.5 The MOSS Meta-Heuristic

As discussed in section 4.2, scatter search procedures have proven to be effective solution mechanisms for complex multi-objective optimisation problems. The multi-objective scatter search ($MOSS$) procedure proposed in this research is summarised in figure 4.5. This figure shows that the proposed heuristic largely adheres to the basic steps used by earlier authors (Glover et al., 2000):

1. Diversification generation
2. Subset generation method
3. Solution combination method
4. Improvement (Intensification) method
5. Reference set update method

Nevertheless, each of these methods has been tailored to the highly complex solution space of the contract optimisation problem, specifically regarding the multi-dimensional components of the contract structure. Each of these components will be discussed in detail in the sections below.

Table 4.3 lists all the parameters used to tune the behaviour of the MOSS heuristic, as well as the values used for the different implementations (PC, PPc and PPv - see section 4.6) of the heuristic. These parameters are used to define the population size, the nature

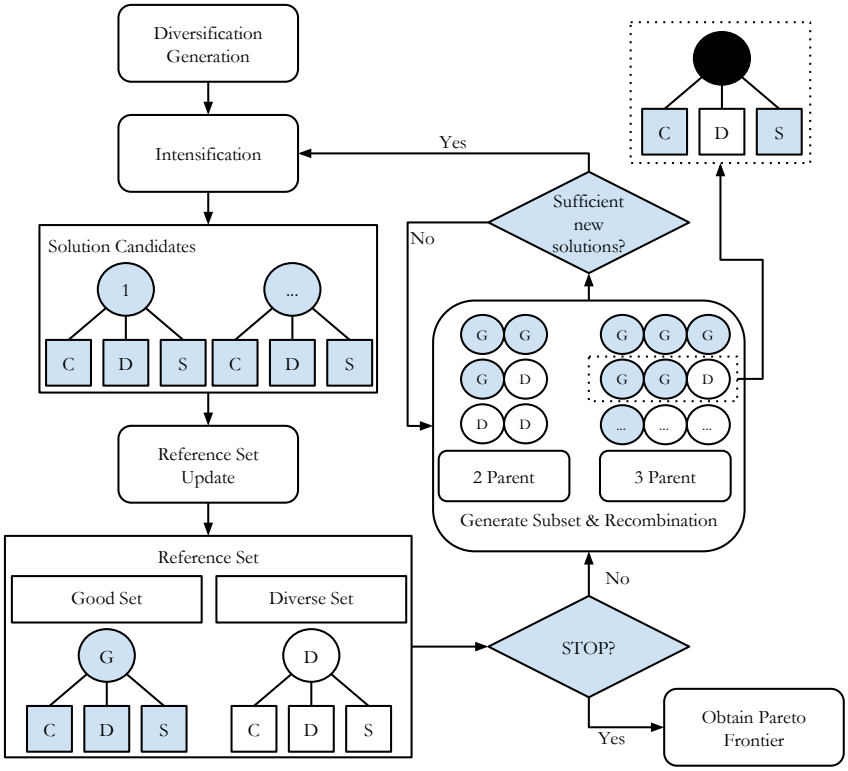


Figure 4.3: Schematic overview of the *MOSS* heuristic.

of the contract, the number of intensification moves, the stopping criterion and the nature of the solution recombination procedure. The sections below explain the precise dynamic of the scatter search procedure, as well as the role of the parameters within this procedure.

4.5.1 Diversification Generation

The algorithm starts by generating a diverse set of solutions, that serve as the starting point for the *MOSS* heuristic. The number of solutions in this initial set is equal to the size of the good set plus the diverse set (see table 4.3). The diversity of the initial population is guaranteed by evenly spreading the created contracts over all the archetypes under consideration. Specifically, each of the contract dimensions (cost, duration and scope) should have at least one contract of each archetype under consideration (linear, piecewise linear with n segments, non-linear). The initial parameters for these contracts (target, sharing ratios, incentive amounts,...) are set by drawing random numbers, which of course have to respect the basic principles of sensible contract design. (e.g: The bounds for piecewise linear contracts must have a correct position: the lower bound for the n-th region must be larger than or equal to the bound for the $(n-1)$-th region.)

To prevent the complete elimination of a contract clause archetype, a lower bound is defined for the size of the diverse set, which can be calculated as: $4 \cdot (r_{\max} + 1) - 3$. Where $r_{\max}$ is equal to the maximal number or regions allowed for the piecewise linear archetypes (see table 4.3). This number is equal to the sum of the number of different contract types used for all dimensions, minus the three archetypes that are guaranteed to be present in the good population (one for each of the three contract clauses). Hence, this formula represents the worst-case scenario where every contract in the good population uses an identical archetype for each of the three outcome dimensions. The method used to update the reference set in turn guarantees that no archetypes are permanently lost from the population (see section 4.5.3). For the computational experiments presented in this research, the maximal number of segments has been set to 8, resulting in a minimal size of 33 for the diverse solution set.

The separate clauses which have been created for the incentive dimensions (cost, duration and scope) are then combined at random in order to form a complete incentive contract, which can be seen as the solution encoding for the problem at hand. Next, these newly created contracts are passed to the intensification procedure (see section 4.5.2), which performs an initial local search in order to improve the values of the randomly chosen contract parameters.

4.5.2 Intensification Procedure

The local search procedure iteratively searches a better value for each of the contract parameters. The outline of this procedure is described by algorithm 1. The procedure randomises the order in which the incentive dimensions (cost, duration and scope) are searched, as well as the order in which the parameters of these contracts are searched. The archetypes for the contract clauses remain unchanged during the intensification procedure.

Algorithm 1 Intensification Procedure

```
1: procedure LOCAL SEARCH(solution, int nbMoves, bool inflow)
2:     contractComponents ← [C, D, S]
3:     contractComponents.randomShuffle
4:     for dim in contractComponents do
5:         parameters ← [p_1, p_2, ..., p_n]
6:         parameters.randomShuffle
7:         for p in parameters do
8:             for i in 1 to nbMoves do
9:                 newSolution ← solution
10:                newSolution.dim.p ← rand | constraints
11:                if newSolution is infeasible then
12:                    continue
13:                if newSolution ≻ solution then
14:                    solution ← newSolution
15:                else if inflow then
16:                    if not (newSolution ≺ solution) then
17:                        candidateSet.add(newSolution)
```

The procedure changes one of the contract's parameters to a random value, and verifies whether this move satisfies the conditions imposed regarding the minimal earnings and the minimal range (see section 4.3.3). Next, the procedure verifies if the newly created solution dominates the original solution. If this is the case, the newly created solution replaces the old solution. The number of new values that are tested is determined by the *nbMoves* parameter. As shown in table 4.3, this parameter is given a different value depending on whether the procedure is being called for the first round of intensification, or during one of the further iterations of the heuristic.

The $inflow$ parameter (see table 4.3) controls what happens when the newly created solution does not dominate the original solution. If inflow is allowed the algorithm verifies if the new solution is dominated by the old solution, if this is not the case the newly created solution may also be a Pareto efficient solution and is therefore added to the set of solution candidates. When inflow is not allowed, the newly created solution is simply discarded. Whereas allowing inflow potentially improves the quality of the good and diverse solution sets, this may cause a substantial inflation of the required computation time due to the larger number of solutions that have to be considered to construct the good and diverse solution sets.

4.5.3 Reference Set Update Method

As shown by figure 4.5, the reference set consists of two distinct populations: the good set and the diverse set. The latter is used to guarantee a better search of the solution space by favouring solutions which are different from those already included in the overall population to be used in recombinations. Specifically for the problem investigated in this research, the diverse set guarantees that no contract archetypes are completely removed from the reference set.

Because the problem which is optimised has multiple objectives, the way in which the good and diverse populations are updated is more complex than for optimisation problems with a single objective. The update method first creates a new set of good solutions, using Pareto-optimality as the basic inclusion criterion. Once the set of good solutions has been created, the algorithm scans the remainder of the candidate solution set for solutions to include in the diverse solution pool. The acceptance condition for the diverse solution pool is based on whether or not a contract of a similar structure (i.e. the same archetypes for the respective outcome dimensions) is already included in the good or diverse populations. When several contracts have identical diversity scores, the tie is broken by measuring the crowding distance of the solutions in the outcome dimensions. The update procedures for both the good and diverse populations are summarized by algorithm 2 and 3, and are described in detail in the remainder of this section.

Algorithm 2 Update Good Population

```
1: procedure UPDATE GOOD POPULATION(int goodSetSize)
2:     candidateSet = candidateSet ∪ goodSet
3:     goodSet ← {}
4:     while | goodSet |< goodSetSize do
5:         /* Extract a Pareto front P from the set of candidate solutions */
6:         P ← {i ∈ candidateSet : {i' ∈ candidateSet : i' ≻ i, i' ≠ i} = ∅}
7:         /* Reduce the size of front P if needed */
8:         while |P|> goodSetSize −|goodSet| do
9:             P ← P \ {i} : {∃i' ∈ P : ε(i, i') = ε_min, i ≠ i'}
10:        goodSet ← goodSet ∪ P
11:        candidateSet ← candidateSet \ P
```

The good solution pool is created by iteratively selecting efficient frontiers from the set of candidate solutions (see algorithm 2). If the size of the newly created efficient frontier is less than or equal to the remaining number of solutions required in the good solution pool, it is integrally added to the solution pool. If the number of solutions in the frontier exceeds the total number of solutions still required in the good solution pool, solutions are eliminated from the efficient frontier based on the crowding distance of the points (i.e. the solutions which have the lowest euclidean distance to other solutions in the solution space are eliminated from the frontier). The solutions belonging to the good solution pool are stored in their respective Pareto frontiers as these have been used to create the good solution pool. This information, specifically the 'level' of the Pareto frontier is later used as a criterion for the subset generation method, as is explained in detail in section 4.5.4.

The purpose of the diverse solution pool is to guarantee that the algorithm continually retains a focus on diversification. Specifically for the problem at hand, the purpose of the diverse solution set is to avoid that complete archetypes or rare combinations of contract archetypes disappear from the solution pool completely. The primary selection criterion for accepting a criterion into the diverse solution pool is its diversity score. This is an integer value in the range $[0, 6]$, where a higher value represents a high structural diversity when compared to solutions already in the good or diverse solution pools (the diversity score of the candidates is continually updated as the diverse solution pool grows by iteratively adding solutions). The interpretation of the seven different diversity scores is as follows:

- **6:** 3/3 dimension archetypes not present.
- **5:** 2/3 dimension archetypes not present.
- **4:** 1/3 dimension archetypes not present.
- **3:** 3/3 combinations not present.
- **2:** 2/3 combinations not present.
- **1:** 1/3 combinations not present.
- **0:** Archetype combination already present in solution pool(s).

The highest diversity scores $[6, 4]$ focus on the number of dimensions (cost, duration and scope) for which the contract archetype (e.g. a piecewise linear contract with 4 segments) that is used is not yet included in either the good or diverse population. The lower diversity scores $[3, 0]$ test if combinations across dimensions of specific archetypes are already present in the reference set.

The calculation of these diversity scores is best illustrated using a simple example. Table 4.4 shows the members of the good and diverse populations, as well as the potential entrants for the diverse population. This table lists the archetypes used for each of the incentive dimensions of the contracts (e.g. $3PL$ represents a piecewise linear contract with three segments, and NL represents a non-linear contract). In order to evaluate the diversity of these candidates, both the good and diverse solutions have to be compared to the candidate solutions.

Good Population			
i	Cost	Duration	Scope
1	3PL	3PL	NL
2	2PL	1PL	1PL
3	3PL	3PL	NL
Diverse Population			
i	Cost	Duration	Scope
1	NL	2PL	2PL
2	1PL	NL	3PL
3	2PL	4PL	5PL

Diverse Candidates			
i	Cost	Duration	Scope
1	4PL	NL	2PL
2	NL	1PL	2PL

Table 4.4: Diversity Score Example

The first step to calculate the diversity score is to verify if there is one or more of the archetypes that has not yet been used for the specific dimension. For the example it can be seen that the first candidate uses a $4PL$ archetype for the cost dimension, which has not yet been used by any of the contracts in either the good or diverse population. The archetypes for the duration and scope dimensions on the other hand have already been used in other contracts (diverse population members 1 and 2). This observation results in a diversity score of **4** for the first candidate for the diverse population.

For the second candidate it can immediately be verified that each of dimension specific archetypes has already been used by one or more of the members in the good or diverse population. Hence, it is already certain that the diversity score, which can be awarded

to this candidate will not exceed 3. In order to determine the exact score the occurrence of archetype combinations has to be verified. Three different combinations can be distinguished: cost-duration, cost-scope and duration-scope. When comparing the combinations to the good and diverse population it is apparent that both the cost-duration and the duration-scope combinations of the second candidate are not yet present. The combination of cost and scope archetypes (NL and $2PL$) however is already present in both the third member of the good population as well as the first member of the diverse population. Based on these observations the diversity score of the second candidate can be set to **2**.

This diversity score calculation is used by algorithm 3, which shows how the diverse population is updated. The procedure starts by adding the diverse solution pool from the preceding iteration to the the pool of candidate solutions. Note that at this time the set of candidate solutions is equal to the set of candidate solutions at the end of the update of the good population (see algorithm 2). Next the diverse solution pool is emptied and diverse solutions are added iteratively until the required number of solutions in the diverse solution pool is attained.

Algorithm 3 Update Diverse Population

```
1: procedure UPDATE DIVERSE POPULATION(int diverseSetSize)
2:     candidateSet = candidateSet ∪ diverseSet
3:     diverseSet = {}
4:     while |diverseSet|< diverseSetSize do
5:         candidateSet.updateDiverseRank()
6:         newEntrant = ∅
7:         maxDistance = 0
8:         for s ∈ candidateSet | rank =rank_max  do
9:             s.eDist = 0
10:            for i in goodSet do
11:                s.eDist + = distance(s,i)
12:            for i in diverseSet do
13:                s.eDist + = distance(s,i)
14:            if s.eDist > maxDistance then
15:                newEntrant = s
16:                maxDistance = s.eDist
17:        diverseSet.add(newEntrant)
```

The loop starts by updating the diversity rank ($\in \{0, 1, 2, ...6\}$) for each of the solutions in the set of candidate solutions. The algorithm continues with the set of solutions which attains the highest diversity rank. Among this set, the solution which has the greatest euclidean distance to the set of solutions in the good population as well as the solutions already included in the diverse solution set is added to the set of diverse solutions.

After the reference set has been updated (see # Generations in table 4.3, the algorithm verifies whether a new iteration has to be started, or if the required number of generations has been attained and the algorithm stops.

4.5.4 Subset Generation Method

Figure 4.4 shows an outline of the subset generation method that selects solutions from the good or diverse populations to be recombined to form new solution candidates using the solution recombination method (see section 4.5.5). First the algorithm randomly decides to use either two or three parents to form a new solution. Also according to a predefined probability the algorithm decides for each individual parent whether it will be drawn from the good or diverse pool (also see figure 4.5 for the basic outline of the procedure). Once the algorithm has decided from which population (good or diverse) the parent will be drawn, each solution has a certain probability of being chosen. The way in which these probabilities are calculated differs between the good solution pool - where solutions are selected based on their overall fitness - and the diverse solution pool, where the diversity of a solution is the main selection criterion.

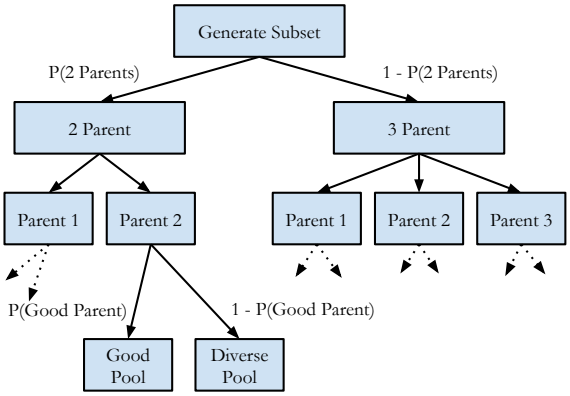


Figure 4.4: Subset Generation.

The selection probabilities of solutions in the good population are based on a two-stage fitness assignment process for multi-objective optimisation which was proposed by Zitzler et al. (2001). The set of points in the good solution pool ($i \in G$) is divided into the set of points lying on the efficient Pareto front (P) and the set of points that are dominated (P') by the points on the Pareto front (hence: $G = P \cup P'$). For each point on the Pareto frontier a fitness score can be calculated as follows:

$$f_i = \frac{n_i}{|P'|+1} \quad \forall i \in P \tag{4.12}$$

where n_i is the number of points in P' that are dominated by the point i:

$$n_i = |\{j | i \succ j, \forall j \in P'\}| \quad \forall i \in P \tag{4.13}$$

As such, each point on the efficient frontier ($i \in P$) receives a fitness score in the range $[0, 1[$, where a lower score signifies a greater solution quality. Hence, the algorithm is tailored to favour solutions situated in more sparsely populated regions of the solution frontier. Next the fitness scores for the points in the set of dominated solutions are calculated:

$$f_j = 1 + \sum_{i \in P | i \succ j} f_i \quad \forall j \in P' \tag{4.14}$$

Equation 4.14 assigns worse fitness values to solutions that are dominated my a larger number of solutions on the Pareto frontier. Based on these fitness values, a selection probability ($Prob$) for each member (i) in the good population (including both dominated (set P') and non-dominated (set P) solutions) can be defined as follows:

$$Prob(i) = \frac{\frac{1}{f_i}}{\sum_{j=1}^{|G|} f_j} \qquad \forall i \in G \tag{4.15}$$

The selection probabilities of solutions in de diverse solution set (D) are calculated based on the diversity scores d_i (see section 4.5.3). The motivation for this being that the key attribute of solutions in the diverse solution pool is their diversity, rather than their fitness value. Hence the selection probability of the solutions in the diverse solution set is calculated as:

$$Prob(i) = \frac{d_i}{\sum d_i} \quad \forall i \in D \tag{4.16}$$

4.5.5 Solution Recombination

The solution recombination procedure takes advantage of the three-dimensional structure of the problem formulation. A newly combined solution simply inherits the cost, duration and scope contract clauses from its parents. For a two parent crossover this means that the new solution will receive one contract component (cost, duration or scope) from one parent, and the remaining two components from the other parent. Similarly if a new solution has three parents, the clauses for each of the dimensions will originate from a different solution. Which parent provides which contract clause is decided randomly. This procedure is visualised in figure 4.5, which shows a complete flowchart of the MOSS meta-heuristic. The origin of the parents for these crossovers are determined by the subset generation method (see section 4.5.4 and figure 4.4).

4.6 Parallel Implementation of the MOSS Heuristic

Thanks to the increased adoption of multi-core processing power in recent decades, the possibility of improving the performance of meta-heuristic procedures though parallelisation has received increasing attention by researchers (Alba et al., 2013). Parallelisation has also been successfully applied to the scatter search meta-heuristic on multiple occasions (Bożejko and Wodecki, 2008; Garcıa-López et al., 2003; López et al., 2006). However, to the best of the author's knowledge no implementation of a parallelised multi-objective scatter search as proposed in this research has been reported in the literature.

The two most convenient forms of parallelisation available for population based meta-heuristics are the parallelisation of intensive computations on population members and the use of multiple populations, each of which is assigned to a processor (Alba et al., 2013). The former technique improves the algorithm performance by assigning each of the available processing cores a part of the population on which local search moves, and consequently a large number of objective function evaluations, are performed. This principle is illustrated on figure 4.5, which shows that during the intensification phase of the algorithm the newly created solutions are assigned to a specific core which performs the local search and then returns the improved solution. The results of such a parallelisation technique are identical to the results of a serial algorithm, but the required wall time can be decreased significantly. Moreover, because each population member is a standalone solution, the algorithm does not have to keep track of any shared resources (i.e. what is happing to the other solutions in no way affects the solution currently being worked on). Because of this, the overhead created by the parallelisation is negligible.

The second method uses parallelisation by creating separate populations which are then searched by the separate processors. This improves the diversity of the search by allowing different populations to evolve in different directions. Often times this diversity is reinforced by using different starting points and/or parametrisations for the populations, increasing the divergence of the different solution populations. The degree of communication between these generations can differ between implementations. At one end of the spectrum the populations can be separated at the start of the algorithm and only be reunited to observe the final solution when the algorithm is finished. Alternatively, cross-breeding between different populations could be allowed after a certain number of generations. The second parallel heuristic tested in this research is situated at the former end of this spectrum. Effectively, this is similar to running the $MOSS$ simultaneously on multiple processors.

Both parallel approaches are tested and compared in this research. The first method method will be dubbed *parallel computations (PC)*. The second method which uses the

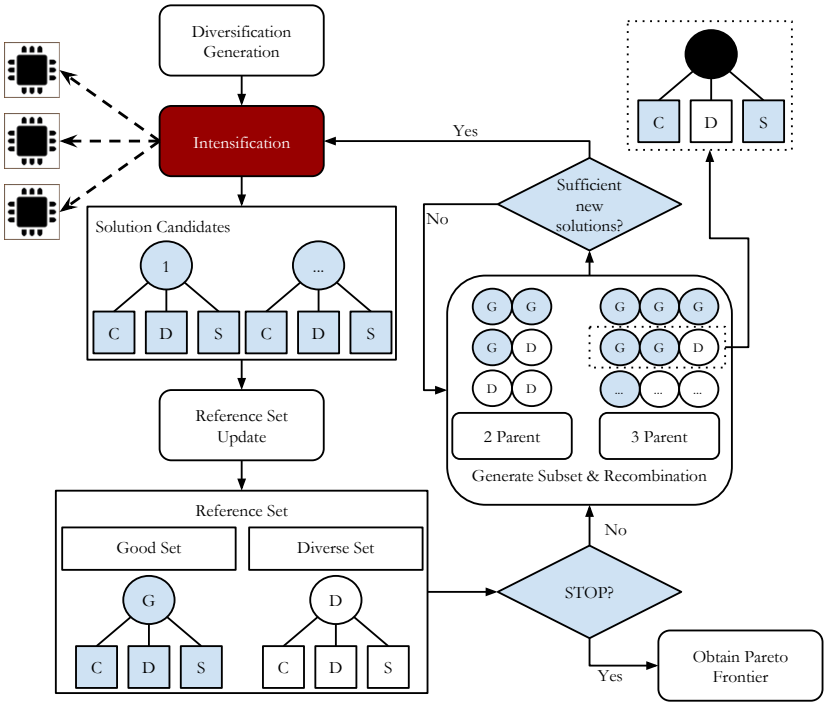


Figure 4.5: *MOSS* heuristic with parallel computations.

available processing power to create multiple populations is referred to as *parallel populations (PP)*. This latter method is tested with both constant (PPc) and variable (PPv) parameters.

The tests themselves were carried out on the STEVIN Supercomputer Infrastructure at Ghent University, specifically on the Golett cluster which uses 2 Intel Haswell processors with 24 cores on each node.

4.7 Computational Experiments

The computational experiments presented in this section compare the quality of the efficient frontiers created using several variations of the *MOSS* heuristic, as well as the full-factorial heuristic used in chapter 3, and a weighted sum approach implemented using Matlab's non-linear solver (Messac, 2015). The dataset used for the experiment contains 38 problem instances, each of which consists of a trade-off and an evaluation model (see sections 4.3.2 and 4.3.3).

Section 4.7.1 gives a brief overview of the settings that have been used for the different solution algorithms. Next, section 4.7.2 discusses the results from the computational experiments.

4.7.1 Solution Methods

4.7.1.1 Full Factorial search (FF)

This brute-force solution method was used in chapter 3 to investigate the relative performance of multi-dimensional versus uni-dimensional incentive contracts. The solution approach starts by creating a set of cost, duration and scope contract clauses and then tests the performance of each possible combination. The parameters for these clauses are set in the way which maximises the coverage over the possible (rational) parameter values.

A total of 189 cost, 4,536 duration and 225 scope contract clauses is used by this solution technique, resulting in a total of 193 million unique combinations. Naturally this method can only test a very small subset of the solution space, making it unlikely that the true efficient frontier can be full approximated. For more information on the specific parameter settings, the reader is referred to chapter 3.

4.7.1.2 Weighted sum approach (NLP)

A second yardstick for the performance of the $MOSS$ heuristic is the traditional weighted sum approach, implemented using Matlab's non-linear solver. Because the dataset is constructed in a way which guarantees that the objectives do not differ by multiple orders of magnitude, the following maximisation objective function can be constructed without having to rescale the different outcome objectives:

$$w_1 \cdot E[NOG] + w_2 \cdot \frac{1}{MAD} + w_3 \cdot \frac{1}{|MD|} \tag{4.17}$$

Where w_1, w_2 and w_3 are the objective function weights that always satisfy $w_1 + w_2 + w_3 = 1$. Figure 4.6 shows how the non-linear optimisation model presented in section 4.3.3 has been implemented using the Matlab optimisation tools. All parameters describing the problem environment and the constraints imposed on the solution are contained within the base model, whereas the contract structure is included as a modular component. Hence, the base model simply contains the logic expressed by equations 4.2 until 4.11. The modular component on the other hand is a representation of the contract model (see section 4.3.1). Hence, the modular component contains decision variables that represent the parameters of the contract clauses, which in turn belong to a specific archetype that was defined extraneously to the non-linear solution procedure.

Because the archetypes of the contract clauses are extraneous to the non-linear optimisation, covering the complete solution space requires looping over the various possible archetype combinations. A disadvantage of this approach is that the time the solver spends in attractive and unattractive archetype combinations is similar. However, including the

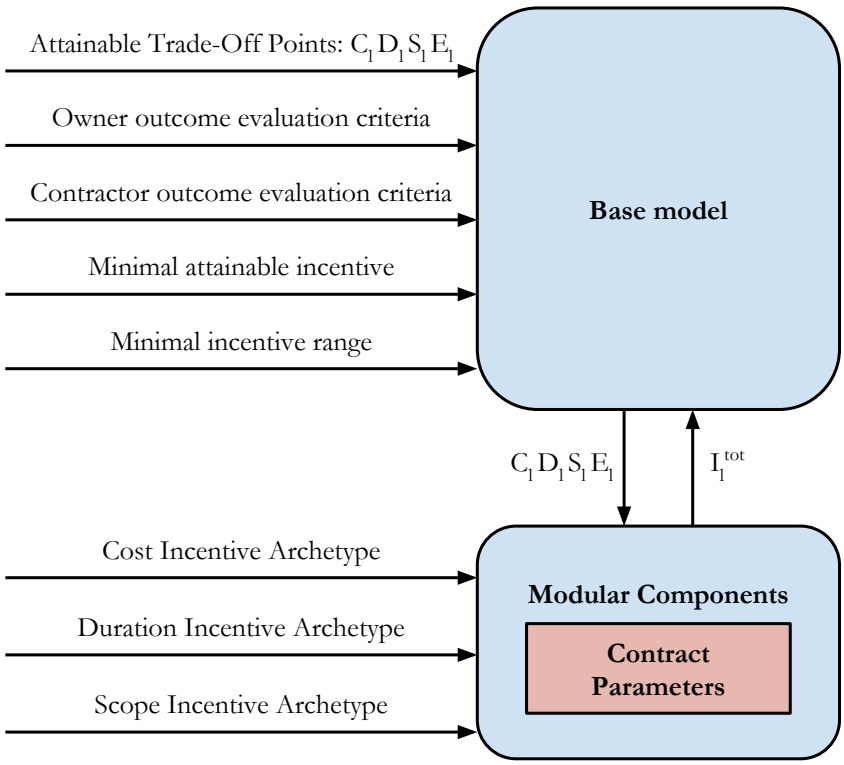


Figure 4.6: Structure of the Matlab optimisation.

choice of archetype combination using additional variables prevented the model from converging on a set of feasible solutions due to the increased complexity of the formulation. The precise way in which the weights are altered is outlined in algorithm 4. The computation time allowed for the algorithm was set to 4,800 seconds, which is equivalent to 4 million evaluations of the objective function.

Algorithm 4 Call Matlab non-linear solver

```
1: procedure NLP(problemInstance)
2:     for w1 = 0; w1 ≤ 1; w1+ = 0.25 do
3:         for w2 = 0; w2 ≤ 1 − w1; w2+ = 0.25 do
4:             w3 = 1 − w1 − w2
5:             for c in cArchetypes do
6:                 for d in dArchetypes do
7:                     for s in sArchetypes do
8:                         callMatlabSolver(w1, w2, w3, c, d, s)
9:     mergeFronts()
```

4.7.1.3 MOSS Heuristic

Six different variations of the $MOSS$ heuristic have been tested. The first difference is based on whether parallel processors are used to reduce wall-time within a single population, or to create multiple parallel populations that can be reunited when the algorithm has finished. The latter category can be implemented using either identical or different parameters for the parallel populations (see table 4.3). Finally, all of these heuristics can be implemented with or without additional inflow allowed from the local search procedure (see section 4.5.2 and algorithm 1). This results in six different variations: parallel computations (PC), parallel populations with constant (PPc) as well as variable parameters (PPv), plus all these implementations with inflow from the intensification heuristic allowed (PCi, $PPci$, and $PPvi$ respectively).

The parameters for the MOSS heuristics (see table 4.3) have been calibrated using a separate dataset to yield the best possible result within approximately 4 million schedule evaluations. Hence, an equal number of schedule evaluations is used as in the Matlab implementation, and substantially less schedule evaluations are used when compared to the full-factorial search which uses 193 million schedule evaluations.

4.7.2 Experimental Results

The results from the computational experiments are summarised in figures 4.7 and 4.8. The quality of the efficient frontiers is judged by looking at three critical aspects (Okabe et al., 2003; Sarker and Coello, 2002): (i) The number of elements on the frontier; a greater number implying a better coverage of the frontier. (ii) The proximity of the frontier to the theoretically optimal efficient frontier, and (iii) the spread of the elements on the efficient frontier. A statistical analysis of these results is also included in appendix 4.A.

4.7.2.1 Elements on the Frontier

The upper left panel of figure 4.7 shows the number of points on the frontiers created by the various algorithms. It is immediately apparent that the full factorial search used in chapter 3 has a factor of magnitude more points on the efficient frontier than any of the other algorithms. This is a logical result from the much larger number of contracts that are tested (193 million versus 4 million) when using the full factorial approach. The next best performance in terms of the number of points is attained by the PC and PCi algorithms, with considerably better performance attained by allowing inflow in the latter algorithm. The $MOSS$ implementations using parallel populations appear to have a larger number of points on the global frontier when different parameters are used for the different populations. For this performance metric, the performance of all $MOSS$ heuristics are at

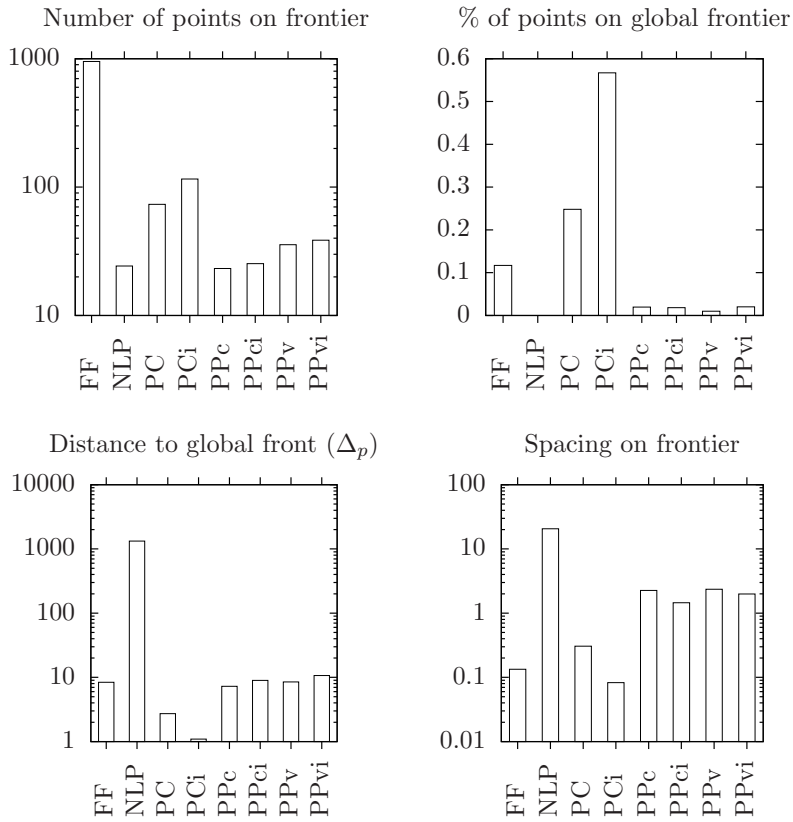


Figure 4.7: Comparison of frontier quality.

least marginally superior to the non-linear weighted sum implementation.

4.7.2.2 Proximity to Global Frontier

One of the simplest metrics to measure the distance of a frontier to the global efficient frontier is the fraction of points that is also present on the global efficient frontier. Since the global efficient frontier is unknown for the problem instances studied in this chapter, the global frontier is approximated by creating a frontier from all non-dominated points found using the different solution techniques.

The values for this proximity metric are shown by the top right panel on figure 4.7. From this panel it can be observed that although the full factorial method results in large frontiers, these do not necessarily contain a proportionate amount of solutions that are on the global frontier. The *MOSS* heuristic that uses a single population and allows inflow is shown to yield the largest fraction of points that are also on the best known global frontier.

A more advanced metric to measure the distance to the global frontier is the averaged Hausdorff distance Δ_p (Schutze et al., 2012). This metric was introduced to avoid two major pitfalls when calculating the distance to the global frontier: (i) convergence to zero when the number of points increases, and (ii) over-sensitivity to the most distant points in the frontiers. The precise calculation of this metric is explained in appendix 4.C.

The lower left panel of figure 4.7 shows the average results for this metric across the different solution methods. Again, it can be observed that although the full factorial approach yields a large frontier, this frontier is not necessarily closer to the global frontier. Rather, it appears that the large size of the frontier generated by the FF method is merely the result from the substantially larger number of contracts tested when using this technique. The best performance for this metric is again attained by the PC and PCi implementations of the $MOSS$ heuristic. Markedly, the performance of the general purpose weighted sum approach (NLP) is exponentially worse when compared to the $MOSS$ heuristic proposed in this research.

4.7.2.3 Spread of Solutions on the Frontier

A fourth and final metric for the quality of the constructed frontier is the 'efficient set spacing' (ESS) Sarker and Coello (2002), which can be calculated using equation 4.18. In this equation n is equal to the number of elements in the Pareto set, and $\varepsilon^i_{\min}$ is the euclidean distance to the point closest to i in the set. $\overline{\varepsilon}_{\min}$ is equal to the average of these minimal distances over all points in the set. A value of 0 for this metric can be interpreted as an ideal distribution of the points in the Pareto frontier, greater values indicating that the points are not spread evenly across the Pareto frontier. Similarly to the average Hausdorff distance (see appendix 4.C), this metric has been calculated using standardised values in the range $[0, 1]$ for all three dimensions respectively.

$$ESS = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(\overline{\varepsilon}_{\min} - \varepsilon^i_{\min})^2} \tag{4.18}$$

The results for this metric are summarised in the bottom right panel of figure 4.7. Again, the PCi implementation of the $MOSS$ heuristic yields the best results, exceeding the results from the full factorial search. Both the full factorial search and the PC implementation of the $MOSS$ heuristic also show a good spread when compared to the performance of the other solution techniques, in particular the weighed sum approach.

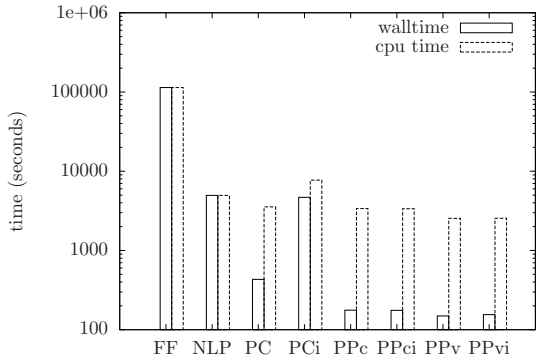


Figure 4.8: Comparison of average resource usage.

4.7.2.4 Resource Usage / CPU Time

Figure 4.8 shows gives an overview of the resource usage for the various solution algorithms. Because of the parallel implementation both wall-time and CPU-time are reported. The difference between these two quantities shows how good the workload is spread over the various cores (i.e. a good wall-time which is significantly lower than the CPU time indicates that the computations are well spread across the different processors). The chart clearly shows that all solution procedures have been given a similar amount of CPU time (with the exception of the full-factorial search which uses a significantly greater amount of CPU time.)

The first key observation here is that the full factorial search (FF) requires a substantially larger amount of CPU and wall-time to complete its search of the solution space. Moreover, as discussed in section 4.7.2 this excessive computation yields worse solutions than the other solution techniques. Hence, by using either the PC or PCi implementation of the $MOSS$ heuristic, all performance criteria other than the number of points can be improved whilst also providing a significant reduction in computation time. Nevertheless, the PCi implementation that allows inflow has a substantially greater wall-time than the PC implementation due to the increased time needed to update the Pareto frontier[1]. This is effect is much less pronounced in the implementations using parallel populations, where the size of the Pareto frontiers of the separate populations is substantially smaller.

[1]Note that the updating of the frontier is not distributed across different processors. This could be an interesting extension for future research.

4.8 Conclusions

This research presented a new optimisation heuristic for incentive contract design in a project environment, from the perspective of the project owner. Computational experiments confirmed that the proposed parallel multi-objective scatter search constructed better frontiers in less computation time than the full-factorial search procedure used by (Kerkhove and Vanhoucke, 2015a). Likewise, the heuristic was also shown to outperform a variable weighted sum approach in terms of both speed and solution quality.

An interesting avenue for future research would be to extend the functionality of the parallel population implementation of the algorithm by allowing interaction between the various populations during the algorithm execution. Ideally this could be done using a number of parameters that define the timing of the cross-breeding as well as the amount of cross breeding at those times.

4.A Statistical Analysis of Relative Algorithm Performance

This appendix investigates the statistical significance of the differences between the frontiers constructed using the various heuristics proposed in this chapter. This is done by means of an $ANOVA$ test that checks if there is a significant difference across the various heuristics as well as a Tukey's range test that provides a one-on-one comparison of the means (Tukey, 1949). The results of these statistics tests are reported for the four different measures of frontier quality as were used in section 4.7.2.1.

4.A.1 Number of Points on the Frontier

For the analysis of the number of points on the frontier the full-factorial (FF) solution procedure is not included in the comparison. The reason for this is that the number of points on the frontier for this technique is several orders of magnitude larger than for the other algorithms. Testing showed that this difference also highly statistically significant ($< 0.000001\%$), but this large difference for this single technique skews the comparisons between the other algorithms.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Number of Points on Frontier	6	263995	43999	39.91	<2e-16
Residuals	252	277828	1102		

Table 4.5: $ANOVA$ analysis for the number of points on the Pareto frontier

The results from the $ANOVA$ analysis are shown in table 4.5. The key observation here being that there are very significant differences in the means for the various heuristics. These differences are investigated in more detail in table 4.6 which shows the one-on-one comparison of means using Tukey's range test. The final column of this table shows the P-values for the average number of points on the frontier for the various solution methods. From this it becomes apparent that the best performing heuristic (PCi) has a mean which is significantly higher than all other solution methods (i.e. the technique performs statistically better for this metric).

heuristics	diff	lwr	upr	Adjusted P value
PC-NLP	49.054054	26.108547	71.99956	0
PCi-NLP	91.243243	68.297736	114.18875	0
PPc-NLP	-1.054054	-23.999561	21.89145	0.9999994
PPci-NLP	1.027027	-21.91848	23.97253	0.9999995
PPv-NLP	11.324324	-11.621183	34.26983	0.764143
PPvi-NLP	14.297297	-8.64821	37.2428	0.5142334
PCi-PC	42.189189	19.243682	65.1347	0.0000023
PPc-PC	-50.108108	-73.053615	-27.1626	0
PPci-PC	-48.027027	-70.972534	-25.08152	0
PPv-PC	-37.72973	-60.675237	-14.78422	0.0000372
PPvi-PC	-34.756757	-57.702264	-11.81125	0.0002063
PPc-PCi	-92.297297	-115.242804	-69.35179	0
PPci-PCi	-90.216216	-113.161723	-67.27071	0
PPv-PCi	-79.918919	-102.864426	-56.97341	0
PPvi-PCi	-76.945946	-99.891453	-54.00044	0
PPci-PPc	2.081081	-20.864426	25.02659	0.999968
PPv-PPc	12.378378	-10.567129	35.32389	0.6802411
PPvi-PPc	15.351351	-7.594156	38.29686	0.4245007
PPv-PPci	10.297297	-12.64821	33.2428	0.8353993
PPvi-PPci	13.27027	-9.675237	36.21578	0.603955
PPvi-PPv	2.972973	-19.972534	25.91848	0.9997413

Table 4.6: Tukey multiple comparison procedure for the number of points on the Pareto frontier

4.A.2 Fraction of Points on Global Frontier

A similar analysis as for the number of points on the frontier is repeated for the second metric: the fraction of points that lies on the global frontier. Table 4.7 shows the result from the $ANOVA$ analysis, and table 4.8 shows the results from the pairwise comparisons. These results indicate that the superior performance of the PCi heuristic is highly statistically significant for this metric as well.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Frac Points on Global	7	10.1	1.4426	69.48	<2e-16
Residuals	288	5.98	0.0208		

Table 4.7: $ANOVA$ analysis for the fraction of points that lie on the global Pareto frontier

heuristics	diff	lwr	upr	Adjusted P value
NLP-FF	-0.117039958	-0.21933347	-0.014746452	0.012724
PC-FF	0.131095342	0.02880184	0.233388849	0.0028414
PCi-FF	0.449918812	0.34762531	0.552212319	0
PPc-FF	-0.097383788	-0.19967729	0.004909719	0.0749652
PPci-FF	-0.09874859	-0.2010421	0.003544917	0.067162
PPv-FF	-0.107203221	-0.20949673	-0.004909715	0.0324942
PPvi-FF	-0.09695835	-0.19925186	0.005335156	0.0775453
PC-NLP	0.248135301	0.14584179	0.350428808	0
PCi-NLP	0.56695877	0.46466526	0.669252277	0
PPc-NLP	0.019656171	-0.08263734	0.121949678	0.9990152
PPci-NLP	0.018291368	-0.08400214	0.120584875	0.9993845
PPv-NLP	0.009836737	-0.09245677	0.112130244	0.9999906
PPvi-NLP	0.020081608	-0.0822119	0.122375115	0.9988684
PCi-PC	0.31882347	0.21652996	0.421116976	0
PPc-PC	-0.22847913	-0.33077264	-0.126185623	0
PPci-PC	-0.229843932	-0.33213744	-0.127550426	0
PPv-PC	-0.238298564	-0.34059207	-0.136005057	0
PPvi-PC	-0.228053693	-0.3303472	-0.125760186	0
PPc-PCi	-0.5473026	-0.64959611	-0.445009093	0
PPci-PCi	-0.548667402	-0.65096091	-0.446373895	0
PPv-PCi	-0.557122033	-0.65941554	-0.454828527	0
PPvi-PCi	-0.546877162	-0.64917067	-0.444583655	0
PPci-PPc	-0.001364802	-0.10365831	0.100928704	1
PPv-PPc	-0.009819434	-0.11211294	0.092474073	0.9999908
PPvi-PPc	0.000425437	-0.10186807	0.102718944	1
PPv-PPci	-0.008454631	-0.11074814	0.093838875	0.9999967
PPvi-PPci	0.00179024	-0.10050327	0.104083746	1
PPvi-PPv	0.010244871	-0.09204864	0.112538378	0.9999876

Table 4.8: Tukey multiple comparison procedure for the fraction of points that lie on the global Pareto frontier

4.A.3 Distance of Points to Global Frontier

The next metric to be analysed is the average distance to the global frontier. For this metric the performance of the non-linear programming model was clearly much worse than the other heuristics. Similarly to the analysis of the number of points on the global frontier the NLP heuristic is therefore excluded from this analysis.

Table 4.9 and 4.10 show the results for the $ANOVA$ and Tukey statistics. These results indicate that the superior performance of the PCi heuristic is statistically insignificant for this metric.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Distance to global	6	2765	460.9	1.817	0.0963
Residuals	252	63931	253.7		

Table 4.9: *ANOVA* analysis for the distance to the global frontier

heuristics	diff	lwr	upr	Adjusted P value
PC-FF	-5.6664436	-16.673378	5.340491	0.7263992
PCi-FF	-7.2921814	-18.299116	3.714753	0.4369449
PPc-FF	-1.134687	-12.141622	9.872248	0.999932
PPci-FF	0.6068149	-10.40012	11.613749	0.9999983
PPv-FF	0.1059545	-10.90098	11.112889	1
PPvi-FF	2.3043885	-8.702546	13.311323	0.9960537
PCi-PC	-1.6257378	-12.632672	9.381197	0.9994485
PPc-PC	4.5317566	-6.475178	15.538691	0.8844191
PPci-PC	6.2732585	-4.733676	17.280193	0.6206897
PPv-PC	5.7723981	-5.234536	16.779333	0.708655
PPvi-PC	7.9708321	-3.036102	18.977767	0.3254899
PPc-PCi	6.1574944	-4.84944	17.164429	0.6414821
PPci-PCi	7.8989964	-3.107938	18.905931	0.3365575
PPv-PCi	7.3981359	-3.608799	18.40507	0.4186377
PPvi-PCi	9.5965699	-1.410365	20.603504	0.1329791
PPci-PPc	1.7415019	-9.265433	12.748436	0.9991819
PPv-PPc	1.2406415	-9.766293	12.247576	0.9998852
PPvi-PPc	3.4390755	-7.567859	14.44601	0.967648
PPv-PPci	-0.5008605	-11.507795	10.506074	0.9999995
PPvi-PPci	1.6975736	-9.309361	12.704508	0.9992932
PPvi-PPv	2.1984341	-8.8085	13.205369	0.9969577

Table 4.10: Tukey multiple comparison procedure for the distance to the global frontier

4.A.4 Spacing of Points on Frontier

The fourth and final metric to be analysed is the spacing of the points on the frontier. Again *ANOVA* and Tukey's test have been applied. The results of these measures can be found in tables 4.11 and 4.12 respectively. The results of these tests indicate that only the performance of the non-linear programming model is significantly worse than that of the other solution algorithms.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Spacing	7	12116	1730.8	12.96	1.75E-14
Residuals	287	38331	133.6		

Table 4.11: *ANOVA* analysis for the spacing of the points on the Pareto frontier

heuristics	diff	lwr	upr	Adjusted P value
NLP-FF	20.48025829	12.219272	28.741244	0
PC-FF	0.17199007	-8.032219	8.376199	1
PCi-FF	-0.05098784	-8.255197	8.153221	1
PPc-FF	2.12466338	-6.079545	10.328872	0.993503
PPci-FF	1.31644708	-6.887762	9.520656	0.999699
PPv-FF	2.22514807	-5.979061	10.429357	0.9914004
PPvi-FF	1.85527445	-6.348934	10.059483	0.9972023
PC-NLP	-20.30826823	-28.569254	-12.047282	0
PCi-NLP	-20.53124614	-28.792232	-12.27026	0
PPc-NLP	-18.35559491	-26.616581	-10.094609	0
PPci-NLP	-19.16381121	-27.424797	-10.902825	0
PPv-NLP	-18.25511023	-26.516096	-9.994124	0
PPvi-NLP	-18.62498384	-26.88597	-10.363998	0
PCi-PC	-0.22297791	-8.427187	7.981231	1
PPc-PC	1.95267332	-6.251535	10.156882	0.9961426
PPci-PC	1.14445702	-7.059752	9.348666	0.9998821
PPv-PC	2.053158	-6.151051	10.257367	0.994734
PPvi-PC	1.68328438	-6.520924	9.887493	0.998495
PPc-PCi	2.17565123	-6.028558	10.37986	0.992494
PPci-PCi	1.36743493	-6.836774	9.571644	0.9996127
PPv-PCi	2.27613591	-5.928073	10.480345	0.9901471
PPvi-PCi	1.90626229	-6.297946	10.110471	0.9966817
PPci-PPc	-0.8082163	-9.012425	7.395992	0.9999889
PPv-PPc	0.10048468	-8.103724	8.304693	1
PPvi-PPc	-0.26938894	-8.473598	7.93482	1
PPv-PPci	0.90870098	-7.295508	9.11291	0.9999753
PPvi-PPci	0.53882736	-7.665381	8.743036	0.9999993
PPvi-PPv	-0.36987362	-8.574082	7.834335	1

Table 4.12: Tukey multiple comparison procedure for the spacing of the points on the Pareto frontier

4.A.5 Conclusion of statistical tests

These statistic tests show that the *PCi* algorithm yields statistically better performance for two of the four indicators used. For the remaining two indicators the performance is also better, but statistically insignificant. Overall, these observations support the notion that the *PCi* is the most adequate heuristic.

4.B Overview of Notation

Table 4.13 presents an overview of all the notation used in this chapter.

Model	Variable	Description
Contract Model	C, D, S	The observed cost, duration and scope at the end of the project.
	E	The effort invested in the project by the contractor.
	I^C, I^D, I^S, I^{tot}	Cost, duration, scope and total incentive awarded.
	C^t, D^t, S^t	Targets for cost, duration and scope as specified in the contract.
	s_r^C	Sharing ratio for (piecewise) linear cost incentive contract in region r.
	v_r^D, v_r^S	Region valuation parameter for piecewise linear duration/scope incentive.
	B_r^C, B_r^D, B_r^S	The lower bound of cost, duration or scope region r in a (piecewise) linear contract.
	UB^C, UB^D, UB^S	Upper bound of the region over which a cost, duration or scope incentive is spread using a non-linear contract.
	LB^C, LB^D, LB^S	Lower bound of the region over which a cost, duration or scope incentive is spread using a non-linear contract.
	$I^C_{\max}, I^D_{\max}, I^S_{\max}$	Greatest incentive amount which can be awarded in a non-linear cost, duration or scope contract.
	$I^C_{\min}, I^D_{\min}, I^S_{\min}$	Greatest disincentive amount which can be awarded in a non-linear cost, duration or scope contract.
	D^t_{ls}	Target date associated with a lump-sum duration incentive amount.
	I^D_{ls}	Incentive amount for a lump-sum duration incentive clause.
Trade-Off	D_i, S_j, E_k	The attainable values for duration, scope and effort as defined by a trade-off model.
	$C_{\min}$	The lowest attainable cost for the project.
	S_D, S_S, S_E	The slope of the relationship between the cost (dependent) and duration, scope and effort (independents).
	m_a^b	Slope multiplicator, max impact of dimension b on the slope of the relationship between dimension a and the cost.
	CM^D, CM^S, CM^E	The convexity magnitude for the relation between C (dependent) and D, S and E (independent) respectively.
Contract Evaluation	$E[NOG]$	Expected Net Owner Gain
	MD	Mean deviation, a measure for the payoff alignment
	MAD	Mean absolute deviation, a measure for the payoff alignment
	E_l^{cost}	Cost of effort
	I_l^{tot}	Total incentive awarded in scenario l
	NCG_l	Net contractor gain for scenario l
	ROI^E	Average ROI for effort investments by the contractor.
	NCG_l^{adj}	Net contractor gain for scenario l taking opportunity cost into account
	NOG_l	The value the owner derives from scenario l
	C_l, D_l, S_l, E_l	The cost, duration, scope an effort associated with scenario l.
	μ_D, μ_{ls}, μ_S	The monetary valuation of time, deadline and scope of the owner
	$RNCG_l^{adj}$	Relative net contractor gain: NCG_l^{adj} rescaled to $[0, 1]$
	$RNOG_l$	Relative net owner gain: NOG_l rescaled to $[0, 1]$
	$I^{tot}_{\max}$	The maximal attainable incentive amount for the contractor
	$LB_{I^{tot}_{\max}}$	A lower bound for the maximal incentive amount which can be earned by the contractor
	R_{Itot}	The range of the incentive earnings by the contractor
	DL	External deadline relevant to the project owner.

Table 4.13: Overview of notation

4.C Average Hausdorff Distance

The average Hausdorff distance is a metric to measure how far a set of points is removed from the global frontier. This metric was introduced by Schutze et al. (2012), and was also used in chapter 3. The average Hausdorff distance (Δ_p) between two sets of points X and Y can be calculated using the following formula:

$$\Delta_p(X,Y) = \max\left[\left(\frac{1}{N}\sum_{i=1}^{N} dist(x_i,Y)^p\right)^{\frac{1}{p}}, \left(\frac{1}{M}\sum_{i=1}^{M} dist(y_i,X)^p\right)^{\frac{1}{p}}\right] \tag{4.19}$$

Where N and M represent the number of points contained in set X and Y respectively. The individual points being represented by x_i and y_i. The *dist* function is used to represent

the shortest Euclidean distance between a point and any of the points in the other set. The average of these values (from both perspectives) are then raised to the power $\frac{1}{p}$, where p equals the number of objectives taken into account when constructing the pareto front (e.g. $p = 3$ for the frontiers constructed in this research).

Specifically for this research, the distances are measured in standardised intervals $[0, 1]$ for each dimension, to avoid allotting a greater weight to dimensions with a greater range of values.

5

Optimising the Contractor's Schedule in Incentivised Projects

This chapter presents a novel quantitative methodology to optimise the scheduling of subcontracted projects from the perspective of the contractor. Specifically, the scenario where the contractor's remuneration is performance dependent is investigated. Based on the incentive methodology introduced in chapter 3, a novel mixed integer programming formulation as well as a greedy local search heuristic to solve the contractor's problem are presented and tested in a computational experiment. For this experiment, a database containing 3,150 contract-project combinations with diverse structures has been created. The results from this experiment demonstrate the efficiency of the MIP formulation even for larger problem instances, as well as the influence of the project and contract structure on the contractor's earnings.

5.1 Introduction

Recent years have shown an increased adoption of incentive clauses in project contracts. Such clauses enable the project owner to mitigate potential conflicts of interest between her/himself and the contractor executing the contract (Bower et al., 2002). During the last decades this has also resulted in a considerable body of literature on the design and effects of incentive contracts (see chapter 2). However, the impact of such contract structures on the contractor's scheduling efforts has remained largely uninvestigated. This research aims to fill this void using recent advances that allow a more quantitative approach to both the project and the contract to which it is subjected (Kerkhove and Vanhoucke, 2015a). To this end, a multi-dimensional multi-mode scheduling model is presented which maximises the contractor's returns. These returns are defined as a combination of the incentives (s)he receives, and the additional costs that are made to improve the performance of the contract.

We consider a situation where the owner of a project subjects the contractor to a contract with one or more incentive clauses. These incentives are grouped into three categories: cost, duration and scope - the latter being an umbrella term for a very wide range of aspects deemed important to the value of the project by the owner.

The focus lies on the scheduling phase of the project, during which the contractor decides the execution modes and starting times of all activities in the project. Depending on the selected execution mode, each activity has a specific cost, duration, scope and contractor effort level. The contractor effort being a quantification of the investments made by the contractor to improve activity performance.

Recent research in this domain has analysed the design of the contract from the perspective of the owner of the project (Kerkhove and Vanhoucke, 2015a)

Chapters 3 and 4 of this dissertation have analysed the contract design decision from the owner's perspective, as well as optimisation procedures that can be used for such an optimisation. However, such a quantitative analysis of the contractor's perspective has not yet been conducted. This chapter aims to fill this gap by applying a similar methodology, but taking an operational perspective on the project by considering the scheduling of individual activities rather than the aggregate project. Table 5.1 gives an overview of the differences between these approaches. The first major difference being that this optimisation takes the contractor's perspective rather than that of the owner. The immediate consequence of this being that the optimisation objective is the contractor's profit rather than the dual objective of maximising motivational alignment and owner profits. The different perspective also has implications for the level at which the analysis is conducted. Whereas the analysis from the owner's perspective looks at the project as

a whole, the contractor has a much more in-depth control over the individual activities in the project. Because of this the decision variables are also defined on a more operational level, specifically as the discrete execution modes that can be selected for each activity. The models used to represent this optimisation problem are based on a combination of the classical scheduling techniques presented in section 5.2 and the contract structures discussed in chapter 2.

	Chapters 3 and 4	**Chapters 5 and 6**
Perspective	Project owner	Contractor
Optimisation Objective	max(Owner profit) and max(Objective alignment)	Max(Contractor profit)
Trade-off	Project level	Activity level
Decision variables	Contract structure Contract parameters	Activity modes

Table 5.1: Comparison of this research and the study in chapter 3.

The remainder of this chapter is structured as follows. Section 5.2 gives an overview of the existing literature, specifically literature on project scheduling. Next, a detailed outline of the problem is presented in section 5.3. This includes a novel mixed integer programming formulation (section 5.3.3). This problem formulation is then illustrated using an example in section 5.4. Section 5.5 discusses the methodology used to generate realistic contracts (section 5.5.1) and projects (section 5.5.2). Both exact and heuristic solution techniques are discussed in section 5.6. The results of the experiments conducted with these solution techniques are discussed in section 5.7 prior to formulating an overall conclusion in section 5.8.

5.2 Multi-Mode Scheduling Literature

The research presented in this chapter builds on two key research domains. The first research area is the literature on multi-mode project scheduling, specifically concerning the balancing of multiple outcome dimensions of projects. The second research domain investigates the contract structures employed by project owners to improve the alignment of the contractor's motivations with their own. The former domain will be discussed in detail in section 5.2, and the second has been treated extensively in chapter 2.

Scheduling projects has been a key topic in operations research since the 1950s. A classical project scheduling problem which is frequently encountered in practice is the time-cost trade-off problem (De et al., 1995). This problem formulation requires the scheduler to decide on the manner in which specific activities are to be executed in order to construct a schedule for the complete project.

The simplest form of this problem assumes a linear relationship between the duration and the cost of every activity (Kelley and Walker, 1959). More recent research frequently uses more complex models that include more than two dimensions, and uses discrete rather than continuous trade-off points which better reflect the nature of the decisions made by the scheduler in practice (Ghodsi et al., 2009). One of the most actively researched variations on this problem is the multi-mode resource constrained scheduling problem, where the duration and resource consumption of an activity depends on the selection of a discrete execution mode (see Van Peteghem and Vanhoucke (2014) for an overview of recent research on this topic).

A substantial number of authors have defined trade-off problems including the scope (sometimes also defined as quality) of activities as well as the cost and duration (Afshar et al., 2007; Hu and He, 2014; Iranmanesh et al., 2008; Keren and Cohen, 2012; Pollack-johnson and Liberatore, 2006; Pour et al., 2010; Rahimi and Iranmanesh, 2008; Ramón and Cristóbal, 2009; Shahsavari Pour et al., 2010; Tareghian and Taheri, 2006; Tavana et al., 2013; Zhang and Xing, 2010; Zhang et al., 2012). These three dimensions are commonly known as the *iron triangle* in project management (Marques et al., 2011).

Within the context of subcontracting, the three-dimensional trade-off (cost, duration and scope) has also been extended by a fourth dimension which can be dubbed *contractor effort* (Abu-Hijleh and Ibbs, 1989; Arditi and Yasamis, 1998; Bayiz and Corbett, 2005; Chapman and Ward, 1994; Choi and Kwak, 2012; El-Rayes, 2001a; El-Rayes and Kandil, 2005; Kandil and El-Rayes, 2006; Kerkhove and Vanhoucke, 2015a,b; Lee and Thomas, 2007; Lippman et al., 2013; Sillars, 2007). Although the nomenclature difffers between the different authors, this dimension is used to reflect the contractor's investments beyond the minimally required investment to complete the project, which are not directly reimbursed by the project owner. An example of this is the allocation of additional personnel to increase the speed at which the project is completed in projects where wages are not directly paid for by the project owner (El-Rayes and Kandil, 2005).

A variety of solution techniques has been employed to solve these multi-dimensional scheduling problems. The most popular solution techniques being genetic algorithms (El-Rayes and Kandil, 2005; Hu and He, 2014; Iranmanesh et al., 2008; Kandil and El-Rayes, 2006; Pour et al., 2010; Shahsavari Pour et al., 2010; Tavana et al., 2013), as well as other population-based meta-heuristics such as ant colony (Afshar et al., 2007) and particle swarm heuristics (Rahimi and Iranmanesh, 2008; Zhang and Xing, 2010; Zhang et al., 2012). Several authors have also used mixed integer and mathematical programming approaches to solve these scheduling problems (Khang and Myint, 1999; Pollack-johnson and Liberatore, 2006; Ramón and Cristóbal, 2009; Tareghian and Taheri, 2006; Tavana et al., 2013).

Similar to classic project scheduling optimisation several authors focus on optimising one of the trade-off dimensions, while setting minimal requirements for the other dimensions (Hu and He, 2014; Pollack-johnson and Liberatore, 2006; Pour et al., 2010; Ramón and Cristóbal, 2009; Tareghian and Taheri, 2006). A substantial number of authors has however departed from this premise and elected to optimise multiple objectives simultaneously, creating a set of non-dominated solutions rather than a single solution point (Afshar et al., 2007; El-Rayes and Kandil, 2005; Iranmanesh et al., 2008; Kandil and El-Rayes, 2006; Keren and Cohen, 2012; Khang and Myint, 1999; Rahimi and Iranmanesh, 2008; Shahsavari Pour et al., 2010; Tavana et al., 2013; Zhang and Xing, 2010; Zhang et al., 2012).

5.3 Problem Description

This chapter discusses a multi-mode scheduling problem where the contractor optimises the high-level trade-off between investing effort in a project and the return he gets in the form of an additional incentive payment. This is done by optimising the trade-offs at the level of the individual activities by selecting the correct activity modes. Moreover, the model is designed to be capable of handling several types of linear and piecewise linear contract clauses, as well as incentive contracts defining incentives for multiple dimensions simultaneously. An overview of the notation used in this section is given in appendix 5.C.

An important note regarding the nature of the optimisation problem is that whereas the contractor's decisions are made on the level of the activity, the incentive contract is linked to the outcome of the complete project. Hence, section 5.3.1 explains the nature of the individual activities and section 5.3.2 analyses the contracts linked to the performance of the complete project. This aggregation of the activity level information is made explicit in the linear programming formulation (section 5.3.3).

5.3.1 Project Model

The project is defined as a directed acyclic graph $G = (V, E)$ representing an activity on the node network, where V is the set of nodes and E is the set of arcs connecting the nodes. Each activity $i \in V$ has a number of discrete modes ($m = 1, ..., M_i$). Each of these activity modes has a value for the four trade-off dimensions associated with it: cost (C_{im}), duration (D_{im}), scope (S_{im}) and contractor effort (E_{im}). The cost of the contractor effort (E^{cost}_{im}) is also defined for every mode of the activity. An important distinction is between the effort cost (E^{cost}_{im}) and the regular cost (C_{im}) is that the former is paid for by the contractor, whereas the latter represent the direct costs associated with an activity that are paid for by the owner of the project. The values for these activity modes are set based

on the methodology introduced in chapter 3. More details on how the values for these dimensions for the projects in the dataset have been created can be found in section 5.5.2.

5.3.2 Contract Models

The simplest and most frequently studied form of incentivised contract is the linear incentive cost incentive contract. Based on the notation used by Perry and Barnes (2000) this contract can be represented as:

$$Y = F + I^C = F + s^C \cdot (C^t - C) \tag{5.1}$$

where Y represents the total remuneration of the contractor, consisting of a fixed component F and an incentive amount I^C. This incentive is calculated based on the magnitude of the deviation between the cost target C^t and the cost observed at the completion of the project C, multiplied with a pre-specified sharing ratio $s^C \in [0, 1]$.

A single **incentive contract** can include multiple **incentive clauses**. Such a contract clause determines an incentive amount based on the outcome of the project for a single dimension (cost, duration or scope). Hence, using the methodology used in this research an incentive contract can consist of up to three incentive clauses: one for cost, duration and scope respectively.

The manner in which the incentive amounts for the duration and scope dimensions are calculated is similar to the incentives calculated for the cost dimension with one key difference. Where the magnitude of the cost incentive is determined by a fraction s^C, representing the part of the cost over- or under-run transferred to the contractor, the other two dimensions require a monetary valuation per unit of time and scope to be specified. The simplest linear for of such contract clauses can be expressed as follows:

$$I^D = v^D \cdot (D^t - D) \tag{5.2}$$

$$I^S = v^S \cdot (S - S^t) \tag{5.3}$$

Where v^D is equal to the (dis-)incentive amount awarded per time unit of deviation from the duration target D^t, and v^S equals the (dis-)incentive amount for the scope dimension per unit of scope.

Piecewise linear alternatives for these simple linear equations are also often used in practice, as is illustrated by (Broome and Perry, 2002) in several practical examples. The basic principle of these piecewise linear equations is that different sharing ratios and valuations can be used for the different regions (indexed with r). Equations 5.4, 5.5 and

5.6 illustrate such piecewise linear equations with three segments[1].

$$I^C = \begin{cases} s_1^C \cdot (C^t - B_2^C) + s_2^C \cdot (B_2^C - C) & \text{if } C > B_2^C \\ s_1^C \cdot (C^t - C) & \text{if } B_1^C \leq C \leq B_2^C \\ s_1^C \cdot (C^t - B_1^C) + s_0^C \cdot (B_1^C - C) & \text{if } C < B_1^C \end{cases} \tag{5.4}$$

$$I^D = \begin{cases} v_1^D \cdot (D^t - B_2^D) + v_2^D \cdot (B_2^D - D) & \text{if } D > B_2^D \\ v_1^D \cdot (D^t - D) & \text{if } B_1^D \leq D \leq B_2^D \\ v_1^D \cdot (D^t - B_1^D) + v_0^D \cdot (B_1^D - D) & \text{if } D < B_1^D \end{cases} \tag{5.5}$$

$$I^S = \begin{cases} v_1^S \cdot (B_2^S - S^t) + v_2^S \cdot (S - B_2^S) & \text{if } S > B_2^S \\ v_1^S \cdot (S - S^t) & \text{if } B_1^S \leq S \leq B_2^S \\ v_1^S \cdot (B_1^S - S^t) + v_0^S \cdot (S - B_1^S) & \text{if } S < B_1^S \end{cases} \tag{5.6}$$

The magnitude of the incentive awarded is specified for each of the regions using the s_r^C, v_r^D and v_r^S parameters for the cost, duration and scope dimensions respectively. Moreover, this type of equation also requires the project owner to specify bounds for each of the regions. Using the notation above the B_r^C represents the lower bound of the r-th region for the cost dimension. A similar interpretation is used for the bounds of the duration (B_r^D) and scope (B_r^S) dimensions. Note that B_r^C can also be interpreted as the upper bound of the $(r-1)$-th region.

Figure 5.1 presents a graphical interpretation of the piecewise linear cost contract with three segments as described by equation 5.4. From this figure it is clear that rate at which the incentive amount is accrued differs between the different segments. Given that there are three segments in the contract ($NR^C = 3$), 4 bound variables B_r^C are defined. Both the first and last of these bounds are always fixed: $B_1^C = 0$ and $B_{NR^C+1}^C = \infty$, where $B_{NR^C+1}^C$ can be interpreted as the upper bound of the NR^C-th region.

An important remark on these expressions is that the target (C^t, D^t, D^t) is not necessarily situated in the center region of the piecewise linear contract. The models presented in this research are especially designed to be capable of dealing with any position of the target within such contacts, as is evident from the linear programming formulation in section 5.3.3.

[1]The construction of incentive contracts with a greater or smaller number of segments is trivial and will not be discussed in detail.

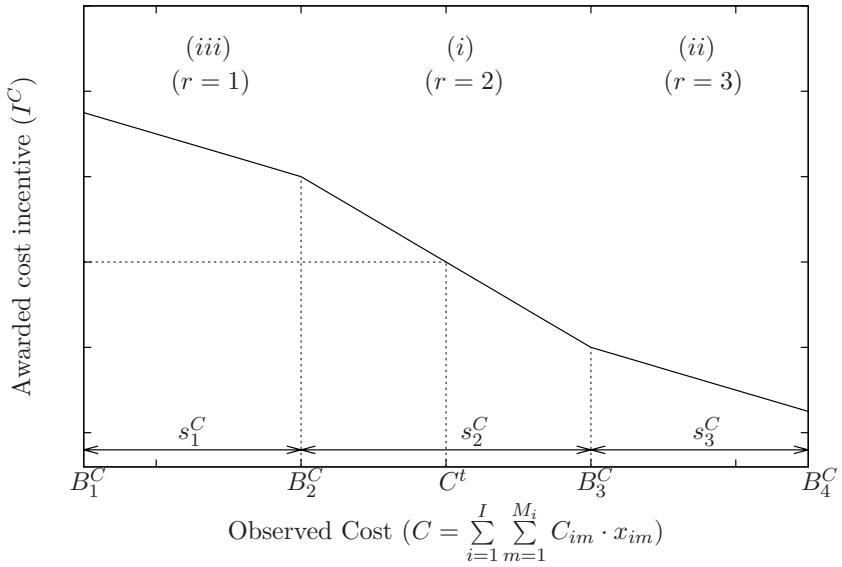


Figure 5.1: Piecewise linear cost contract with 3 segments.

5.3.3 Linear Programming Formulation

The linear programming model formulation is divided into a main model and the incentive-clause specific equations. The main model contains the binary decision variable x_{im} which defines whether (1) or not (0) mode m of activity i has been selected. With each of these modes five parameters are associated: activity cost (C_{im}), duration (D_{im}), scope (S_{im}), effort (E_{im}) and effort cost (E_{im}^{cost}), as defined in section 5.3.1.

The main model also enforces the precedence relations between the activities in the project, and calculates the sum of the different incentive clauses (I^{tot}). The values for the individual clauses (I^C, I^D, I^S) are calculated by the equations explained in sections 5.3.3.2 and 5.3.3.3 for linear and piecewise linear contracts respectively. As such, the actual set of equations used in the linear model differs depending on the contract which has been implemented.

5.3.3.1 Main Model

$$\text{Max. } I^{tot} - \sum_{i=1}^{|V|} \sum_{m=1}^{M_i} E_{im}^{cost} \cdot x_{im} \tag{5.7}$$

$$\sum_{m=1}^{M_i} x_{im} = 1 \quad \forall i \in V \tag{5.8}$$

$$f_i \geq f_j + \sum_{m=1}^{M_i} (D_{im} \cdot x_{im}) \quad \forall i \in V \setminus \{1\}, j \in P_i \tag{5.9}$$

$$f_1 = 0 \tag{5.10}$$

$$I^{tot} = I^C + I^D + I^S \tag{5.11}$$

$$x_{im} \in \{0, 1\} \quad \forall i \in V, m = 1, ..., M_i \tag{5.12}$$

$$f_i \geq 0 \quad \forall i \in V \tag{5.13}$$

The objective of this model is the maximisation of the contractor's profits, as reflected by expression 5.7. This objective equals the total incentive earned, minus the total cost of effort invested in the project. Because the fixed remuneration is constant, this factor is not included in the objective function.

Constraint 5.8 guarantees that exactly one execution mode is selected for every activity in the project. Using the variable f_i to represent the finishing time of activity i; the precedence relations of the project are enforced by equation 5.9, in combination with equation 5.10 which specifies that the first dummy activity finishes at time 0. In equation 5.9 P_i represents the set of predecessors of activity i. The total incentive amount earned I^{tot} is set as equal to the sum of the incentives for the separate dimensions in equation 5.11. Finally, the nature of the variables is enforced by equations 5.12 and 5.13.

5.3.3.2 Linear Contracts

Linear contract clauses with only a single segment can be added using only a single equation. The equations used are simply adaptations of equations 5.1, 5.2 and 5.3 that accommodate the definitions of the multiple modes used in the model. Equations 5.14, 5.15 and 5.16 determine the value for the cost, duration and scope incentives when linear contracts are used.

$$I^C = s_1^C \cdot (C^t - \sum_{i=1}^{|V|} \sum_{m=1}^{M_i} C_{im} \cdot x_{im}) \tag{5.14}$$

$$I^D = v_1^D \cdot (D^t - f_{|V|}) \tag{5.15}$$

$$I^S = v_1^S \cdot (\sum_{i=1}^{|V|} \sum_{m=1}^{M_i} S_{im} \cdot x_{im} - S^t) \tag{5.16}$$

5.3.3.3 Piecewise Linear Contracts

Constructing linear equations for the piecewise linear contracts is slightly more complex and requires each segment of the piecewise linear equation to be evaluated separately. Hence, the total cost incentive becomes the sum of the incentive awarded in each segment r:

$$I^C = \sum_{r=1}^{NR^C} I_r^C \tag{5.17}$$

Two assisting binary variables are used to determine the positioning of the cost outcome relative to the bounds of the piecewise linear regions: α_r^C which is equal to 1 only if the project cost exceeds the upper bound of the region r, and β_r^C which is equal to 1 only if the project cost lies below the lower bound of region r. The values for these two binary variables are determined using by expressions 5.18 until 5.22. These expressions use a large value M (i.e. the big-M method), which should not be confused with the last mode index M_i of activity i.

$$\sum_{i=1}^{|V|} \sum_{m=1}^{M_i} C_{im} \cdot x_{im} > B_{r+1}^C - M \cdot (1 - \alpha_r^C) \quad \forall r = 1, ..., NR^C \tag{5.18}$$

$$\sum_{i=1}^{|V|} \sum_{m=1}^{M_i} C_{im} \cdot x_{im} \leq B_{r+1}^C + M \cdot \alpha_r^C \quad \forall r = 1, ..., NR^C \tag{5.19}$$

$$\sum_{i=1}^{|V|} \sum_{m=1}^{M_i} C_{im} \cdot x_{im} > B_r^C - M \cdot \beta_r^C \quad \forall r = 1, ..., NR^C \tag{5.20}$$

$$\sum_{i=1}^{|V|} \sum_{m=1}^{M_i} C_{im} \cdot x_{im} \leq B_r^C + M \cdot (1 - \beta_r^C) \quad \forall r = 1, ..., NR^C \tag{5.21}$$

$$\alpha_r^C, \beta_r^C \in \{0, 1\} \quad r = 1, ..., NR^C \tag{5.22}$$

Next, a number of equations are added which calculate the incentive amount added for each individual segment. The type of equations added differ depending on the relative position of the target to the region. Three different scenario's can be distinguished: (i) the target is contained within the bounds of region r: $B_r^C \leq C^t \leq B_{r+1}^C$, (ii) the target is lower than the lower bound of the target region r: $C^t < B_r^C$, (iii) the target exceeds the upper bound of the region r: $B_{r+1}^C < C^t$. These three scenario's are indicated in figure 5.1. The equations used in each of these scenario's are discussed in the following

paragraphs. Because the equations used for the different dimensions are largely similar, only the equations for the cost dimension are covered in detail. The equations for the duration and scope dimension are included in appendix 5.B.

5.3.3.3.1 (i) Region containing the target Assuming that the region r contains the target cost, three possible scenario's for the location of the total cost of the project denoted as $\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}$ can be distinguished: the total cost can fall within, above or below the range r. Which of these scenario's is true determines the formula that has to be used to calculate the correct incentive accrued within the region.

This is modelled using three pairs of inequalities: equations 5.23 and 5.24 for the scenario where the final cost falls within the region r, equations 5.25 and 5.26 when the cost exceeds region r's upper bound, and equations 5.27 and 5.28 when the cost falls below region r's lower bound. Using the assisting variables α_r^C and β_r^C the correct pair of equations is made binding, and all others are made non-binding by using a large value defined as M.

$$I_r^C \geq s_r^C(C^t - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) - M(\alpha_r^C + \beta_r^C) \tag{5.23}$$

$$I_r^C \leq s_r^C(C^t - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) + M(\alpha_r^C + \beta_r^C) \tag{5.24}$$

$$I_r^C \geq s_r^C(C^t - B_{r+1}^C) - M(1 - \alpha_r^C) \tag{5.25}$$

$$I_r^C \leq s_r^C(C^t - B_{r+1}^C) + M(1 - \alpha_r^C) \tag{5.26}$$

$$I_r^C \geq s_r^C(C^t - B_r^C) - M(1 - \beta_r^C) \tag{5.27}$$

$$I_r^C \leq s_r^C(C^t - B_r^C) + M(1 - \beta_r^C) \tag{5.28}$$

5.3.3.3.2 (ii) Target is lower than region lower bound The reasoning for the second scenario is largely similar, equations 5.29 and 5.30 are binding when the total cost falls within the range r, equations 5.31 and 5.32 are made binding when the total cost falls above the range r, and equations 5.33 and 5.34 are binding when the total cost falls below the range r.

$$I_r^C \geq s_r^C(B_r^C - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) - M(\alpha_r^C + \beta_r^C) \tag{5.29}$$

$$I_r^C \leq s_r^C(B_r^C - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) + M(\alpha_r^C + \beta_r^C) \tag{5.30}$$

$$I_r^C \geq s_r^C(B_r^C - B_{r+1}^C) - M(1 - \alpha_r^C) \tag{5.31}$$

$$I_r^C \leq s_r^C(B_r^C - B_{r+1}^C) + M(1 - \alpha_r^C) \tag{5.32}$$

$$I_r^C \geq 0 - M(1 - \beta_r^C) \tag{5.33}$$

$$I_r^C \leq 0 + M(1 - \beta_r^C) \tag{5.34}$$

5.3.3.3.3 (iii) Target exceeds region upper bound The third and final scenario again uses the same methodology: equations 5.35 and 5.36 are binding when the total cost falls within the range r, equations 5.37 and 5.38 are made binding when the total cost falls above the range r, and equations 5.39 and 5.40 are binding when the total cost falls below the range r.

$$I_r^C \geq s_r^C(B_{r+1}^C - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) - M(\alpha_r^C + \beta_r^C) \tag{5.35}$$

$$I_r^C \leq s_r^C(B_{r+1}^C - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) + M(\alpha_r^C + \beta_r^C) \tag{5.36}$$

$$I_r^C \geq 0 - M(1 - \alpha_r^C) \tag{5.37}$$

$$I_r^C \leq 0 + M(1 - \alpha_r^C) \tag{5.38}$$

$$I_r^C \geq s_r^C(B_{r+1}^C - B_r^C) - M(1 - \beta_r^C) \tag{5.39}$$

$$I_r^C \leq s_r^C(B_{r+1}^C - B_r^C) + M(1 - \beta_r^C) \tag{5.40}$$

5.4 Example

This section presents a small example to clarify the theory presented in the preceding sections. The example project contains four activities ($|V|= 4$), and four arcs representing the precedence relations ($|E|= 4$) as shown in figure 5.2. Activities 1 and 4 are dummy activities, representing the dummy start and finish times of the network respectively.

Both of the two non-dummy activities have two possible execution modes, each of which is associated with five parameters describing the nature of the activities when executed

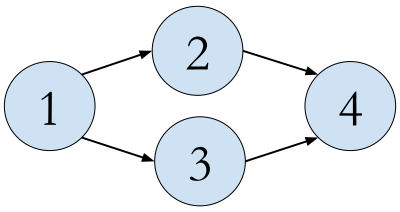


Figure 5.2: Example project network.

with these modes (see table 5.2)[2]. Hence, there are a total of four possible ways to execute the project, which can be expressed using the decision variable x_{im}. A summary of the possible project outcomes is presented by table 5.3.

i	m	C_{im}	D_{im}	S_{im}	E_{im}	E_{im}^{cost}
2	1	10	9	7	5	5
2	2	5	15	12	15	3
3	1	8	6	9	3	4
3	2	3	7	1	10	2

Table 5.2: Modes for non-dummy activities.

Project execution				**Project outcomes**				
x_{21}	x_{22}	x_{31}	x_{32}	C	D	S	E	E^{cost}
1	0	1	0	18	9	16	8	9
1	0	0	1	13	9	8	15	7
0	1	1	0	13	15	21	18	7
0	1	0	1	8	15	13	25	5

Table 5.3: Possible outcomes.

Furthermore, assume that this project is subjected to an incentive contract consisting of a piecewise linear incentive clause with three segments for the cost dimension (see figure 5.1), and a linear incentive clause for both the duration and the scope dimension. The parameters of the piecewise linear cost contract are set as follows: The target cost (C^t) is set at 14, the bounds are given the following values: $B_1^C = 0$, $B_2^C = 10$, $B_3^C = 15$, $B_4^C = \infty$, and the sharing ratios are set as follows: $s_1^C = .25$, $s_2^C = .5$, $s_3^C = .25$. The linear contracts for the duration and the scope only require the target and the valuation per unit to be set: $D^t = 14$, $v^D = .5$, $S^t = 15$, and $v^S = .75$.

[2]The dummy activities only have a single execution mode, with a value of zero for all parameters and have been omitted from the table.

Hence, the mathematical model representing this scenario would add the equations 5.15 and 5.16 to represent the linear duration and scope contracts respectively. Because the cost contract is piecewise linear, equations 5.17 up until 5.22 are added to calculate the incentive amount as the sum of the different segments and to assign the correct values to the dummy variables α_r^C and β_r^C. Next, for each region a number of equations is added depending on its relative positioning to the target. Hence, for the leftmost region (see figure 5.1) equations 5.35 until 5.40 are added[3]. Similarly, the center region requires equations 5.23 until 5.28 to be added and the rightmost region requires equations 5.29 up until 5.34 to be added.

When considering the fourth execution scenario ($x_{21} = 0$, $x_{22} = 1$, $x_{31} = 0$, $x_{32} = 1$), the dynamic of the constraints used to represent the cost contract unfolds as follows. Given that that the total project cost equals 8 (see table 5.2), the leftmost region contains the cost outcome. Hence, both the α_1^C and β_1^C variables equal zero - implying that the cost is neither greater than the upper bound nor smaller than the lower bound of the region. Hence, looking at the set of equations 5.35 until 5.40 it can be seen that only equations 5.35 and 5.36 are binding. Hence, the awarded incentive amount for the first region equals $I_1^C = s_1^C(B_2^C - \sum_{i=1}^{|V|}\sum_{m=1}^{M_i} C_{im} \cdot x_{im}) = 0.25 \cdot (10 - 8) = 0.5$. Similarly, for the middle region ($r = 2$), it can be seen that the cost outcome falls below the lower bound B_2^C of the region, making $\beta_2^C = 1$. This in turn ensures that equations 5.27 and 5.28 are binding. Hence, the awarded incentive amount for the second region can be calculated as: $I_2^C = s_2^C(C^t - B_2^C) = 0.5 \cdot (14 - 10) = 2$. Following the same logic, it can be deduced that for the the third region only equations 5.33 and 5.34 are binding, indicating that no incentive amount is earned within this region.

x_{21}	x_{22}	x_{31}	x_{32}	I^C	I^D	I^S	I^{tot}	Obj
1	0	1	0	-0.75	1.5	0.75	1.5	-7.5
1	0	0	1	0.5	1.5	-5.25	-3.25	-10.25
0	1	1	0	0.5	-1.5	4.5	3.5	-3.5
0	1	0	1	2.5	-1.5	-1.5	-0.5	-5.5

Table 5.4: Possible outcomes.

The calculation of the incentive amounts for the other dimensions are trivial, resulting in the total incentive amounts shown in table 5.3. By subtracting the total cost of effort ($\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} E_{im}^{cost} \cdot c_{im}$) from the total incentive the objective function of the optimisation can be obtained (see equation 5.7). Hence, it is clear that the optimal solution for this example

[3]Or in other words, the set of equations is added for $r = 1$.

project is to execute activity 2 in mode 2 ($x_{22} = 1$) and activity 3 in mode 1 ($x_{31} = 1$), resulting in a total return of -3.5.

5.5 Data Generation

The goal of the data generation procedure is to create a comprehensive set of multi-mode scheduling problems, including multi-dimensional incentive contracts. The method used to do this consists of three phases: (*i*) generation of the set of possible contract parameter combinations, (*ii*) defining the multi-mode activities that comprise the project, and (*iii*) linking the contract parameters to a specific project by adjusting the relevant contract parameters.

Component	Parameter	Meaning	Value(s)
Contract	C^t, D^t, S^t	Target value	$\frac{C_{\max}-C_{\min}}{2}$, $\frac{D_{\max}-D_{\min}}{2}$, $\frac{S_{\max}-S_{\min}}{2}$
	B_r^C, B_r^D, B_r^S	Region lower bound	Evenly spaced
	$r_{\max}$	Maximal # regions	3
	s_r^C	Sharing ratio	$\{0, 0.25, 0.5, 0.75, 1\}$
	v_r^D	Duration region valuation	$\{0, 0.25, 0.5, 0.75, 1\} \cdot \theta^D$
	v_r^S	Scope region valuation	$\{0, 0.25, 0.5, 0.75, 1\} \cdot \theta^S$
Trade-offs	$\|V\|-2$	# non-dummy activities	$\{30, 60, 90\}$
	SP	Serial/parallel network indicator	$\{0.16, 0.33, 0.5, 0.66, 0.83\}$
	$\Delta D, \Delta S, \Delta E$	Dimension Range	$Tri(\frac{c}{2}, \frac{3c}{2}, c) \vert c = U(1, 10)$
	n	Number of points in range	11
	S_D, S_S, S_E	Slope of the t/o curve	$\upsilon(0.5, 1.5)$
	m_a^b	Interaction multiplicators	$\upsilon(0.0, 0.4)$
	$C_{\min}$	Minimal cost	$\Delta C \cdot \upsilon(2, 10)$
	CM_D, CM_S, CM_E	Convexity magnitude	$\upsilon(0, 0.25)$
	M_i	# modes of act i	$U(1, 10)$

Table 5.5: Parameters used to generate the dataset. ($Tri()$: Triangular distribution, $U()$: discrete uniform distribution, $u()$: continuous uniform distribution.)

Table 5.5 presents an overview of the parameters that are used in the data generation procedure. The nature of these parameters and their role in the data generation procedure are described in the sections below.

5.5.1 Contract Parameters

In the first phase of the data generation procedure a set of contract clauses for each of the outcome dimensions (cost, duration and scope) is defined. To limit the number of possible contract forms, a number of assumptions has been made regarding the structure of the contracts. Firstly, it was assumed that the segments of piecewise linear contracts are always evenly spread across the attainable range of the respective outcome dimension (e.g. if the maximal and minimal duration of a project are 200 and 100 days respectively, a piecewise

linear cost clause with two segments will be structured as follows: $]100, 150]$ and $]150, 200[$.) The second assumption is that the target value specified in the contract (C^t, D^T, S^t) always lies exactly in the center of the attainable region (e.g. for the contract described earlier $D^t = 150$). Naturally, these two assumptions decrease the diversity of contract clauses considered, but nevertheless the majority of contracts studied in the literature (see chapter 2 for a more extensive overview of literature) use structures that approximate this basic form. By taking these assumptions, the parameters for the generation procedure are reduced to the maximal number of piecewise linear segments ($r_{\max}$), and the values used for the sharing ratios (s_r^C) or region valuations (v_r^D and v_r^S).

As seen in table 5.5, the maximal number of segments for the piecewise linear contracts is set at 3. This implies that clauses including 1, 2 or 3 segments are included in the dataset. The sharing ratios and region valuations that specify the magnitude of the incentive given to the contractor are selected from a list of values (see table 5.5). For the cost clause sharing ratios this is simply a fraction of the cost incurred, however for the duration and scope contracts these fractions are multiplied by the owner's valuation of an unit of time (θ^D) and scope (θ^S) respectively[4]. For each dimension all possible combinations of these values are included in the dataset, meaning that the total number of unique contract clauses for a dimension can be calculated as:

$$\eta = \sum_{r=1}^{r_{\max}} \left[|s| \cdot (|s|-1)^{r-1} \right] = \sum_{r=1}^{3} \left[5 \cdot 4^{r-1} \right] = 105 \tag{5.41}$$

Where η is equal to the number of unique contract clauses, $r_{\max}$ equals the maximum number of regions allowed, and $|s|$ represents the number of sharing ratio values. Note that piecewise linear contracts that have identical sharing ratios in consecutive regions are not included. The reasoning being that these contracts do not have the correct number of piecewise linear segments since the consecutive regions with identical sharing ratios can be considered to be a single segment. This can be illustrated using a simple example: consider a cost incentive clause with two segments, if both segments of this contracts have a sharing ratio of 25% the contract is identical to a cost incentive clause with a single segment and a sharing ratio of 25%.

The contract clauses for the individual dimensions (cost, duration and scope) are then merged into a contract by means of random combinations. The size of the dataset can be increased by including each contract clause multiple times in the combination procedure. For the dataset used in this research, each contract clause has been included 10 times, resulting in a total of 1,050 contracts.

[4]The values for the θ^D and θ^S parameters are defined based on the properties of the project, see section 5.5.3.

5.5.2 Project with Multi-mode Activities

The second step in the data generation procedure is the creation of the project data. As was stated in section 5.3.1 this comprises a set of interrelated activities, each of which has a number of discrete modes in which the activity can be executed. Figure 5.3 gives a high-level overview of the approach used to create diverse projects based on realistic assumptions regarding the interrelations of the trade-off dimensions. An overview of the parameters and distributions used in this generation procedure can be found in table 5.5.

First, a detailed overview of the methodology used to generate this data is presented in section 5.5.2.1. Because this methodology is quite complex, the mechanism is then illustrated using a simplified example in section 5.5.2.2.

5.5.2.1 Methodology

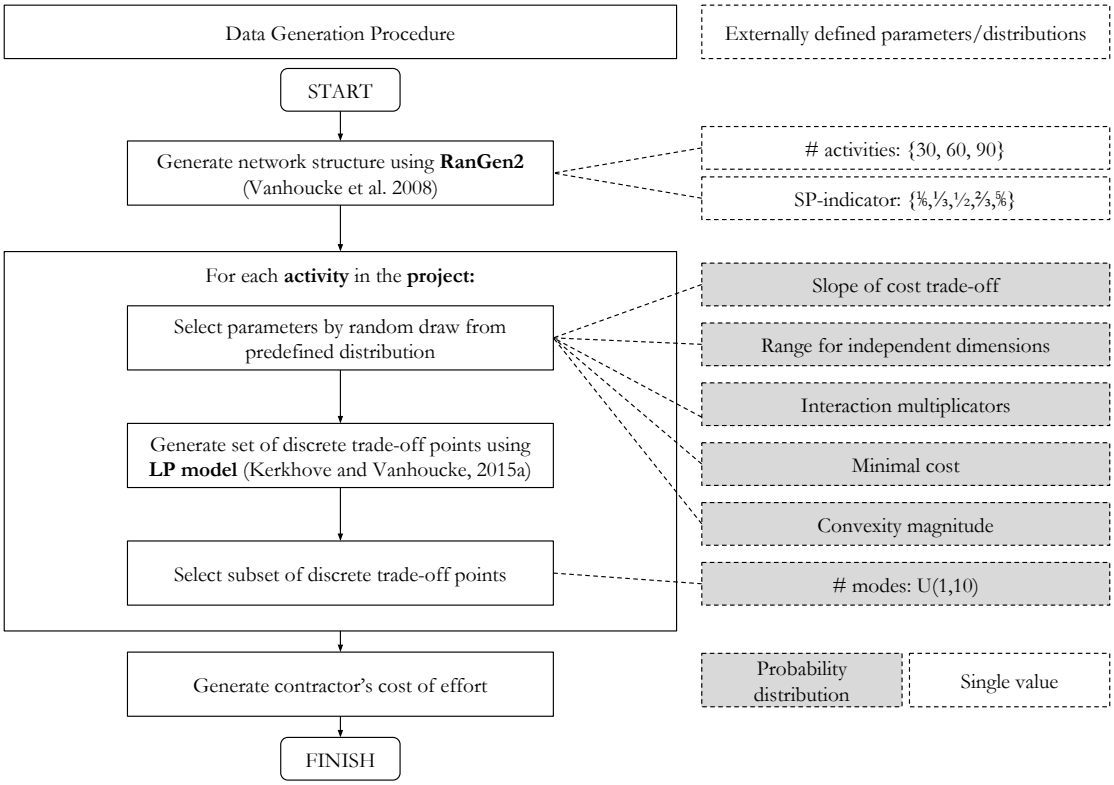


Figure 5.3: Generation procedure for projects with multi-mode options.

The generation procedures consists of two components, the creation of the project structure and the creation of the execution modes for the individual activities. For the former step the *RanGen2* tool created by Vanhoucke et al. (2008) is used in order to

guarantee a topologically diverse set of projects. The second step in the procedure iterates over each individual activity of the project and creates a number of realistic execution modes. This is done by adapting the methodology introduced in chapter 3 to work on the level of individual activities rather than on the project level.

The *RanGen2* network generator requires two input parameters in order to create a project network: the number of activities and the serial parallel indicator (Tavares et al., 1999, 2002; Vanhoucke et al., 2008). The latter is an indicator of where the project is situated in the range between a completely serial project with a single chain of activities and a completely parallel project where each activity is only connected to the start and finish dummy activities. A total of 3,150 networks have been generated for the computational experiments (section 5.7). This comprises 1,050 projects with 30, 60 and 90 non-dummy activities respectively. The SP-indicator for these projects was set to one of six values: $\{0.16, 0.33, 0.5, 0.66, 0.83\}$ (e.g. 175 projects with 30 activities and $SP = .16$).

Once the project structure has been created, the data generation procedure iterates over all the activities to define a number of discrete execution modes. The methodology introduced in chapter 3 uses a linear programming model in combination with a number of user-defined parameters to define a discrete set of execution strategies for the project, which satisfy six axioms defining the interrelation between the trade-off dimensions (cost, duration, scope and effort). These axioms and the model in general define the cost as the dependent variable, i.e. the cost is a function of the duration, scope and effort[5].

The axioms to which the trade-off points have to adhere describe the relation between the individual independent dimensions (duration, scope and effort) and the dependent dimension (cost), as well as the interrelation between the independent dimensions. The former can be summarised as follows: "The direct cost is a non-increasing convex function of both the duration and contractor effort. Similarly, the cost is a non-decreasing convex function of the scope of an activity." The interplay between the dependent dimensions is modelled as an influence on their relation to the dependent dimension. For example the interplay between the duration and scope dimensions can be summarised as follows: "Shortening the duration of a high-scope activity results in a larger or equal cost increase than shortening the duration of a low-scope activity. Similarly, a scope increase will be at least as costly when the activity is executed in a shorter time-frame." Section 3.4.1 gives an overview of the six individual axioms. The next paragraphs describe how the project-based approach was adjusted for use on the level of individual activities within a project.

As shown in figure 5.3, first step on the level of the individual activities is to define

[5]This is merely a design choice, it is equally feasible to use any of the other dimensions as the dependent variable.

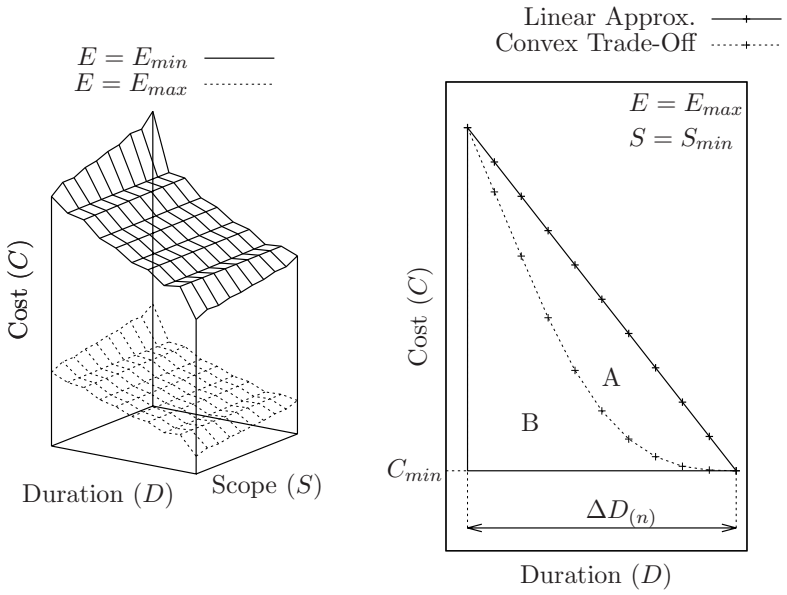


Figure 5.4: Generation of discrete trade-off points: a multidimensional representation of generated trade off points (left) and a subset of the generated points (right).

values for the parameters needed for the generation of discrete trade-off points. Figure 5.4 illustrates the nature of these trade-off points. The left panel of this figure shows a subset of the four-dimensional trade-off points in a three-dimensional space. The right panel shows a subset of these points and will be used to explain the meaning of the parameters listed in table 5.5. A more extensive discussion of these parameters can be found in chapter 3.

The first parameter is the range over which the value of the independent dimensions can vary (ΔD, ΔS, ΔE). The right panel of figure 5.4 illustrates the available range for the duration dimension. Within this range n equidistant points are defined to represent the discrete choices available for the independent dimension. As shown in table 5.5 the size of these ranges is drawn from a triangular distribution. The parameters of this triangular distribution are expressed in terms of another random variable c which can have any value in the range $\{1, ..., 10\}$. This random variable c is an abstract representation of the 'size' of an activity. As such, larger activities will typically have longer durations, larger scope and more potential for effort investments. A consequence of this is that larger activities will also have larger costs on average. This is one of the key differences between the project-based approach presented in chapter 3 and the activity-based trade-offs used in this research. The number of discrete options available in these ranges (n) in set to 11 for each of the three independent dimensions. This results in a total of 1,331 ($n^3 = 11^3$)

possible combinations of the independent dimensions, each of which is associated with a cost.

The range for the independent dimensions (ΔD, ΔS, ΔE) is converted to discrete options by multiplying the lowest and highest cost options, indexed 1 and n respectively, with these ranges. The base values for these options are as follows: $D_1 = 2$, $D_n = 1$, $S_1 = 1$, $S_n = 2$, $E_1 = 1$, and $E_n = 0$. This reflects that the cost increases when the duration or the effort decreases, or when the scope increases. As such the highest and lowest values for the independent dimensions are equal to: $[1 \cdot \Delta D; 2 \cdot \Delta D]$, $[1 \cdot \Delta S; 2 \cdot \Delta S]$ and $[0; 1 \cdot \Delta E]$.

Determining the cost for each of these discrete execution modes starts by defining an approximate linear relationship between the independent dimensions and the dependent (cost) dimension, as shown in the right panel of figure 5.4. The slope of this linear approximation is determined by the slope parameters (S_D, S_S, S_E) and the multiplicators used to represent the interaction between the independent dimensions[6].

A mathematical expression for the approximate relation between the cost and the independent variables is given by equation 5.42. In this expression the independent duration, scope and effort dimensions are indexed using i, j and k respectively. These indices run from 1 to n where n is the option resulting in the highest possible cost (i.e. the shortest duration, highest scope and lowest effort respectively.) Equation 5.42 shows how the linear approximation of the total cost is modelled as a function of the independent dimensions. The impact of these dimensions depends on both the slope (S_D, S_S, S_E) and the values of the other parameters, the latter being quantified by the m_a^b parameters.

$$\begin{aligned} C_{ijk} =& C_{\min} + |S_D| \left(m_D^S \frac{\Delta S_j}{\Delta S_n} + 1 \right) \left(m_D^E \frac{\Delta E_k}{\Delta E_n} + 1 \right) \Delta D_i + \\ & |S_S| \left(m_S^D \frac{\Delta D_i}{\Delta D_n} + 1 \right) \left(m_S^E \frac{\Delta E_k}{\Delta E_n} + 1 \right) \Delta S_j + \\ & |S_E| \left(m_E^D \frac{\Delta D_i}{\Delta D_n} + 1 \right) \left(m_E^S \frac{\Delta S_j}{\Delta S_n} + 1 \right) \Delta E_k \end{aligned} \tag{5.42}$$

The solid line in the right panel of figure 5.4 shows a subset of these points where the scope and effort have been assigned their lowest cost options[7]. As such the lowest point of the line is the lowest possible cost (C_{min}). The linear relation then shows how a decrease in the duration of the activity inflates the cost of the activity.

The realism of this relation is improved by modelling a convex rather than linear

[6] m_a^b where b and a are both independent dimensions and m_a^b represents the increase in the slope of the relation between the cost and the independent dimension a when dimension b is at its most expensive value, as opposed to its lowest cost setting. The accrual of slope between these extrema of b is assumed to be linear.

[7] Note that the highest effort of the contractor results in the lowest cost for the owner.

relationship, as stated by the axioms defining the interrelationships between the various dimensions. The degree of convexity is defined as the convexity magnitude CM (Kerkhove and Vanhoucke, 2015a). The interpretation of this metric is illustrated on the right panel of figure 5.4. Within the triangle formed by the linear approximation and the projection of its extreme points towards the axes two zones can be distinguished. Zone A is the area between the linear approximation and the convex curve and zone B being the remainder of the triangle. Using these two areas the convexity magnitude CM^D (for the duration dimension) is defined as: $\frac{A}{A+B}$. The same linear model as was used in chapter 3 is used to create a smooth convex curve which satisfied the specified convexity for the respective dimensions. The combination of all these curves results in a set of points within a four-dimensional space.

At this point a large number (1,331) of discrete trade-off points have been created for the activity. Whereas such a wide range of possible modes may be realistic on a project level (because of the large number of potential combinations of activity modes), the number of ways in which an individual activity may be executed is generally more limited. Hence, rather than making all possible execution modes available, a random subset of these points is selected. The number of modes M_i for activity i is drawn from a discrete uniform distribution $U(1, 10)$. The modes which are used are selected at random from the set of 1,331 generated modes.

As a final step in the generation procedure for the multi-mode activities, the monetary cost of effort to the contractor is determined. This is done by rescaling the maximal total cost of effort in relative terms to the total cost variability of the project ($\Delta C = C_{\max} - C_{\min}$)[8]

$$E_{\max}^{cost} = \upsilon(0.1, 0.9) \cdot \Delta C \tag{5.43}$$

where $E_{\max}^{cost}$ represents the maximal cost of effort for the complete project. The effort cost associated with a specific activity mode can then be calculated as follows:

$$E_{im}^{cost} = E_{im} \cdot \frac{E_{\max}^{cost}}{E_{\max}} \quad \forall i, m \tag{5.44}$$

where E_{im}^{cost} is the effort cost associated with mode m of activity i, E_{im} is the effort expressed on a ratio scale as was originally generated, and $E_{\max}$ is the maximal possible effort for the total project, which is calculated as the sum of the maximal effort option of all activities.

[8]Where $C_{\min}$ is a parameter and $C_{\max}$ is obtained by selecting the highest cost mode for each activity.

5.5.2.2 Example

The first step in the creation of a multi-mode project is the generation of the project network. Assume that the number of non-dummy activities ($|V|-2$) is set to 2 and the serial-parallel indicator is set equal to 0. This would result in a network as shown in figure 5.2 where two non-dummy activities are executed in parallel.

Next, the discrete execution modes are generated for the first non-dummy activity (indexed 2 on figure 5.2). By means of a random draw, the c parameter which reflects the general size of the activity is set to 7. Hence, the range for the independent dimensions (ΔD, ΔS, ΔE) are drawn from a triangular distribution $Tri(3.5, 10.5, 7)$.

A random draw from this triangular distribution results a value of 6 for ΔD, 5 for ΔS and 8 for ΔE. This range is then converted to a linear scale. For the duration and scope this is done by multiplying the range with the 1 and 2 to obtain the extreme points, based on the assumption that an activity will always have a non-zero duration and scope. The bounds of the effort dimension are obtained by multiplying the value of the range by 0 and 1, the reasoning here being that the project can be executed without execution any extra effort. Hence, the following ranges are created for the duration, scope and effort dimensions respectively: $[6, 12]$, $[5, 10]$ and $[0, 8]$. For this example 3 rather than 11 discrete and equidistant options are created within these ranges. As such the duration, scope and effort are selected from the following sets: $\{6, 9, 12\}$, $\{5, 7.5, 10\}$ and $\{0, 4, 8\}$. This results in $3^3 = 27$ total possible execution modes for the activity, each of which has to be associated with a specific cost.

The first step to associate a specific cost with each of these execution modes is the creation of a linear approximation. This linear approximation is based on the slope parameters (S_D, S_S, S_E), the inter-dimensional multiplicators (m_D^S, ...) and the lowest possible cost $C_{\min}$, as shown in equation 5.42. Because the minimal cost depends on the value of the range over which the cost varies (ΔC, see table 5.5), the incremental costs are calculated first. To do this, the following parameters are drawn from their respective random distributions: $S_D = 0.5$, $S_S = 1$, $S_E = 1.5$, $m_D^S = 0.2$, $m_D^E = 0.2$, $m_S^D = 0.1$, $m_S^E = 0.1$, $m_E^D = 0.2$ and $m_E^S = 0.1$.

In the following paragraphs these parameters will be used to determine the marginal cost for the linear approximation, the total cost for the linear approximation and the total convex cost. The results for these costs are shown in the final three columns of table 5.6 and are denoted as 'ΔC lin', 'C lin' and 'C conv' respectively.

Using these parameters the incremental cost ΔC_{ijk} can be calculated based on equation 5.42[9]. Table 5.6 shows these incremental costs for each of the 27 possible scenarios (ΔC

[9] $\Delta C_{ijk} = C_{ijk} - C_{min}$

nb	D	S	E	ΔC **lin**	C **lin**	C **conv**
1	12.0	5.0	8.0	0.0	131.1	131.1
2	12.0	5.0	4.0	6.0	137.1	135.9
3	12.0	5.0	0.0	12.0	143.1	143.1
4	12.0	7.5	8.0	2.5	133.6	133.1
5	12.0	7.5	4.0	8.9	140.0	138.2
6	12.0	7.5	0.0	15.4	146.4	145.9
7	12.0	10.0	8.0	5.0	136.1	136.1
8	12.0	10.0	4.0	11.9	142.9	141.6
9	12.0	10.0	0.0	18.7	149.8	149.8
10	9.0	5.0	8.0	1.5	132.6	132.0
11	9.0	5.0	4.0	8.3	139.3	137.4
12	9.0	5.0	0.0	15.0	146.1	145.4
13	9.0	7.5	8.0	4.3	135.4	134.2
14	9.0	7.5	4.0	11.5	142.6	139.9
15	9.0	7.5	0.0	18.7	149.8	148.5
16	9.0	10.0	8.0	7.0	138.2	137.4
17	9.0	10.0	4.0	14.8	145.9	143.6
18	9.0	10.0	0.0	22.5	153.6	152.7
19	6.0	5.0	8.0	3.0	134.1	134.1
20	6.0	5.0	4.0	10.5	141.6	140.2
21	6.0	5.0	0.0	18.0	149.1	149.1
22	6.0	7.5	8.0	6.1	137.2	136.6
23	6.0	7.5	4.0	14.1	145.2	143.1
24	6.0	7.5	0.0	22.1	153.2	152.6
25	6.0	10.0	8.0	9.1	140.2	140.2
26	6.0	10.0	4.0	17.7	148.8	147.2
27	6.0	10.0	0.0	26.2	157.3	157.3

Table 5.6: Generated linear and convex costs.

lin). From these values it is clear that the total range over which the cost varies (ΔC) equals 26.2 (the difference between the highest and lowest value of ΔC lin). Hence, the minimal cost value $C_{\min}$ will lie somewhere in the range $[52.4; 261.0]$ (see table 5.5). For this example a random value of 131.1 has been drawn, resulting the values for the total cost for the activity modes as shown in the C lin column of table 5.6.

Next, the linear approximation has to be transformed into a convex relationship. The degree of convexity is defined as the convexity magnitude, which is defined for each of the independent dimensions (CM^D, CM^S and CM^E respectively, see section 5.5.2.1 for the definition of this metric). The values for these parameters are drawn from an uniform distribution (see table 5.5). For this example the following parameters are used: $CM^D = 0.2$, $CM^S = 0.1$ and $CM^E = 0.1$.

The convexity is inserted by considering the impact on the cost of each dimension separately, for each possible combination of values of the other two independent dimensions. An example for the duration dimension is given by figure 5.5. The horizontal axis on this figure contains the three possible values of the duration dimension, and the vertical axis shows the relative cost increase when compared to the minimal cost ($C_{\min}$). The convex relation is obtained by means of a downward shift of the point(s) in between the highest and lowest value of the independent dimension. For the example in figure 5.5 this means decreasing the cost associated with D_2 from ΔC_{2jk} to $\Delta C'_{2jk}$. This analysis has to be made for every possible combination of the j and k indices to determine the convexity of the relation between the duration and the cost. An analogous procedure has to be followed for the scope and effort dimensions.

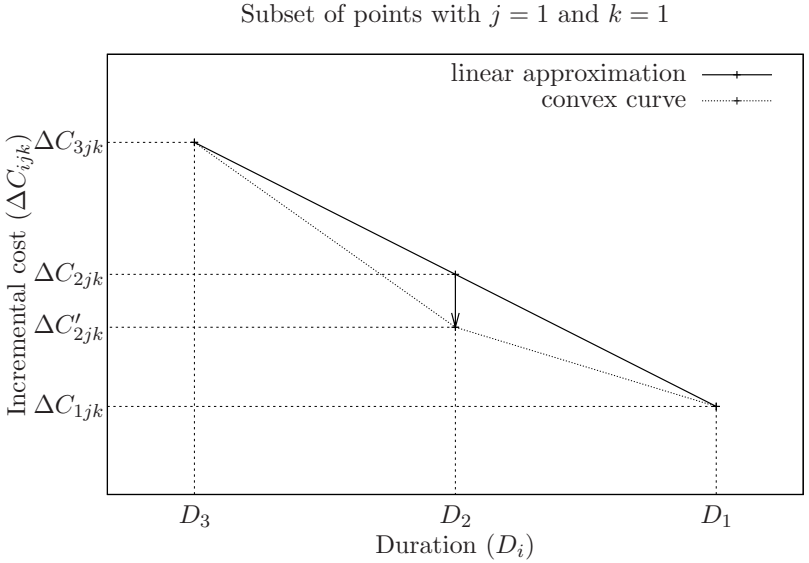


Figure 5.5: Creation of the convex relation between duration and cost.

Because only three possible values are considered in this example, it is straightforward to calculate the appropriate cost to associate with D_2 algebraically. The simplest way to do this is to calculate the surface of the complete triangle ($A + B$, equation 5.45) and the surface B which can be viewed as a combination of two triangles and a (equation 5.46). Using these expressions it is easy to reconstruct the formula for the convexity magnitude ($CM^D = A/(A + B)$). Solving this expression for $\Delta C'_{2jk}$ yields the sole possible value for the cost which satisfies the proposed convexity, as shown by equation 5.47. Similar expressions can also be determined for the scope and effort dimensions (equations 5.48

and 5.49 respectively)[10].

$$A + B = \frac{(\Delta C_{3jk} - \Delta C_{1jk})(D_1 - D_3)}{2} \tag{5.45}$$

$$B = \frac{(\Delta C_{3jk} - \Delta C'_{2jk})(D_2 - D_3)}{2} + (\Delta C'_{2jk} - \Delta C_{1jk})(D_2 - D_3) + \frac{(\Delta C'_{2jk} - \Delta C_{1jk})(D_1 - D_2)}{2} \tag{5.46}$$

$$\Delta C'_{2jk} = \frac{\Delta C_{1jk}(CM^D(D_1 - D_3) + D_2 - D_3)}{D_1 - D_3} + \frac{\Delta C_{3jk}(CM^D(D_3 - D_1) + D_1 - D_2)}{D_1 - D_3} \tag{5.47}$$

$$\Delta C'_{i2k} = \frac{\Delta C_{i1k}(CM^S(S_1 - S_3) + S_2 - S_3)}{S_1 - S_3} + \frac{\Delta C_{i3k}(CM^S(S_3 - S_1) + S_1 - S_2)}{S_1 - S_3} \tag{5.48}$$

$$\Delta C'_{ij2} = \frac{\Delta C_{ij1}(CM^E(E_1 - E_3) + E_2 - E_3)}{E_1 - E_3} + \frac{\Delta C_{ij3}(CM^E(E_3 - E_1) + E_1 - E_2)}{E_1 - E_3} \tag{5.49}$$

For the 10-th possible permutation (see table 5.6) only the impact of the cost has to be adjusted since both the scope and effort are at their lowest cost levels[11]. The values for ΔC_{1jk} and ΔC_{3jk} can be found by locating the scenarios where the scope and effort have the same value. For this scenario this is scenario 1 and 19 respectively. This means that ΔC_{1jk} and ΔC_{3jk} get the values 0 and 3 respectively. In combination with the three possible values for the duration dimension ($D_1 = 12$, $D_2 = 9$ and $D_3 = 6$) this allows formula 5.47 to be filled out. This results in a value of 0.9 for $\Delta C'_{2jk}$, which is lower than

[10] It is important to note that these equations are only valid when three possible values are used for the dependent dimensions. For the actual data a mixed integer programming model was used (see chapter 3

[11] As shown in figure 5.5 only the middle point has to be adjusted in this example.

the original value of 1.5. Effectively, the second term of equation 5.42 is replaced by this new value. As shown in the table this decreases the linear cost for the 10-th permutation from 132.6 to 132.0, the latter being the value of C conv for the 10th permutation. For all other modes a similar logic is followed in order to obtain all the values in the final column of table 5.6.

The process now continues by selecting between one and ten execution modes at random for this activity, all others being discarded. Next, the process is repeated for the second activity in the project network. As such, a number of discrete execution modes is obtained for both activities.

5.5.3 Adjust Contract Parameters to Project

The third and final step in the data generation procedure is the combination of the contract structures and the generated multi-mode projects. To do this, the parameters of the contract have to be adjusted to the properties of the project. As shown in table 5.5, the contracts generated in section 5.5.1 as defined using relative ranges. Hence, the first step in adjusting the contract parameters is to calculate the extreme outcomes for each of the three outcome dimensions perceived by the project owner. For the cost and scope dimension the minimal ($C_{\min}$, $S_{\min}$) and maximal ($C_{\max}$, $S_{\max}$) values can be calculated by simply taking the summation of the minimal and maximal modes for the respective dimension for each activity. Calculating the minimal and maximal project duration is slightly more complex and requires the critical path to be calculated when the minimal and maximal duration modes are selected for each individual activity.

For the cost dimension of the contract, the alignment is limited to setting the target cost equal to the middle of the attainable range ($(C_{\max} - C_{\min})/2$), and spreading the bounds evenly across the range as explained in section 5.5.1. Both the duration and scope dimensions also require determining the owner's monetary valuation per unit of time and scope (θ^D, θ^S). The values for these parameters are defined by specifying the importance of the duration and scope dimensions in relative terms to the monetary impact of the cost dimension, and then rescaling this quantity per unit of time and scope respectively:

$$\theta^D = \frac{\upsilon(0.5, 2.0) * \Delta C}{\Delta D} \tag{5.50}$$

$$\theta^S = \frac{\upsilon(0.5, 2.0) * \Delta C}{\Delta S} \tag{5.51}$$

These values are then used to calculate the incentive amounts awarded per unit of time or scope in the various regions, as is explained in section 5.5.1. The adjustment of the targets and the bounds for the contracts is done in a similar manner.

5.6 Solution Methodology

Two solution methods are compared in this chapter. The first is simply an implementation of the mixed integer programming model presented in section 5.3.3. The performance of this solution technique is compared to the performance of a greedy local search that can be compared to a 'rational' strategy followed by a contractor when faced with the optimisation problem. By comparing these two approaches in section 5.7 the added value of the exact solution approach, as well as the impact of project and contract structure on potential contractor earnings is investigated.

5.6.1 Commercial Solver Software

The model that has been presented in section 5.3.3 has been implemented using Gurobi 6.0 using the C++ interface. The experiments have all been carried out on a dual core 2.4GHz computer with 16GB of RAM.

5.6.2 Greedy Local Search

A greedy local search for the problem is presented in this section. This greedy local search follows a logical progression starting from the solution requiring the least effort, and moves on to solutions requiring incrementally greater effort stopping when the next move does not yield a positive change in the objective function value. A pseudo-code representation of this heuristic is given by algorithm 5. In this algorithm χ is used to represent the complete set of x_{im} variables, i.e. a solution to the multi-mode scheduling problem.

Algorithm 5 Greedy local search

```
1: χ ← x_{im'} = 1  ∀i ∈ V,  m' : E_{im'} ≤ E_{im} ∀m ∈ {1, ..., M_i}
2: ι ← {1, ..., |V|}
3: while objective for χ has improved do
4:     ι.randomShuffle()
5:     for i in ι do
6:         improved ← true
7:         while improved = true do
8:             improved ← false
9:             χ' ← χ| lowest increment of effort for act i
10:            if χ'.objective() > χ.objective() then
11:                χ ← χ'
12:                improved ← true
13: return χ^best
```

The algorithm starts by selecting the mode with the lowest effort for all of the activities in the project (line 1). Any ties are broken arbitrarily. Next, an array containing the indices for all the activities (ι) is defined (line 2). The procedure then starts the improvement loop (line 3). Within this improvement loop, the procedure iterates over all the activities in

random order (line 4 and 5). For each activity, the activity mode with the lowest increase in effort is considered and as long as the preceding move has yielded an improvement the next increment for the activity is tested. Naturally, this process stops when the mode with the highest amount of effort is reached. This heuristic corresponds to a 'lowest hanging fruit' mindset which is likely to be followed or approximated by practitioners.

5.7 Computational Experiments

Both the exact and heuristic solution techniques have been tested on the dataset described in section 5.5. All of these instances were solved to optimality by the exact approach. The results of these experiments will be discussed in this section. Specifically, the impact of the problem size (section 5.7.1), the network structure (section 5.7.2) and the contract structure (section 5.7.3) on the performance of the algorithms will be discussed in detail.

5.7.1 Impact of the Problem Size

Figure 5.6 illustrates the impact of the problem size, as measured by the number of activities, on the objective function value and the computation time of both solution procedures. The objective function value is shown on the left axis, and the optimality gap for the heuristic approximation is shown on the right axis. This optimality gap is calculated as: $1 - \frac{OF_{heuristic}}{OF_{exact}}$.

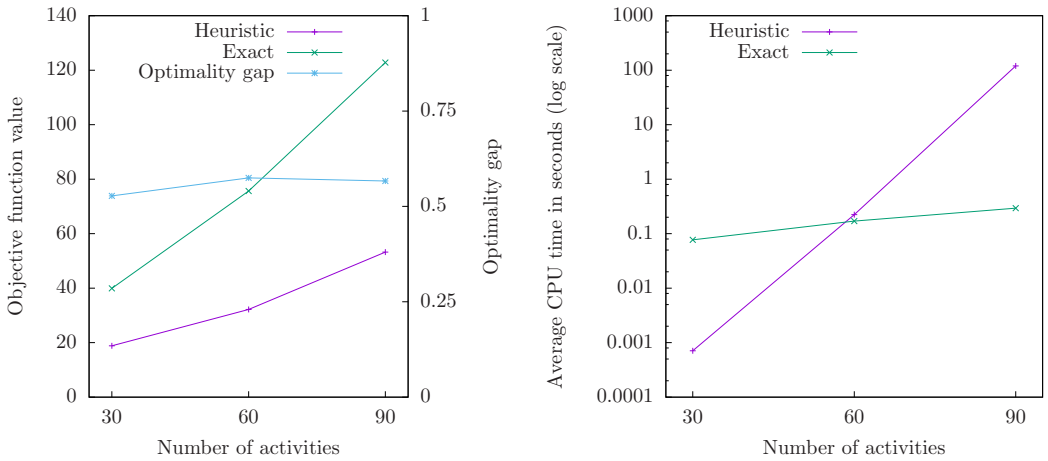


Figure 5.6: Influence of the problem size on the solution quality and speed.

Naturally the optimal objective function value increases as the size of the project increases since a larger amount of work warrants a higher payment. When comparing the

exact model to the heuristic solution procedure, it is clear that the absolute gap between the two solutions increases as the size of the project increases. The relative performance (as indicated by the optimality gap) of the heuristic approach remains quasi constant across all possible project sizes.

Looking at the second panel of figure 5.6 reveals that the computation time needed for the exact solution approach increases much faster for the heuristic solution approach than for the exact solution approach. This leads to the conclusion that the effort needed to heuristically optimise the project execution increases exponentially with the problem size (note the exponential scale use for the chart on the right). Moreover, this also indicates that the mixed integer programming formulation is highly efficient since the increase in project size has only a limited impact on the CPU time needed to find the optimal solution.

5.7.2 Impact of the Network Structure

This section discusses the influence of the network structure on the performance of the exact and heuristic solution methods. The network topology is defined using six network topology indicators:

Coefficient of network complexity: A simple measure of the complexity of an activity on the node network expressed as the number of direct arcs divided by the number of nodes. A higher value for this indicator implies that the network is more complex since there is a higher number of precedence relations per activity (Davis, 1975; Pascoe, 1966).

Order strength: A measure for the number of precedence relations in the network structure. The OS is defined as the number of precedence relationships (including transitive relationships) divided by the total number of possible precedence relationships in an acyclic activity on the node network ($\frac{n(n-1)}{2}$). This metric falls within the range $[0, 1]$, a larger number implying more relationships and therefore a more complex network structure (Herroelen and De Reyck, 1999; Mastor, 1970).

Serial parallel indicator: Indicates to what degree a network resembles a completely serial (1) or parallel (0) network structure[12] (Tavares, 1999; Tavares et al., 2002; Vanhoucke et al., 2008).

Activity distribution: This metric gives insight in how the activities are distributed within the project. This is done by categorising the activities into levels based on the maximal number of arcs which separate them from the dummy start node. Next, the distribution of activities among these levels is measured and expressed in

[12]Note that this indicator is used as an input parameter for the network generation tool used in this research, hence the specific values used are shown in figure 5.2, rather than the ranges for the values as was done for the other topological indicators.

the range $[0, 1]$. A value of 0 implies that all levels contain an identical number of activities, whereas a value of 1 implies that one level contains the maximal number of activities and all other levels contain a single activity (Tavares, 1999; Tavares et al., 2002; Vanhoucke et al., 2008).

Length of arcs: This metric also uses the same categorisation of activities in levels depending on the maximal number of arcs separating them from the starting node. The LA metric measures the length of the precedence relationships as measured by the number of levels these precedence relationships span. When $LA = 1$ all precedence relationships have a length of 1, meaning that the successors of all activities are on the adjacent level. Inversely, as LA approaches 0, there are many precedence relations spanning a large number of levels (Tavares, 1999; Tavares et al., 2002; Vanhoucke et al., 2008).

Topological float: Measures the degree to which activities can be shifted to other levels without violating the maximal level of the network. Again this metric is contained within the range $[0, 1]$. A value of 0 implies that the network is 100% dense and no activities can be shifted. A value of 1 indicates that the network consists of a single chain of activities with the remainder of activities scheduled parallel to this chain. Hence, all activities scheduled in parallel to the chain of activities have a substantial amount of float (Tavares, 1999; Tavares et al., 2002; Vanhoucke et al., 2008).

These indicators are frequently used in project management research and their specific formulae will not be explained here. For a more detailed explanation on the calculation of these topological indicators, the reader is referred to Vanhoucke (2010a).

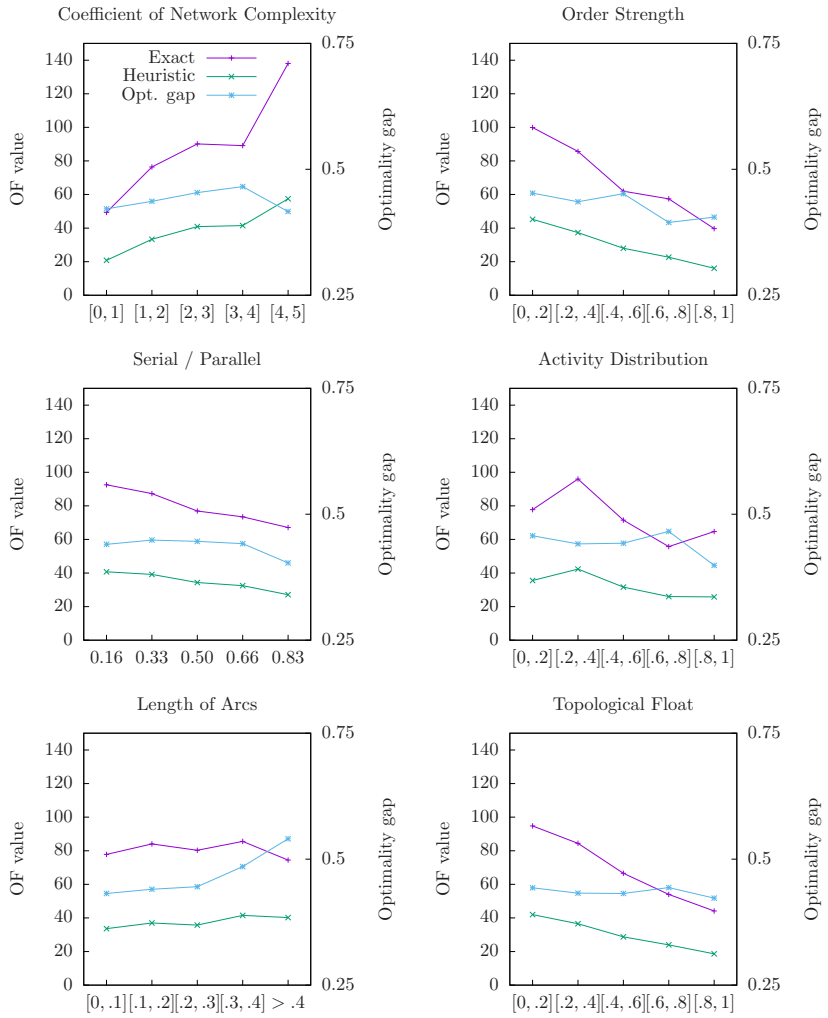


Figure 5.7: Influence of the network structure on the absolute and relative solution quality.

Figure 5.7 shows that the optimal outcome of the project is significantly influenced by the network topology. The relative performance of the heuristic solution method is also affected by the network structure, yet always remains significantly below the optimal value. This implies that optimal solution methods should be preferred regardless of the structure of the network. The specific influences of network structure on the contractor's profits can be summarised as follows:

- An increase in the **coefficient of network complexity** results in higher objective function values. This somewhat counter-intuitive observation can be explained by the significant positive correlation (0.365) between the problem size and the network complexity indicator. Hence, the increase in objective function value is largely due to the

increase in problem size. As noted in Vanhoucke (2010a), such issues are not uncommon when using the CNC metric. The relative performance remains markedly stable.

- The impact of the **order strength** is more intuitive: as the relative amount of precedence relations increases, the optimal objective decreases. The reason for this being that making efficient alterations in the duration of the project becomes increasingly difficult as the number of precedence relations increases. This also adds to the complexity of finding the optimal solution, as is reflected by the decreasing relative performance of the heuristic solution procedure.
- The observations for the **SP-indicator** indicate that a higher profit for the contractor can be obtained for more parallel networks (i.e. lower values for the SP indicator). The relative performance is more or less constant, with the exception of the most serial networks where a dip in the relative performance of the heuristic solution procedure can be observed.
- The impact of the **activity distribution** is rather ambiguous, no distinctive trend can be observed in these results.
- There is no consistent trend in the optimal objective function for varying levels of the **length of arcs** metric. However, the relative performance of the heuristic is markedly worse when there are longer arcs (i.e. a lower value for the LA metric).
- An increase in the **topological float** has a distinctive negative impact on the optimal objective function value, indicating that denser networks typically yield a better outcome for the contractor. The relative performance of the heuristic approximation does not show a clear trend for this topological indicator.

5.7.3 Impact of the Contract Structure

This section investigates how the structure of the contract influences the objective function value, as well as the (relative) performance of the heuristic solution procedure. Two key elements of the contract structure have been selected as independent variables: the *number of incentivised dimensions* and the *total number of segments* in the contract. Both of these are metrics for the complexity of the contracts, and by extension the pay-off function of the contractor.

The left panel of figure 5.8 shows the impact of the number of dimensions to which an incentive is attached. A first logical observation is that the optimal objective function value increases as the number of dimensions which are incentivised is increased. A second more interesting observation is that the approximation made by the heuristic approximation procedure becomes significantly worse as the number of incentivised dimensions increases. This implies that the added value from using exact optimisation techniques becomes substantially greater when incentives are attached to multiple dimensions.

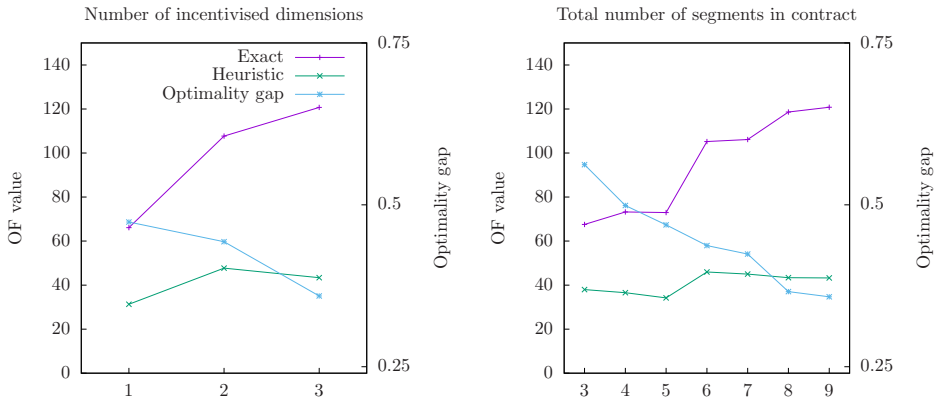


Figure 5.8: Influence of the contract structure on the absolute and relative solution quality.

The right panel of figure 5.8 uses the total number of segments in the contract as a measure for the complexity of the contract. The total number of segments is simply the sum of the number of segments for the cost, duration and scope incentive clauses. Again, the optimal function value increases as the number of segments is increased - which is logical given the nature of the dataset (see section 5.5). The absolute objective function value of the heuristic approximation method increases as well, but this increase is significantly less than the increase of the optimal function value. This is also evidenced by the relative performance measurement.

Based on these observations it can be concluded that the performance of heuristic approximation methods significantly deteriorates when the complexity of the contract (as measured by the number of incentivised dimensions and the total number of segments) increases.

Figure 5.9 illustrates the relation between computation time and contract complexity. Using computation time as a proxy for problem complexity these two graphs clearly illustrate that the problem complexity increases as the complexity of the contract (as expressed by the number of segments or the number incentivised dimensions) increases.

5.8 Conclusions

In this research a novel quantitative methodology to optimise project scheduling from the contractor's perspective is presented. Specifically, the situation where the contractor is subjected to an incentivised agreement is analysed. This methodology extends traditional multi-mode scheduling literature and incorporates multidimensional incentive contracts linked to the cost, duration and scope of the project. These incentive clauses can be of

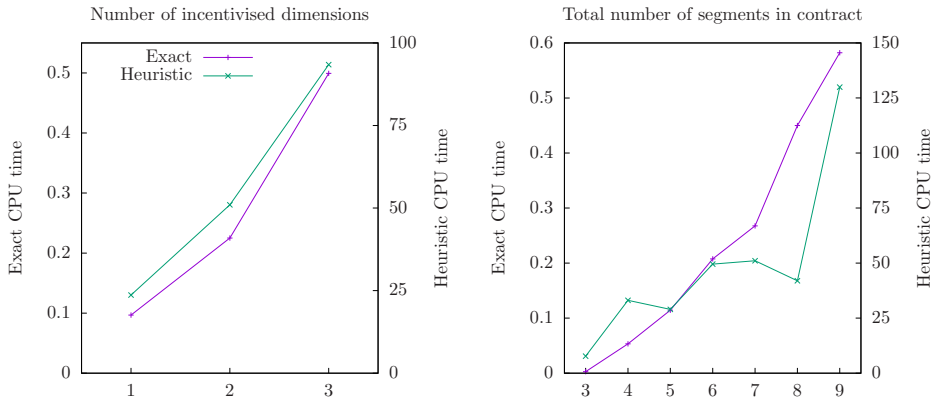


Figure 5.9: Influence of the contract structure on the CPU time.

either linear or piecewise linear nature and linked to the cost, duration and/or scope of the project.

To solve this scheduling problem a novel mixed integer linear programming programming model as well as a greedy local search algorithm are proposed and implemented. To test the performance of these solution techniques, a substantial dataset has been constructed and a computational experiment has been conducted. Based on these computational experiments the relative performance of the solution methods, as well as the influence of contract and project nature on the contractor's pay-off have been investigated.

The main conclusions from the computational experiments are the following. The relative performance in terms of solution quality of the greedy heuristic is independent from the problem size, but the computational expense of the greedy approach increases exponentially when the problem size increases. The mixed integer programming formulation proves to be quite efficient and always results in the optimal solution using less CPU time than the greedy heuristic, even for larger problem instances of up to 90 activities.

Both absolute and relative performance are significantly influenced by the project properties. Specifically a more serial network, a higher topological float or a higher order strength have a negative influence on the contractor's potential earnings. Also, the relative performance of the greedy heuristic is markedly worse when the project contains longer arcs as measured by the length of arcs metric.

A final observation from these experiments is that the relative performance of the greedy heuristic also decreases when the complexity of the contract increases. Complexity in this case being measured as the number of incentivised dimensions or the total number of piecewise linear segments in the incentive contract formulae.

This chapter also opens several avenues for future research. One possible extension

could be the introduction of resource constraints to the current problem formulation. Other extensions include the use of stochastic rather than deterministic values or the use of net present value objective taking into account the timing of the cashflows.

5.A Statistical Analysis

This appendix investigates the statistical significance of the observations made based on the computational experiments in the preceding chapter. Specifically, the observations regarding the impact of the problem size and the influence of contract complexity are tested for their statistical significance.

5.A.1 Impact of problem size

In section 5.7.1 the impact of the problem size on the relative performance of the algorithms is discussed. These experiments showed that there is a considerable difference between the average performance of the linear programming model and the simple rule of thumb heuristic. A simple (paired) t-test can be used to verify whether or not these differences are statistically significant. The results of this test are listed in table 5.7.

Problem size (nb activities)	t	df	p-value	mean difference
30	-23.805	1049	2.2e-16	-21.07647
60	-24.678	1049	2.2e-16	-43.47932
90	-36.322	1049	2.2e-16	-69.58879

Table 5.7: Results from t-test for various problem sizes.

Based on these results, it can be concluded that the differences in performance are highly statistically significant.

5.A.2 Problem structure

A similar analysis is made for the observations made in section 5.7.3, which discussed the impact of the problem structure. This contract structure was defined by two parameters: the number of incentivised dimensions and the total number of segments in a contract. In both cases a greater number indicated a more complex contract. The key question here is if the performance gap increases as the complexity of the contract increases. This was tested statistically by comparing the average of the optimality gap for different levels of contract complexity.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Number of Contract Dimensions	3	1621670	540557	146.6	<2e-16
Residuals	3146	11596929	3686		

Table 5.8: *ANOVA* analysis for the number of contract dimensions.

nb dimensions	diff	lwr	upr	Adjusted P value
1-0	-34.81688	-43.71747	-25.916297	0.0e+00
2-0	-59.97414	-68.87178	-51.076509	0.0e+00
3-0	-77.35481	-88.46752	-66.242111	0.0e+00
2-1	-25.15726	-31.58355	-18.730974	0.0e+00
3-1	-42.53793	-51.79111	-33.284747	0.0e+00
3-2	-17.38067	-26.63101	-8.130325	8.5e-06

Table 5.9: Tukey multiple comparison procedure for the number of contract dimensions.

Tables 5.8 and 5.11 show the results of an $ANOVA$ test as well as a pairwise comparison of the mean (using a Tukey test) of the optimality gap for different numbers of incentivised dimensions. The results from the $ANOVA$ analysis indicate that there are statistically significant differences. Moreover, the pairwise comparison indicates that all the results are statistically significant from each other.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
Number of Contract Segments	6	1447884	241314	64.44	<2e-16
Residuals	3143	11770715	3745		

Table 5.10: $ANOVA$ analysis for the total number of contract segments.

Tables 5.10 and 5.11 also show the results for an $ANOVA$ and a Tukey's test, this time using the total number of contract segments as the indicator for contract complexity. The results of these tests indicate that the observed trend in chapter 5.7.3 is statistically significant. Only three of the pairwise comparisons indicate that the difference is not statistically significant[13].

[13]Notably, these are all adjacent points.

nb segments	diff	lwr	upr	Adjusted P value
4-3	-33.454026	-47.47669	-19.431367	0.0000000
5-3	-35.546726	-45.65783	-25.435623	0.0000000
6-3	-56.022076	-68.60690	-43.437257	0.0000000
7-3	-57.881385	-68.54307	-47.219698	0.0000000
8-3	-72.012797	-89.75786	-54.267737	0.0000000
9-3	-74.308219	-90.24439	-58.372049	0.0000000
5-4	-2.092700	-14.71290	10.527498	0.9989942
6-4	-22.568050	-37.24499	-7.891106	0.0001207
7-4	-24.427359	-37.49283	-11.361886	0.0000008
8-4	-38.558770	-57.84405	-19.273487	0.0000001
9-4	-40.854193	-58.48929	-23.219100	0.0000000
6-5	-20.475350	-31.47590	-9.474800	0.0000009
7-5	-22.334658	-31.06999	-13.599327	0.0000000
8-5	-36.466070	-53.12505	-19.807092	0.0000000
9-5	-38.761493	-53.47869	-24.044294	0.0000000
7-6	-1.859308	-13.36797	9.649353	0.9991331
8-6	-15.990720	-34.25722	2.275775	0.1315573
9-6	-18.286143	-34.80096	-1.771324	0.0189216
8-7	-14.131412	-31.13020	2.867375	0.1773298
9-7	-16.426835	-31.52760	-1.326069	0.0227735
9-8	-2.295423	-23.01374	18.422895	0.9999026

Table 5.11: Tukey multiple comparison procedure for the total number of contract segments.

5.B Linear Model: Equations for Piecewise Linear Duration and Scope Contracts

5.B.1 Piecewise Linear Duration Contract

$$I^D = \sum_{r=1}^{NR^D} I_r^D \tag{5.52}$$

5.B.1.0.1 (0) Assisting binary variables

$$\sum_{i=1}^{|V|} f_{|V|} > B_{r+1}^D - M \cdot (1 - \alpha_r^D) \quad \forall r = 1, ..., NR^D \tag{5.53}$$

$$\sum_{i=1}^{|V|} f_{|V|} \leq B_{r+1}^D + M \cdot \alpha_r^D \quad \forall r = 1, ..., NR^D \tag{5.54}$$

$$\sum_{i=1}^{|V|} f_{|V|} > B_r^D - M \cdot \beta_r^D \quad \forall r = 1, ..., NR^D \tag{5.55}$$

$$\sum_{i=1}^{|V|} f_{|V|} \leq B_r^D + M \cdot (1 - \beta_r^D) \quad \forall r = 1, ..., NR^D \tag{5.56}$$

$$\alpha_r^D, \beta_r^D \in [0,1] \qquad r = 1, ..., NR^D \tag{5.57}$$

5.B.1.0.2 (i) Region containing the target

$$I_r^D \geq s_r^D(D^t - f_{|V|}) - M(\alpha_r^D + \beta_r^D) \tag{5.58}$$

$$I_r^D \leq s_r^D(D^t - f_{|V|}) + M(\alpha_r^D + \beta_r^D) \tag{5.59}$$

$$I_r^D \geq s_r^D(D^t - B_{r+1}^D) - M(1 - \alpha_r^D) \tag{5.60}$$

$$I_r^D \leq s_r^D(D^t - B_{r+1}^D) + M(1 - \alpha_r^D) \tag{5.61}$$

$$I_r^D \geq s_r^D(D^t - B_r^D) - M(1 - \beta_r^D) \tag{5.62}$$

$$I_r^D \leq s_r^D(D^t - B_r^D) + M(1 - \beta_r^D) \tag{5.63}$$

5.B.1.0.3 (ii) Target is lower than region lower bound

$$I_r^D \geq s_r^D(B_r^D - f_{|V|}) - M(\alpha_r^D + \beta_r^D) \tag{5.64}$$

$$I_r^D \leq s_r^D(B_r^D - f_{|V|}) + M(\alpha_r^D + \beta_r^D) \tag{5.65}$$

$$I_r^D \geq s_r^D(B_r^D - B_{r+1}^D) - M(1 - \alpha_r^D) \tag{5.66}$$

$$I_r^D \leq s_r^D(B_r^D - B_{r+1}^D) + M(1 - \alpha_r^D) \tag{5.67}$$

$$I_r^D \geq 0 - M(1 - \beta_r^D) \tag{5.68}$$

$$I_r^D \leq 0 + M(1 - \beta_r^D) \tag{5.69}$$

5.B.1.0.4 (iii) Target exceeds region upper bound

$$I_r^D \geq s_r^D(B_{r+1}^D - f_{|V|}) - M(\alpha_r^D + \beta_r^D) \tag{5.70}$$

$$I_r^D \leq s_r^D(B_{r+1}^D - f_{|V|}) + M(\alpha_r^D + \beta_r^D) \tag{5.71}$$

$$I_r^D \geq 0 - M(1 - \alpha_r^D) \tag{5.72}$$

$$I_r^D \leq 0 + M(1 - \alpha_r^D) \tag{5.73}$$

$$I_r^D \geq s_r^D(B_{r+1}^D - B_r^D) - M(1-\beta_r^D) \tag{5.74}$$

$$I_r^D \leq s_r^D(B_{r+1}^D - B_r^D) + M(1-\beta_r^D) \tag{5.75}$$

5.B.2 Piecewise Linear Scope Contract

$$I^S = \sum_{r=1}^{NR^S} I_r^S \tag{5.76}$$

5.B.2.0.1 (0) Assisting binary variables

$$\sum_{i=1}^{|V|}\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} > B_{r+1}^S - M\cdot(1-\alpha_r^S) \quad \forall r=1,...,NR^S \tag{5.77}$$

$$\sum_{i=1}^{|V|}\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} \leq B_{r+1}^S + M\cdot\alpha_r^S \quad \forall r=1,...,NR^S \tag{5.78}$$

$$\sum_{i=1}^{|V|}\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} > B_r^S - M\cdot\beta_r^S \quad \forall r=1,...,NR^S \tag{5.79}$$

$$\sum_{i=1}^{|V|}\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} \leq B_r^S + M\cdot(1-\beta_r^S) \quad \forall r=1,...,NR^S \tag{5.80}$$

$$\alpha_r^S, \beta_r^S \in [0,1] \quad r=1,...,NR^S \tag{5.81}$$

5.B.2.0.2 (i) Region containing the target

$$I_r^S \geq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} - S^t) - M(\alpha_r^S + \beta_r^S) \tag{5.82}$$

$$I_r^S \leq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im}\cdot x_{im} - S^t) + M(\alpha_r^S + \beta_r^S) \tag{5.83}$$

$$I_r^S \geq s_r^S(B_{r+1}^S - S^t) - M(1-\alpha_r^S) \tag{5.84}$$

$$I_r^S \leq s_r^S(B_{r+1}^S - S^t) + M(1-\alpha_r^S) \tag{5.85}$$

$$I_r^S \geq s_r^S(B_r^S - S^t) - M(1-\beta_r^S) \tag{5.86}$$

$$I_r^S \leq s_r^S(B_r^S - S^t) + M(1-\beta_r^S) \tag{5.87}$$

5.B.2.0.3 (ii) Target is lower than region lower bound

$$I_r^S \geq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im} \cdot x_{im} - B_r^S) - M(\alpha_r^S + \beta_r^S) \tag{5.88}$$

$$I_r^S \leq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im} \cdot x_{im} - B_r^S) + M(\alpha_r^S + \beta_r^S) \tag{5.89}$$

$$I_r^S \geq s_r^S(B_{r+1}^S - B_r^S) - M(1 - \alpha_r^S) \tag{5.90}$$

$$I_r^S \leq s_r^S(B_{r+1}^S - B_r^S) + M(1 - \alpha_r^S) \tag{5.91}$$

$$I_r^S \geq 0 - M(1 - \beta_r^S) \tag{5.92}$$

$$I_r^S \leq 0 + M(1 - \beta_r^S) \tag{5.93}$$

5.B.2.0.4 (iii) Target exceeds region upper bound

$$I_r^S \geq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im} \cdot x_{im} - B_{r+1}^S) - M(\alpha_r^S + \beta_r^S) \tag{5.94}$$

$$I_r^S \leq s_r^S(\sum_{i=1}^{|V|}\sum_{m=1}^{M_i} S_{im} \cdot x_{im} - B_{r+1}^S) + M(\alpha_r^S + \beta_r^S) \tag{5.95}$$

$$I_r^S \geq 0 - M(1 - \alpha_r^S) \tag{5.96}$$

$$I_r^S \leq 0 + M(1 - \alpha_r^S) \tag{5.97}$$

$$I_r^S \geq s_r^S(B_r^S - B_{r+1}^S) - M(1 - \beta_r^S) \tag{5.98}$$

$$I_r^S \leq s_r^S(B_r^S - B_{r+1}^S) + M(1 - \beta_r^S) \tag{5.99}$$

5.C Overview of Notation

Index	**Description**
$i \in V = \{1, ..., \lvert V \rvert\}$	Activity index
$m \in \{1, ..., M_i\}$	Activity mode index
$r \in R^C \vee R^D \vee R^S$	Contract region index
Variable	**Description**
x_{im}	Binary variable = 1 when activity i is executed in mode m
f_i	Finishing time of activity i
I^{tot}	Total incentive amount earned
I^C I^D, I^S	Total cost, duration and scope incentive
I_r^C I_r^D, I_r^S	Cost, duration and scope incentive earned in region r
α_r^C, α_r^D, α_r^S	Binary variable = 1 if project cost/duration/scope exceeds the upper bound of region r
β_r^C, β_r^D, β_r^S	Binary variable = 1 if project cost/duration/scope is lower than the lower bound of region r
Parameter	**Description**
C_{im}, D_{im}, S_{im}, E_{im}	Cost, duration, scope and effort associated with mode m of activity i
E_{im}^{cost}	Cost of effort associated with mode m of activity i
s_r^C	Cost sharing ratio in region r
v_r^D, v_r^S	Incentive amount per unit of time/scope in region r of the contract
C^t, D^t, S^t	Cost, duration and scope target in the contract
B_r^C, B_r^D, B_r^S	Lower bound of region r of the cost/duration/scope contract
M	A large value (used for the big-M method)

Table 5.12: Overview of notation

6

Improving Contractor's Control During Project Execution: Earned Incentive Management

This research introduces a novel project control method for projects that use cost and/or time incentives. The proposed earned incentive management (EIM) technique improves upon the traditional earned value management (EVM) methodology for project control. This is done by measuring the deviation in the accrual of incentives, rather than the time and cost performance relative to the planned schedule. The proposed dedicated approach avoids two key issues when controlling incentivized projects using traditional earned value management techniques. Firstly, the impact of variations in the cost and time dimensions are adequately weighted in the control signals. Secondly, the technique is capable of monitoring the potential non-linear accrual of incentive amounts throughout the project. The performance of the proposed technique is tested by means of a computational experiment on 4,200 projects of varying size, structure and type of incentive contract. The results show that the proposed technique significantly outperforms traditional EVM techniques.

6.1 Introduction

Project managers monitoring complex and risky projects rely on support from models to guide their decision making. These models must provide the correct information to the responsible actors, without inundating them with extraneous data (Browning, 2010). This research presents a novel method to monitor the progress of incentivised projects. This approach is based on the well known earned value and earned schedule techniques, and extends the methodology in order to give more accurate signals to represent the status of the project. The technique has been designed to need no more information than the widely adopted earned value metrics, significantly lowering the threshold for adoption by practitioners.

This new methodology 'earned incentive management' (EIM) circumvents two key issues related to the monitoring of incentivised projects. Firstly, the relative importance of the time and cost dimensions, as represented by the magnitude of their respective incentives is incorporated in the monitoring signal. Because of this, it becomes easier to set managerially relevant thresholds based on the expected impact on the bottom line. A second issue resolved by the EIM technique is that the accrual of incentives, as opposed to the accrual of value and the progression of time and cost as used in traditional EVM/ES, is not necessarily a strictly increasing function over time. This makes it harder to judge whether or not a project is progressing as planned, earning the desired incentive pay-off.

Because of the increased adoption of incentive agreements in practice, the literature on the subject has been steadily expanding in recent years (see section 6.3). Nevertheless, a practical methodology for monitoring the progress in incentivised projects has yet to be proposed. Filling this gap is the main objective of this research. To achieve this objective, the quantitative incentive framework proposed in chapter 3 is used as a foundation to model incentivised projects.

The scope of this research is limited to the controlling phase of the project. As such, it is assumed that the contract between the project owner and the contractor has already been negotiated, and that the contractor has already optimized the manner in which (s)he will execute the contract. As such, the objective is to optimize the control signals to assist the contractor during the execution of the contract.

A key difference between this chapter and the preceding chapters on incentives in projects is that the scope dimension of the trade-off is not included. The reason for this is that the traditional EVM and ES methodology does not include this dimension. Hence, including this dimension would require extending multiple facets of the original technique simultaneously which would complicate the model presented in this chapter. Therefore, the choice has been made not to include the scope dimension and to focus instead on a

solid comparison of this technique and the well-known and widely used version of earned value management.

The remainder of this article is structured as follows. First the literature which has formed the basis for the *EIM* methodology is reviewed in section 6.2. Next, section 6.4 introduces the *EIM* methodology, using both a theoretical overview (section 6.4.1) and a practical example of the methodology (section 6.4.2). The validity of this methodology is then investigated using computational experiments in section 6.5, prior to formulating a general conclusion in section 6.6. An overview of the notation used can be found in appendix 6.A.

6.2 Literature on Project Control

The relevant literature consists of two key research areas. Firstly, the existing research on project control, specifically earned value management and earned schedule which is discussed in this section. Secondly, the literature on incentive contracting for projects, which was analysed in detail in chapter 2.

Project control is one of the three key components of dynamic scheduling, a concept introduced by Uyttewaal (2005) and refined by Vanhoucke (2013). During the control phase, the responsible actor periodically monitors the project's progress, taking action when (s)he believes the project is liable to go out of control. To do this, actors frequently employ quantitative control techniques that provide warning signals linked to the project performance.

Having a correct view on how a project is performing during its execution has a profound impact on the quality of the decisions made by the project manager (Bendoly and Swink, 2007). One of the most frequently used techniques to create such performance data is *earned value management*, a technique which has been in use since the 1960s (Vanhoucke, 2012a). The key advantage of this technique being the low effort required to track the performance of a project. In spite of its considerable age, earned value management and its more recent extension earned schedule have been the topic of numerous publications in recent years (Acebes et al., 2015; Batselier and Vanhoucke, 2015; Chen et al., 2016; Elshaer, 2013; Khamooshi and Golafshani, 2014; Naeni et al., 2014).

An introduction on the basic and more advanced concepts of earned value management is given by several authors such as Anbari (2003), Fleming and Koppelman (2005), Vanhoucke (2010b) and Vanhoucke (2014). For a recent and extensive review of literature on earned value management and project control the reader is referred to Willems and Vanhoucke (2015). A brief summary of the basic principles of earned value management, as well as the basic formulae is presented in appendix 6.B.

Despite its popularity, the earned value management technique is of course imperfect. Hence, the technique is continually being extended and modified to fit specific situations or in order to cover more aspects of project management. One example of this is the earned schedule technique (Lipke, 2003). More recently, an extension has been proposed by Browning (2014), who created a novel framework to account for quality, uncertainty, risk, as well as opportunities when controlling project execution. This chapter also presents an extension of this model, specifically to account for the use of incentive contracts.

6.3 Incentive Contracts

The project control methodology proposed in this research focuses on the two most frequently used incentive dimensions: project cost and duration. The EIM methodology can be used regardless of the type of contract, but for this research only the most frequently used contract archetypes are used: linear, piecewise linear and quadratic (non-linear) formulations. The notation used for these equation is identical to the notation used in chapter 3.

A linear incentive clause is the simplest possible incentive. Such an incentive only requires the project owner to specify a specific target, and the magnitude of the (dis-)incentive awarded to the contractor for deviations from this target.

$$I^C(C) = s^C \cdot (C^t - C) \tag{6.1}$$

Equation 6.1 is an example of such a linear contract for the cost dimension, and defines $I^C(\cdot)$ as a function calculating the awarded incentive amount, based on the final cost of the project C. This equation is defined by two parameters: the target cost C^t and the fraction of the deviation ($s^C \in [0, 1]$) from this target which will be awarded to the contractor as an incentive.

A similar equation can be constructed for the time performance of the contractor, the sole difference being that the magnitude of the incentive is no longer a fraction of the deviation, but a monetary valuation per unit of time (v^D).

A piecewise linear contract is an extension of these simple linear formulations, allowing for different incentive magnitudes to be used at varying distances of the target cost or duration. As shown in equation 6.2 and illustrated in figure 6.1 for the cost dimension, this is done by defining a number of regions r wherein specific sharing ratios s^C_r are used. These regions are simply defined by their lower bounds B^C_r. A similar approach can be taken for the duration dimension, again with the sole difference being that the magnitude is no longer defined by a sharing ratio in the range $[0, 1]$, but by a monetary valuation per time unit v^D_r (see equation 6.3).

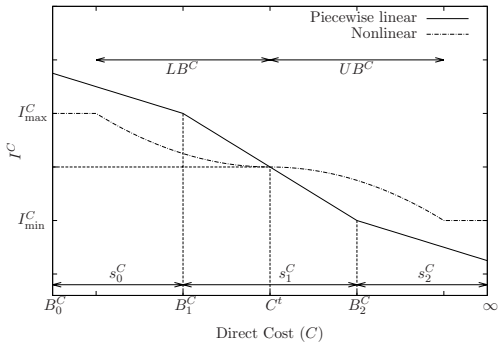


Figure 6.1: Illustration of piecewise linear and non-linear cost incentive clauses.

$$I^C(C) = \begin{cases} s_1^C \cdot (C^t - B_2^C) + s_2^C \cdot (B_2^C - C) & \text{if } C > B_2^C \\ s_1^C \cdot (C^t - C) & \text{if } B_1^C \leq C \leq B_2^C \\ s_1^C \cdot (C^t - B_1^C) + s_0^C \cdot (B_1^C - C) & \text{if } C < B_1^C \end{cases} \tag{6.2}$$

$$I^D(D) = \begin{cases} v_1^D \cdot (D^t - B_2^D) + v_2^D \cdot (B_2^D - D) & \text{if } D > B_2^D \\ v_1^D \cdot (D^t - D) & \text{if } B_1^D \leq D \leq B_2^D \\ v_1^D \cdot (D^t - B_1^D) + v_0^D \cdot (B_1^D - D) & \text{if } D < B_1^D \end{cases} \tag{6.3}$$

A third type of equation which is used to determine the incentive amount awarded is a non-linear incentive clause. The main advantage of such an incentive is the progressiveness of the reward and/or penalty, which often allows a closer accord with the underlying cost structure of the contractor, and therefore a better alignment of the objectives of both actors.

Equations 6.4 and 6.5 show the non-linear formulations which are used in this research (these equations are based on those used by Shr and Chen (2003)).The profile of the non-linear cost incentive equation is also indicated on figure 6.1. Similarly to the (piecewise) linear contracts, the non-linear contracts require the owner to specify a target for the cost (C^t) or duration (D^t) dimensions. The magnitude of the incentive is determined by the greatest incentive and disincentive amounts ($I^C_{\max}$, $I^C_{\min}$, $I^D_{\max}$ and $I^D_{\min}$). Besides the overall magnitude of the incentive, the owner also has to specify the range over which these amounts should be spread. This range is expressed as a distance from the target cost or duration. The positive incentives are spread over the ranges LB^C and LB^D for the cost and duration dimensions respectively. Similarly, the disincentive amounts are spread over the ranges UB^C and UB^D for the cost and duration dimensions respectively.

$$I^C(C) = \begin{cases} \min\left(I^C_{\max}; I^C_{\max} \cdot \left(\frac{C^t - C}{LB^C}\right)^2\right) & \text{if } C \leq C^t \\ \max\left(I^C_{\min}; I^C_{\min} \cdot \left(\frac{C - C^t}{UB^C}\right)^2\right) & \text{if } C > C^t \end{cases} \tag{6.4}$$

$$I^D(D) = \begin{cases} \min\left(I^D_{\max}; I^D_{\max} \cdot \left(\frac{D^t - D}{LB^D}\right)^2\right) & \text{if } D \leq D^t \\ \max\left(I^D_{\min}; I^D_{\min} \cdot \left(\frac{D - D^t}{UB^D}\right)^2\right) & \text{if } D > D^t \end{cases} \tag{6.5}$$

Naturally, a lot of other contract clauses are used. However, the aforementioned types are some of the most frequently employed in practice and studied in academic literature. Moreover, the goal of this research is mainly to investigate the performance of project control metrics rather than offer an in-depth analysis of the impact of highly-specific incentive contracts.

6.4 Earned Incentive Management

6.4.1 EIM Methodology

Figure 6.2 shows a schematic overview of the proposed EIM methodology. The methodology is based on a comparison of three schedules: the break-even schedule (β), the planned execution schedule (π), and the actual execution (α).

The break-even schedule is a project plan which results in a final cost C^β and duration D^β which are exactly equal to the target cost (C^t) and target duration (D^t) as specified in the contract. Hence, the incentives awarded are exactly zero for this scenario. Defining the break-even schedule requires specifying a duration (D_i^β) and cost (C_i^β) for each activity i. The planned accrual of value (PV_t^β) of the break-even schedule serves as a benchmark against which the accrual of incentives can be compared.

The second schedule is the planned execution schedule (π), which again requires a cost (C_i^π) and a duration (D_i^π) to be specified for each activity i. This schedule is the result of an optimisation which takes into account the properties of the activities constituting the project, as well as the potential incentive earned. The objective of such an optimisation is to maximise the profits for the actor executing the project, weighing his own investments against the marginal incentive earnings, whilst also making the correct trade-off between activity duration and cost. An example of such an optimisation can be found in chapter 5, where a multi-mode optimisation problem for projects with incentive contracts is presented. Hence, a comparison of the break-even (β) and the planned execution (π) provides insight in the desired accrual of the cost (I^C) and duration (I^D) incentives over time, as

shown in figure 6.2.

The third and final component is the actual execution schedule (α), which consists of data gathered during the execution of the project. This includes the percentage completed (PC_i^{α}), as well as the incurred cost (C_i^{α}) for each activity i. This incurred cost is equivalent to the actual cost (AC) definition used in the traditional EVM approach. By comparing the the actual (α) to the break-even (β) schedule, the actual accrual of the cost (I^C) and duration (I^D) incentives can be monitored.

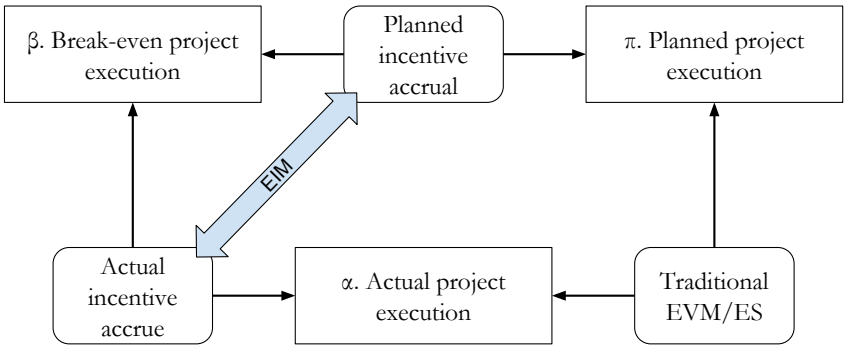


Figure 6.2: Schematic overview of the EIM methodology

The project progress is monitored by comparing the actual accrual of incentives, as derived from the comparison of schedules α and β, and the planned accrual of incentives, as was derived from the comparison of π and β. This approach can be contrasted with the traditional EVM/ES methodology, which compares the planned project execution (π) to the actual project execution (α).

By comparing the accrual of incentives rather than the time and cost performance, the signals are more closely linked to the contractor's bottom line than in the EVM/ES approach. Moreover, this also enables tracking possible irregular accrual of incentives. The remainder of this section explains how the metrics for this comparison can be calculated and interpreted, prior to illustrating this process on an example in section 6.4.2.

6.4.1.1 Step 1: Analyze the Planned Accrue of Value for the Break-even Schedule

The first step of the process simply aggregates the activity-specific information of the break-even schedule in order to accurately represent the accrual of value over time. This is done using discrete time intervals indexed $t = 1, ..., T$, which represent the moments when the project's progress is measured (i.e. the control periods). As such, the planned value of the work for each time period can be calculated as shown in equation 6.6.

$$PV_t^{\beta} = \sum_{i=1}^{n} PC_{it}^{\beta} \cdot C_i^{\beta} \tag{6.6}$$

In equation 6.6, PC_{it}^{β} represents the planned percentage completed of activity i in time period t.

6.4.1.2 Step 2: Calculate the Planned Incentive Accrue

Once the planned value for the break-even (i.e. the benchmark project execution) has been calculated, the next step is to compare the planned execution to the break-even schedule in order to obtain the planned accrual of incentives. The accrual of cost in the planned schedule can simply be calculated as follows:

$$C_t^{\pi} = \sum_{i=1}^{n} PC_{it}^{\pi} \cdot C_i^{\pi} \tag{6.7}$$

Next, the earned value of the planned execution for each time period (EV_t^{π}) is calculated as shown in Equation 6.8.

$$EV_t^{\pi} = \sum_{i=1}^{n} PC_{it}^{\pi} \cdot C_i^{\beta} \tag{6.8}$$

In equation 6.8, PC_{it}^{π} represents the fraction of the work content of activity i which has already been completed in time period t according to the planned execution schedule (π). This earned value information can then be used to determine the planned time performance of the project. This is done using the earned schedule technique, which has the advantage of being expressed in units of time rather than cost, allowing for a more intuitive calculation of the incentive amounts. The earned schedule for the planned project execution (ES_t^{π}) at time period t can be calculated as shown in equations 6.9 and 6.10.

$$ES_t^{\pi} = \tau + \frac{EV_t^{\pi} - PV_{\tau}^{\beta}}{PV_{\tau+1}^{\beta} - PV_{\tau}^{\beta}} \tag{6.9}$$

$$\tau = t | PV_t^{\beta} \leq EV_t^{\pi} < PV_{t+1}^{\beta} \tag{6.10}$$

Once these quantities have been calculated, the planned accrual of cost and duration incentives can be outlined. The planned accrual of the cost incentive at moment t is calculated by inserting the difference of the target cost (C^t) and the cost variation (CV_t^{π}) in the cost incentive equation $I^C(C)$ (see equations 6.2 and 6.4 in section 6.3). This results in a planned cost incentive amount $I_t^{C\pi}$ for time period t.

$$I_t^{C\pi} = I^C(C^t - CV_t^\pi) = I^C(C^t - [EV_t^\pi - C_t^\pi]) \tag{6.11}$$

In a similar vein, the planned accrual of the duration incentive can be calculated by inserting the difference between the target duration (D^t) and the schedule variance ($SV(t)_t^\pi$) into the duration incentive equation $I^D(D)$ (see equations 6.3 and 6.5 in section 6.3). The result of this calculation is a planned duration incentive amount $I_t^{D\pi}$ for time period t.

$$I_t^{D\pi} = I^D(D^t - SV(t)_t^\pi) = I^D(D^t - [ES_t^\pi - t]) \tag{6.12}$$

6.4.1.3 Step 3: Monitor Incentive Accrue During Project Progress

Once the benchmark incentive accrual has been established by calculating $I_t^{C\pi}$ and $I_t^{D\pi}$, the project progress can be monitored. In order to do this the actual project progress is compared to the break-even schedule (see figure 6.2). The calculations required to determine the accrual of the incentives are analogous to those used to determine the planned accrual of incentives, as is shown below.

First and foremost, the actual costs (C_t^α) at a specific moment in time can simply be observed, either on the level of the project or at the level of the complete project[1]. This approach is identical to the measurement of actual costs during traditional earned value management, and is a practice requiring little effort since these costs are tracked using standard cost accounting.

Aside from the incurred costs, the accrual of value has to be measured at the control periods (t) during the execution of the project. This accrual is calculated using the break-even scenario as a reference point, as shown in equation 6.13. In this equation PC_{it}^α represents the actual percentage completed for activity i in time slot t. The earned value for the actual schedule (EV_t^α) is calculated by multiplying this percentage completed with the foreseen cost from the break-even scenario.

$$EV_t^\alpha = \sum_{i=1}^{n} PC_{it}^\alpha \cdot C_i^\beta \tag{6.13}$$

Once the earned value at a certain point in time is known, the earned schedule can be calculated to determine the time performance (equations 6.14 and 6.15).

[1] For the example in section 6.4.2 it was assumed that the accrual of costs happened linearly throughout the activity's progress. As such, actual cost can be calculated as $C_t^\alpha = \sum_{i=1}^{n} PC_{it}^\alpha \cdot C_i^\alpha$, which is analogous to equation 6.7.

$$ES_t^{\alpha} = \tau + \frac{EV_t^{\alpha} - PV_{\tau}^{\beta}}{PV_{\tau+1}^{\beta} - PV_{\tau}^{\beta}} \tag{6.14}$$

$$\tau = t | PV_t^{\beta} \leq EV_t^{\alpha} < PV_{t+1}^{\beta} \tag{6.15}$$

Finally, these values are used to determine the actual accrual of incentives at a certain point in time. Again these equations are similar to those used to determine the planned incentive accrue. Equation 6.16 shows the formula for the actual accrual of the cost incentive ($I_t^{C\alpha}$), and equation 6.17 does the same for the duration incentive ($I_t^{D\alpha}$). Similarly to the equations used to determine the planned accrual of incentives, these calculations link back to the incentive clauses contained within the contract, as represented by equations 6.2 and 6.3.

$$I_t^{C\alpha} = I^C(C^t - CV_t^{\alpha}) = I^C(C^t - [EV_t^{\alpha} - C_t^{\alpha}]) \tag{6.16}$$

$$I_t^{D\alpha} = I^D(D^t - SV(t)_t^{\alpha}) = I^D(D^t - [ES_t^{\alpha} - t]) \tag{6.17}$$

As indicated in figure 6.2, the actual monitoring of the project progress is done by comparing the planned incentive accrual ($I_t^{C\pi}$, $I_t^{D\pi}$) to the actual accrual of incentives ($I_t^{C\alpha}$, $I_t^{D\alpha}$). This can be done either separately for the cost and duration dimensions, or jointly for the incentivised dimensions combined.

The simplest measure of project performance is the difference between these two scenarios at a specific point in time. In line with the EVM/ES nomenclature, these differences are dubbed *incentive variation*. Naturally, this metric can be calculated for the cost ($IV(C)_t$), duration ($IV(D)_t$) or total ($IV(tot)_t$) incentive amounts. Equations 6.18, 6.19 and 6.20 show how these respective measures are calculated.

$$IV(C)_t = I_t^{C\alpha} - I_t^{C\pi} \tag{6.18}$$

$$IV(D)_t = I_t^{D\alpha} - I_t^{D\pi} \tag{6.19}$$

$$IV(tot)_t = (I_t^{C\alpha} + I_t^{D\alpha}) - (I_t^{C\pi} + I_t^{D\pi}) \tag{6.20}$$

A negative value for these metrics can be interpreted as a worse than planned performance, whereas a positive value indicates a better than planned performance. In order to transform this information into a control signal a threshold has to be set by the actor who controls the project. The specific value of this threshold should be motivated from an eco-

nomic perspective, including factors such as the inherent project risk as well as the weight of the project within the total portfolio. A key factor which also has to be considered is the effort which the actor is able to invest in project control. Setting the threshold too tight may cause an inordinate workload to rectify a project that was not truly in trouble. Combined with a threshold, these incentive variations result in a binary control indicator which signifies when the project performance lies outside of the acceptable range.

6.4.2 Example

This section illustrates the EIM methodology using a small example project containing three activities. The structure of this project is summarized in an activity on the node network in figure 6.3. Furthermore, it is assumed that the contractor is subjected to a contract containing a linear incentive for both the duration and the cost of the project. These incentive clauses have the following form:

$$I^C(C) = (C - C^t) \cdot s^C = (C - 22) \cdot 0.5 \tag{6.21}$$

$$I^D(D) = (D - D^t) \cdot v^D = (D - 9) \cdot 0.75 \tag{6.22}$$

Where $I^C(C)$ and $I^D(D)$ represent the awarded cost and duration incentive based on the final project cost (C) and duration (D). The target cost (C^t) and duration (D^t) in this contract are set to 22 and 9 respectively. Cost savings or excesses are shared equally by setting the sharing ratio (s^C) of the cost contract to 0.5. Time savings or overruns are valued (v^D) at 0.75 per unit of time.

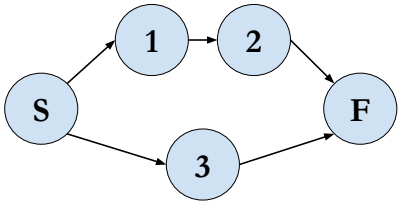


Figure 6.3: Activity on the node representation of the example problem

The first step in the EIM procedure is the creation of a break-even (β) and planned (π) schedule by defining a duration and cost for each of the activities, as shown in table 6.1. From this table it is clear that the total project cost for the break-even schedule (C^β) is equal to the target cost (C^t):

Activity (i)	Break Even (β)		Planned (π)	
	D_i^β	C_i^β	D_i^π	C_i^π
1	5	10	4	9
2	4	5	4	4
3	6	7	8	5

Table 6.1: Activity information for the break-even and planned schedules

$$C^\beta = \sum_{i=1}^{n} C_i^\beta = 10 + 5 + 7 = C^t \tag{6.23}$$

Similarly, the duration of the break-even contract (D^β) is equal to the target duration (D^t) specified in the contract:

$$D^\beta = \max(D_1^\beta + D_2^\beta, D_3^\beta) = 9 = D^t \tag{6.24}$$

Analogous calculations for the planned execution schedule (π) reveal that the contractor is aiming for a duration (D^π) of 8 time periods, and a total cost (C^π) equal to 18. By doing so, the contractor hopes to earn an incentive for both the time and cost dimension, equal to 0.75 and 2 respectively[2].

An important note is that the planned execution does not necessarily result in a positive outcome in terms of awarded incentives, since it may be more beneficial to allocate scarce resources to other projects where the return on investment for these resources is greater. For example, assume that a contractor is simultaneously executing two projects for separate project owners and he has to allocate scarce resources such as equipment, staff or management time between these two projects. If one of these two projects is has substantially higher (dis-)incentive payments, the contractor is likely to allocate a disproportionate amount of his resources to this project, all other factors remaining equal.

The next step is to calculate the planned accrual of both the cost and time incentive based on the planned schedule (π). An earliest start schedule is used for this example, but any type of schedule can of course be used. A second assumption made for this example is that the accrual of value for the activities happens in a linear fashion. Again, this assumption is made to keep the example as simple as possible, and other assumptions can also be used to adapt the technique to the reality of the project without changing the proposed EIM technique. Based on these assumptions, it is straightforward to calculate the planned value of each time period for the break even schedule, as described in section 6.4.1.1. The results of these calculations are shown in the PV_t^β column of table 6.2.

[2] $I^D = (9 - 8) \cdot 0.75$ and $I^C = (22 - 18) \cdot 0.5$

t	PV_t^β	C_t^π	EV_t^π	ES_t^π	$I_t^{C\pi}$	$I_t^{D\pi}$	$I_t^{tot\pi}$	C_t^α	EV_t^α	ES_t^α	$I_t^{C\alpha}$	$I_t^{D\alpha}$	$I_t^{tot\alpha}$
1	3.17	2.88	3.38	1.07	0.25	0.05	0.30	2.51	2.77	0.87	0.13	-0.09	0.03
2	6.33	5.75	6.75	2.13	0.50	0.10	0.60	5.48	6.05	1.91	0.28	-0.07	0.22
3	9.50	8.63	10.13	3.20	0.75	0.15	0.90	8.35	9.23	2.91	0.44	-0.06	0.38
4	12.67	11.50	13.50	4.26	1.00	0.20	1.20	11.12	12.32	3.89	0.60	-0.08	0.52
5	15.83	13.13	15.63	4.93	1.25	-0.05	1.20	13.61	15.34	4.84	0.87	-0.12	0.75
6	18.25	14.75	17.75	5.79	1.50	-0.16	1.34	15.13	17.31	5.61	1.09	-0.29	0.80
7	19.50	16.38	19.88	7.30	1.75	0.23	1.98	16.85	19.55	7.04	1.35	0.03	1.38
8	20.75	18.00	22.00	9.00	2.00	0.75	2.75	18.09	21.16	8.33	1.53	0.24	1.78
9	22.00	-	-	-	-	-	-	18.75	22.00	9.00	1.63	0.00	1.63

Table 6.2: Calculation of the EIM metrics for the example project.

t	$IV(C)_t$	$IV(D)_t$	$IV(tot)_t$
1	-0.12	-0.14	-0.27
2	-0.22	-0.17	-0.38
3	-0.31	-0.21	-0.52
4	-0.40	-0.28	-0.68
5	-0.38	-0.07	-0.45
6	-0.41	-0.14	-0.54
7	-0.40	-0.19	-0.59
8	-0.47	-0.51	-0.97
9	-0.38	-0.75	-1.13

Table 6.3: Incentive variance for the example problem

Using the same two assumptions, the planned accrual of value and incentives can be calculated as described in section 6.4.1.2. The results of these calculations are shown in table 6.2. When looking at these results it can be observed that the accrual of the cost incentive is expected to be quite linear, whereas the accrued duration incentive is more volatile over time. This aspect is highly relevant when monitoring the accrual of the incentive amount over time.

Once these calculations have been made, the progress of the project can be monitored using the formulae described in section 6.4.1.3. For this example it is assumed that the status of the project has been checked at each discrete time period t. The results of the calculations are shown in the rightmost part of table 6.2. Note that the earned value of the actual project execution (EV_t^α) is a given is this table, as this is observed by the contractor. Using this information, the project's progress can be monitored. This is done by calculating the incentive variances (see section 6.4.1.3), as shown in table 6.3.

As a point of reference, the traditional EVM approach has also been applied to the example project. The results of these calculations are shown in table 6.4. As indicated in figure 6.2, the traditional EVM technique compares the actual project execution to the planned project execution. Hence, the planned value (PV) measure in table 6.4 cor-

t	PV	EV	AC	ES	CV	CPI	SV	SPI	$SV(t)$	$SPI(t)$
1	2.88	2.37	2.51	0.83	-0.14	0.94	-0.36	0.83	-0.17	0.83
2	5.75	5.17	5.48	1.80	-0.31	0.94	-0.27	0.90	-0.20	0.90
3	8.63	7.87	8.35	2.74	-0.48	0.94	-0.28	0.91	-0.26	0.91
4	11.50	10.48	11.12	3.64	-0.64	0.94	-0.38	0.91	-0.36	0.91
5	13.13	12.94	13.61	4.88	-0.67	0.95	0.48	0.99	-0.12	0.98
6	14.75	14.46	15.13	5.82	-0.67	0.96	0.38	0.98	-0.18	0.97
7	16.38	16.17	16.85	6.87	-0.69	0.96	0.48	0.99	-0.13	0.98
8	18.00	17.40	18.09	7.63	-0.69	0.96	0.09	0.97	-0.37	0.95
9	18.00	18.00	18.75	8.00	-0.75	0.96	0.75	1.00	-1.00	0.89

Table 6.4: The traditional EVM metrics applied to the example project.

responds to the costs of the planned project execution (C_t^{π}) in the EIM methodology. Similarly, the actual costs (AC) of the EVM methodology are equivalent to the costs of the actual schedule (C_t^{α}) in the EIM methodology. A key difference between the two techniques is that the earned value (EV) of the EVM technique is expressed in terms of the planned schedule, rather than the break-even schedule and does not correspond to any of the earned value calculations in the proposed EIM approach.

The results of the EVM and EIM techniques are visualised in figure 6.4. The top two panels in this figure show the traditional EVM approach, and the bottom two panels show the EIM approach. Both control techniques clearly indicate that the project's performance was worse than the performance outlined in the planned schedule.

When comparing the signals provided by the two techniques, the advantages of the EIM technique are clear. Firstly, the deviations in performance are measured in terms of their actual financial impact on the project performance. A second advantage which follows from this first observation is the possibility of creating a single aggregated metric which represents the overall performance of the project ($IV(tot)_t$ in the bottom right panel). A third key advantage of the EIM approach is the possibility of tracking non-linear or non-increasing accrual of incentives throughout the project. An example of this is the cost performance during the first four periods of the project. Whereas the EVM technique indicates that the cost performance remains relatively stable, it is clear that the deviation from the planned cost incentive is becoming progressively worse during these time periods (the $IV(C)_t$ curve on the bottom right panel is steadily declining).

As was stated in 6.4.1.3, it would be up to the actor controlling the project to define thresholds for the total incentive variation (the $IV(tot)_t$) and/or for the cost and duration incentive variations ($IV(C)_t$ and $IV(D)_t$).

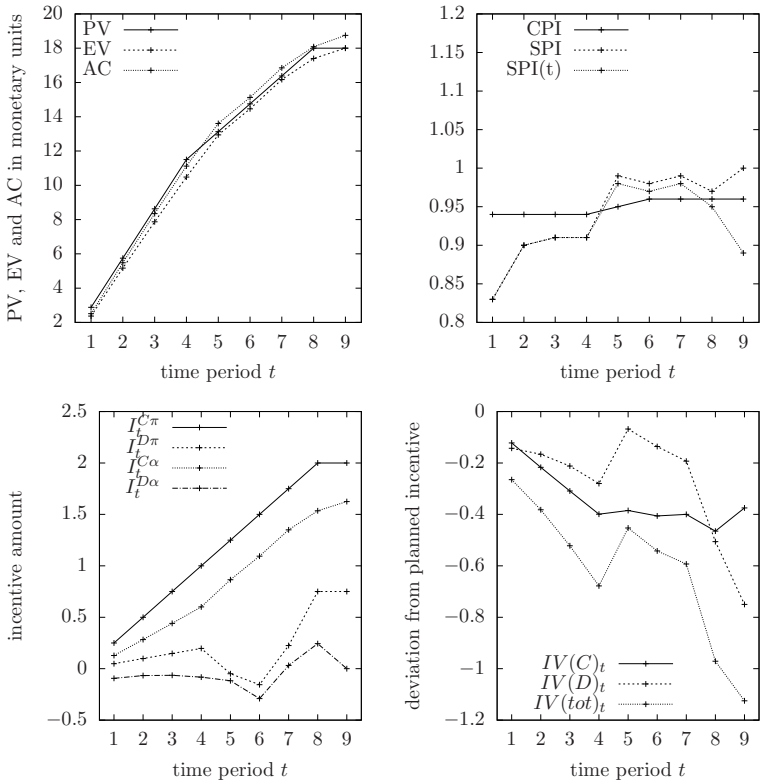


Figure 6.4: Analysis of the example using EVM (top panels) and EIM (bottom panels).

6.5 Computational Experiments

In this section a computational experiment is presented which tests the performance of the proposed EIM methodology when compared to the performance of the traditional EVM approach for project control. The experimental procedure is first outlined in section 6.5.1. Next, the dataset and parameters used for the experiment are discussed in section 6.5.2. Finally, the results which compare the performance of the different project control approaches are discussed in section 6.5.3.

6.5.1 Experiment Design

This section explains the nature of the computational experiment. First, section 6.5.1.1 gives an overview of the simulation procedure used to model activity progress. Next, the way in which the results from these simulation runs are aggregated and analysed is discussed in section 6.5.1.2.

6.5.1.1 Simulation Procedure

To compare the performance of the project control techniques, the actual project progress is simulated. For each of these simulations, the binary signals produced by the control techniques are determined for a number of predefined control periods. These signals are then compared to the actual project outcome, enabling the veracity of signals to be evaluated. Figure 6.5 shows a high-level overview of the simulation procedure which will be explained in more detail in this section.

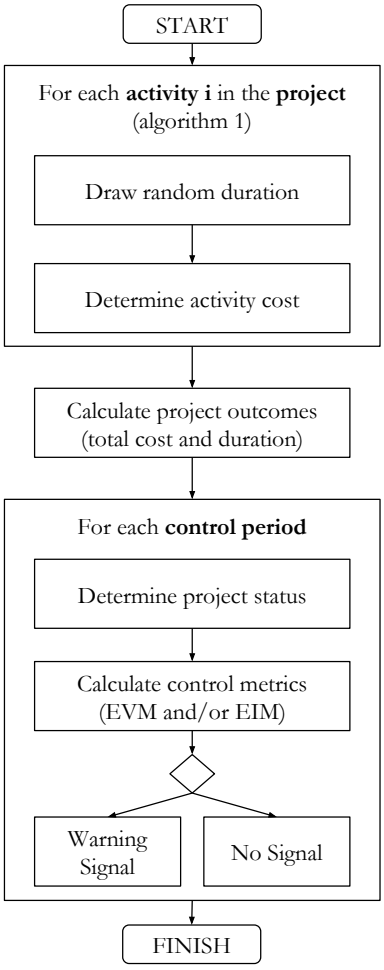


Figure 6.5: Schematic overview of the simulation procedure

The first step in the simulation procedure is the generation of activity durations and activity costs. The two-step procedure used to do this is shown in algorithm 6. First, the actual activity duration (D_i^{α}) for activity i is drawn at random from a triangular distribution. This distribution is defined by a mode, minimal and maximal value for

the activity duration (D_i^{mod}, $D_i^{\min}$ and $D_i^{\max}$ respectively). Next, the generated activity duration is used as one of the components to determine the actual activity cost (C_i^{α}). The actual activity cost is defined as an increasing function of the activity duration, since activity delays are likely to increase the costs associated with an activity. This function is multiplied by a random noise factor to represent possible inconsistencies in the relationship between activity duration and costs. In this equation $C_i^{\min}$ represents the cost associated with the shortest possible duration of activity i, ρ_i signifies the marginal cost increase when the activity have its maximal duration. The noise factor for activity is expressed as a factor δ_i which has to be situated in the range $]0, 1[$.

Algorithm 6 Simulate activity

1: **procedure** SIMULATEACTIVITY(i)
2: $D_i^{\alpha} \leftarrow Tri(D_i^{\min}, D_i^{\max}, D_i^{mod})$
3: $C_i^{\alpha} \leftarrow \left(C_i^{\min} + \frac{D_i^{\alpha} - D_i^{\min}}{D_i^{\max} - D_i^{\min}} \cdot \rho_i\right) \cdot [1 + u(-\delta_i, \delta_i)]$

Once the duration and cost for each activity has been determined, the total project cost and duration can be calculated. The former is simply equal to the sum of the costs of the individual activities ($C^{\alpha} = \sum_i^n C_i^{\alpha}$). The actual project duration (D^{α}) is calculated using a simple recursive procedure which assumes that all activities start as soon as possible.

As shown in figure 6.5, the final step in the simulation procedure is an iteration over the control periods, which are evenly spread over the planned duration (D^{π}) of the project. At each control period t, the status of the project has to be determined. Specifically, the accrued cost and value for the individual activities is calculated. This is done assuming a linear accrual of costs and value throughout the actual duration (D_i^{α}) of the activities.

This information can then be used in combination with a performance metric (see section 6.4) and a signal threshold to determine whether or not a signal is given to the actor controlling the project. The computational experiments presented in this research include four different performance metrics, which are explained below.

Basic EVM performance indicators: This controlling technique simply sets a threshold for the CPI and the $SPI(t)$ indicators as calculated using the traditional EVM and ES methodology (see section 6.2).

EVM forecasting: Rather than setting thresholds for the relative performance indicators, this method uses the EVM metrics to forecast the project duration and costs. At time interval t the duration of the project is forecast as $\tau + (D^{\pi} - ES_t)$. Where τ represents the current time. Similarly, the project cost is forecast as $C_t^{\alpha} + (C^{\pi} - EV_t)$ at time interval t. This technique gives a signal when the forecast duration or cost exceeds the thresholds set for either of these two criteria.

Separate EIM: The first technique for the EIM metric sets a threshold for the cost

and duration incentive variation respectively, as calculated by equations 6.18 and 6.19. If either of these two variations exceeds the predefined threshold, a signal is given to the controlling actor.

Joint EIM: Rather than setting thresholds for the cost and time objectives separately, this method sets a threshold for the total incentive variance (equation 6.20). In theory, this should allow for a perfect weighing of the cost and time performance, also allowing for good performance in one of these dimensions to compensate for deficiencies in the other dimension as long as the total incentive earnings are not at risk.

Rather than testing the performance of these techniques for an arbitrary threshold, a large number of thresholds spread over a reasonable range have been tested. For the basic EVM technique the following thresholds have been tested: 0.6, 0.625, 0.65, 0.675, 0.7, 0.725, 0.75, 0.775, 0.8, 0.825, 0.85, 0.875, 0.9, 0.925, 0.95, 0.975, 0.99, 0.995, 0.999, 1. (i.e. if the CPI or $SPI(t)$ exceeds this threshold a signal is produced.) For the EVM method with forecasting a similar approach is used: a signal is produced when the cost or duration forecast exceeds the planned cost (C^π) or duration (D^π) divided by x respectively, where x is equal to one of the numbers in the set of thresholds which is also used for the basic EVM technique.

Because the EIM technique does not use ratios to measure performance, the threshold used can not simply be an abstract value as for the EVM metrics. To get an adequate spread for the threshold values the project is simulated $1,000$ times in order to estimate the standard deviation for the project cost and duration. Using this information thresholds can be determined as follows:

$$THR^C = I^C \left(C^\pi + x \cdot \sigma(C^\alpha)\right) \tag{6.25}$$

$$THR^D = I^D \left(D^\pi + x \cdot \sigma(D^\alpha)\right) \tag{6.26}$$

$$THR^{tot} = THR^C + THR^D \tag{6.27}$$

Where THR^C and THR^D are the thresholds used for the first EIM technique which measures the performance for the cost and duration dimensions separately, and THR^{tot} is the threshold used for the second EIM technique which monitors global performance. These estimates are made by passing the cost and duration from the planned schedule (C^π and D^π) plus a fraction x of the observed standard deviation for these two quantities ($\sigma(C^\alpha)$ and $\sigma(D^\alpha)$) to the respective incentive equations (see equations 6.2 until 6.5). The tightness of these thresholds is determined by setting a value for the parameter x. For

		True condition	
		Project failure	**Project success**
Predicted status	**Failure (signal)**	True positive	False positive
	Success (no signal)	False negative	True negative

Table 6.5: Overview of possible scenarios

the experiments presented below, the x parameter has been given the following values: $\{0.1, 0.27, 0.44, ..., 3.33\}$.

Once these signals have been obtained, the next step is to judge their adequacy. The next section gives a detailed overview of the classification and metrics used for the signals derived from these calculations.

6.5.1.2 Output Metrics

The first step to judge whether or not a control metric correctly classifies the status of a project is to classify the potential outcomes of the project in two categories: failure and success. The criterion used for this is whether or not the total incentive amount earned is within an acceptable range of the planned incentive amount earned (see equations 6.11 and 6.12).

Similarly to the definition of the threshold for the EIM metrics (see section 6.5.1.1), the inherent cost and time risk of the project is taken into account in order to set the allowable deviation from the target incentives. These target incentive amounts can be expressed as $I^C(C^\pi)$ and $I^D(D^\pi)$ for cost and duration respectively. For this experiment, it was assumed that a deviation up to 50% of one standard deviation would be deemed within acceptable limits. As such, the lower bounds for the acceptable incentive amounts become equal to $I^C(C^\pi + 0.5 \cdot \sigma(C^\alpha))$ and $I^D(D^\pi + 0.5 \cdot \sigma(D^\alpha))$ for cost and duration respectively. The lowest total incentive earnings then simply become the sum of these two quantities. Hence, the project is considered to be successful if:

$$I^{C\alpha} + I^{D\alpha} \geq I^C(C^\pi + 0.5 \cdot \sigma(C^\alpha)) + I^D(D^\pi + 0.5 \cdot \sigma(D^\alpha)) \tag{6.28}$$

Given the binary nature of the project results (success/failure) and the control signal (warning/no warning) there are four possible results for a single control period. Table 6.5 summarises the possible combinations.

By observing the frequencies of observations in each of these categories across multiple simulations and project environments, a number of summary statistics can be calculated. The first of these is the true positive rate (TPR), which is the fraction of positive signals which rightfully indicate that the project is out of control (equation 6.29).

$$TPR = \frac{\sum \text{True positive}}{\sum \text{Project failure}} \tag{6.29}$$

A second summary metric is the false positive rate (FPR), or the likelihood of a positive signal being given when the project was actually under control (equation 6.30).

$$FPR = \frac{\sum \text{False positive}}{\sum \text{Project success}} \tag{6.30}$$

The third summary metric is the positive predictive value (PPV), which represents the fraction of positive signals rightfully indicating the project being out of control (equation 6.31).

$$PPV = \frac{\sum \text{True positive}}{\sum \text{Signal}} \tag{6.31}$$

Similarly, the negative predictive value (NPV) represents the likelihood of the project being under control when no signal is given (equation 6.32).

$$NPV = \frac{\sum \text{True negative}}{\sum \text{No signal}} \tag{6.32}$$

A final metric is the accuracy (ACC), which is the total fraction of project which have been correctly classified by the control technique (equation 6.33).

$$ACC = \frac{\sum \text{True positive} + \text{True negative}}{\sum \text{Total population}} \tag{6.33}$$

6.5.2 Dataset

An extensive dataset containing 4,200 projects has been created in order to test the performance of the proposed EIM methodology. The procedure used to create this dataset consists of four phases, which are explained in detail in the sections below. The first phase is the creation of a network structure for the project (section 6.5.2.1). Once this network structure has been defined, the properties of the individual activities are defined (section 6.5.2.2). This includes stochastic distributions which govern the time and cost performance of the activities. The third step in the procedure is the definition of the planned and break-even schedules which are used by the EIM methodology (section 6.5.2.3). In the forth and final phase, the incentive contract for the project is defined (section 6.5.2.4). The parameters used for the data generation procedure are summarised in table 6.6. In this table the notation $u(\cdot, \cdot)$ is used to represent a continuous uniform distribution.

Parameter	Meaning	Values
n	Number of non-dummy activities	$\{30, 60, 90, 120\}$
SP	Serial-Parallel indicator	$\{0.16, 0.33, 0.5, 0.66, 0.83\}$
D_i^{mod}	Mode of activity duration for activity i	$u(5, 25)$
$D_i^{\min}$	Minimal activity duration for activity i	$u(0.2, 0.9) \cdot D_i^{mod}$
$D_i^{\max}$	Maximal activity duration for activity i	$u(1.1, 2.0) \cdot D_i^{mod}$
$C_i^{\min}$	Minimal activity cost for activity i	$u(0.5, 10.0)$
ρ_i	Marginal cost for maximal time increase for activity i	$u(1.0, 10.0)$
δ_i	Cost noise factor for activity i	$u(0.05, 0.15)$
D_i^{β}	Duration of activity i in the break-even schedule	$u(0.5, 1.5) \cdot D_i^{mod}$
$-$	Contract type	$\{NL, L, 2PL, 3PL\}$
s_r^C	Cost sharing ratio for region r	$u(0.1, 0.9)$
v_r^D	Incentive per unit of time in region r	$u(0.1, 0.9) \cdot TV$
B_r^C, B_r^D	Bounds for piecewise linear cost or duration clause	spread evenly over simulated range
$I_{\max}^C$	Maximal incentive amount in non-linear cost clause	$u(0.1, 0.9) \cdot (C^{\beta} - C_{\min}^{simulated})$
$I_{\min}^C$	Minimal incentive amount in non-linear cost clause	$u(0.1, 0.9) \cdot (C_{\max}^{simulated} - C^{\beta})$
LB^C	Lower bound for non-linear cost clause	$C^{\beta} - C_{\min}^{simulated}$
UB^C	Upper bound for non-linear cost clause	$C_{\max}^{simulated} - C^{\beta}$
$I_{\max}^D$	Maximal incentive amount in non-linear duration clause	$u(0.1, 0.9) \cdot TV \cdot (D^{\beta} - D_{\min}^{simulated})$
$I_{\min}^D$	Minimal incentive amount in non-linear duration clause	$u(0.1, 0.9) \cdot TV \cdot (D_{\min}^{simulated} - D^{\beta})$
LB^D	Lower bound for non-linear duration clause	$D^{\beta} - D_{\min}^{simulated}$
UB^D	Upper bound for non-linear duration clause	$D_{\max}^{simulated} - D^{\beta}$

Table 6.6: Parameters used for data generation.

6.5.2.1 Network Structure

The network structure has been created using the *RanGen*2 tool which has been created by Vanhoucke et al. (2008). This tool takes the number of non-dummy activities and the serial-parallel (SP) indicator as input parameters in order to create a set of topologically diverse network structures. The SP-metric represents a continuum between completely serial (1) and completely parallel (0) project networks (Tavares et al., 2002). As shown in table 6.6, the project size was varied from 30 to 120 projects, and the SP-indicator was set to one of five different values. For each combination of project size and SP-indicator 210 projects are generated, resulting in a total of 4,200 projects.

6.5.2.2 Activity Stochasticity

Once the network structure has been created, the data generation procedure iterates over all the (non-dummy) activities to define the associated stochastic distributions (see section 6.5.1.1). As shown in the second part of table 6.6, the values for these parameters are all drawn from continuous uniform distributions. Note that the minimal and maximal durations are defined in relative terms to the mode of the activity duration.

6.5.2.3 Schedule Generation

As explained in section 6.4.1 the EIM requires three schedules. Specifically, the planned (π), break-even (β) and actual schedules (α) are used. Whereas the actual schedule is the result of the (simulated) project execution, the other two schedules have to be defined beforehand. In order for a schedule to be defined, both the duration and cost of all its activities have to be specified.

The activity durations for the planned schedule are simply equal to the mode of the activity duration distributions ($D_i^{\pi} = D_i^{mod}$). The associated costs (C_i^{π}) for this schedule are generated using the same procedure as shown in algorithm 6.

For the break-even schedule, the activity durations (D_i^{β}) are determined using a random variation around the mode of the activity duration distribution (see table 6.6). Again, the logic from algorithm 6 is used to associate a cost with this activity duration.

6.5.2.4 Incentive Contract Structure

The final step in the problem generation procedure is the creation of incentive agreements for the created projects and schedules. As explained in section 6.3, an incentive contract consists of multiple incentive clauses. For this research two incentive clauses are considered. These clauses are linked to the project cost and duration respectively.

Four different archetypes of incentive equations are considered for the dataset used in these experiments: non-linear (NL, see equations 6.4 and 6.5) linear (L), piecewise linear with two segments ($PL2$) and piecewise linear with three segments ($PL3$, see equations 6.2 and 6.3). Moreover, it is also assumed that the incentive clauses for the cost and duration dimensions have the same archetype (e.g. if the cost incentive is determined by a non-linear function, the duration incentive will be as well). The four archetypes used are spread evenly across the $4,200$ created projects (see section 6.5.2.1), resulting in a total of $1,050$ projects using each contract archetype.

Once the contract type has been determined, the parameters for the contract are defined. As per definition the target cost (C^t) and duration (D^t) are set equal to the cost and duration of the break-even schedule (C^{β} and D^{β} respectively).

The setting of the other parameters is based on the inherent variability of the project as observed from $1,000$ simulation runs of the actual schedule (see section 6.5.1.1). For the linear and piecewise linear contracts the magnitude of the incentive is determined by the s_r^C and v_r^D parameters respectively. Since the former is a relative fraction of the cost deviation which is allocated to the contractor, the value used can simply be drawn from a continuous uniform distribution, as indicated in table 6.6. To determine the magnitude of the duration incentive (v_r^D), a theoretical valuation of the duration dimension relative

to the cost dimension is defined as TV.

$$TV = u(0.5, 2.5) \cdot \frac{C_{\max}^{simulated} - C_{\min}^{simulated}}{D_{\max}^{simulated} - D_{\min}^{simulated}} \tag{6.34}$$

In equation 6.34, $C_{\max}^{simulated}$, $C_{\min}^{simulated}$, $D_{\max}^{simulated}$ and $D_{\min}^{simulated}$ represent the maximal and minimal outcomes of the simulation runs for the project's total cost and duration. The value of TV should be interpreted as the owner's monetary valuation per time unit. This value is determined in relative terms to the cost variability observed in the simulation runs. Hence, the time dimension can be anywhere from half as important to two and a half times as important as the cost dimension in the eyes of the project owner. For each region in the duration contract clause the actual fraction of this value awarded to the contractor as an incentive is again determined using a random draw from a continuous uniform distribution, as shown in table 6.6.

The bounds for piecewise linear contracts (B_r^C and B_r^D) are spread evenly across the range given by the minimal and maximal outcome of these simulations. i.e. If the cost clause uses a piecewise linear contract with two segments the value of B_2^C will be equal to $0.5 \cdot (C_{\max}^{simulated} - C_{\min}^{simulated})$.

For the non-linear contract forms, a similar approach is taken which uses the outcome of simulated project to estimate the variability of the project to define reasonable upper and lower bounds, as well as incentive amounts. The specific distributions used to do this are shown at the bottom of table 6.6.

6.5.3 Computational Results

For each of the 4,200 projects in the dataset (see section 6.5.2) 1,000 simulations runs have been performed. The project control metrics have been calculated in four evenly-spaced control periods for each of these simulations. Using the metrics described in section 6.5.1.2, the results of this computational experiment are discussed in this section. First, the overall performance of the different control techniques is discussed in section 6.5.3.1. Next, the sensitivity of these observations to the problem size, network structure and contract type is discussed in section 6.5.3.2.

6.5.3.1 Overall Performance

Figure 6.6 shows the receiver operator characteristic (ROC) curve for the four different control techniques for each of the four control periods. Since an optimal technique would have a true positive rate equal to 1 and a false positive rate equal to 0 the optimal region on these plots is the upper left corner of the plot. The diagonal from the lower left to the upper right corner represents the theoretical performance of a randomly generated signal.

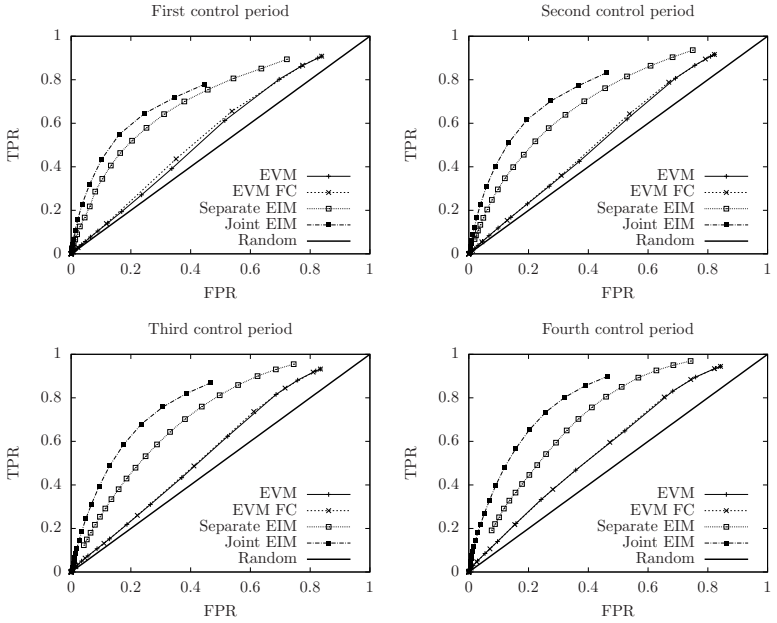


Figure 6.6: Receiver operating characteristic (ROC) chart

Hence, the performance of the binary classifiers can be judged by the surface area between their respective curve and the random diagonal. A number of conclusions can be drawn from these different plots.

(i) All methods perform better than a random signal. For every control period and every controlling technique the signals are better than a random signal, as indicated by the diagonal line on the curve.

(ii) The EIM-techniques consistently outperform the EVM-techniques. The performance of both the *EIM* techniques is consistently better than that of either of the *EVM*-based techniques, across all control periods.

(iii) The joint EIM-techniques perform better than the separate EIM techniques. None of the points on the separate *EIM* curves dominate the points on the joint *EIM* curves, whereas the opposite is true for a large fraction of the points. Hence, it is advisable to use the joint *EIM* technique that defines a single threshold based on the total incentives earned to monitor the project performance. An interesting consequence of this is that the effort needed to control the project is reduced (since only one rather than two metrics has to be checked), whilst also improving the performance of the project control.

(iv) The performance difference between EVM and EIM techniques becomes smaller as the project progresses. As the project progresses the performance of the *EVM*-based techniques becomes better, as indicated by the increased distance from the random di-

agonal. Nevertheless, the performance is still significantly worse than that of the EIM technique.

(v) The performance difference between the joint and separate EIM techniques increases as the project progresses. As shown in figure 6.6, the area between the joint and separate EIM curves increases as the project progresses.

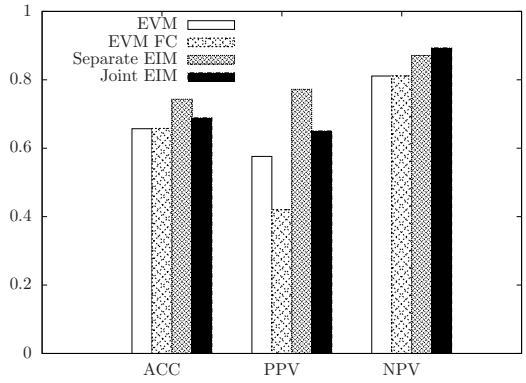


Figure 6.7: Accuracy (ACC), positive predictive value (PPV) and negative predictive value (NPV) averaged over all control periods.

A comparison of the accuracy as well as the positive and negative predictive values (see section 6.5.1.2) of the various techniques is made in figure 6.7. For this chart the best threshold value was selected for each technique for each of these metrics. For each of these metrics, the superior performance of the EIM-based signals is confirmed. When judged in terms of overall accuracy or positive predictive value it appears that the separate EIM thresholds outperform the joint approach.

6.5.3.2 Sensitivity Analysis

Figure 6.8 shows the influence of the project size, the project structure and the nature of the contract on the accuracy of the control signals. A first key observation is that the difference in performance between the various techniques appears to be approximately constant when the environment in which they operate varies.

Table 6.7 shows the results of an $ANOVA$ test that compared the means for each technique and for each of these environment parameters (only the p-values are reported, a large p-value indicating that the null hypothesis that all means are equal cannot be rejected). It is immediately apparent that changes in the serial-parallel structure or the type of contract do not influence the mean performance. There do however seem to be statistically significant differences for the EIM technique for increasing problem size.

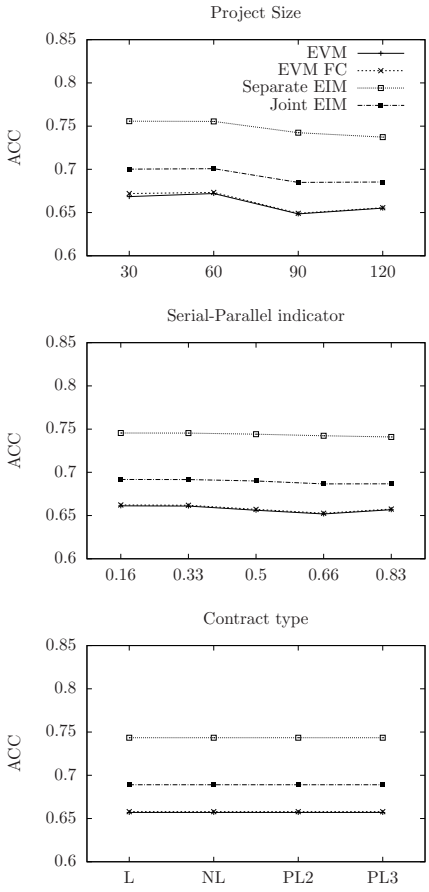


Figure 6.8: Impact of project and contract structure on the accuracy of control signals.

To further investigate this trend for the EIM indicators, a Tukey test is used to perform a pairwise comparison of the means for varying project sizes. The results of this test are summarised in table 6.8 (again only the p-values are reported here). From this table it is clear that the difference in the averages between adjacent points is not always significant, but that the differences between other points appear to be statistically significant. In spite of this, the magnitude of this change is only small and it is unlikely that this difference is substantial in practical applications.

The analysis of the performance sensitivity for the other performance measures (see section 6.5.1.2) revealed similar robustness of the project control techniques. Hence, it can be concluded that the observations made in section 6.5.3.1 are robust to changes in the project and contract structure.

Structure parameter	EVM	$EVMFC$	Separate EIM	Joint EIM
Project size	0.481	0.268	0.002	0.000
Serial-Parallel	0.969	0.942	0.917	0.620
Contract type	1.000	1.000	1.000	1.000

Table 6.7: p-values from $ANOVA$ analysis for project size, SP-indicator and contract type.

	separate EIM	Joint EIM
30-60	0.9998588	0.9707012
60-90	0.0889764	0.0006076
90-120	0.9413071	0.9792367
30-90	0.0745908	0.0031896
30-120	0.0155919	0.0118580
60-120	0.0194406	0.0026689

Table 6.8: Tukey multiple comparison procedure for different project sizes.

6.6 Conclusions

This research introduced a novel project control technique specifically tailored to projects subjected to incentives. This technique is an extension of the traditional EVM/ES techniques and is dubbed 'earned incentive management' (EIM). By taking this more dedicated approach to project control two key issues can be avoided. Firstly, the separate impact of the cost and duration dimensions can be correctly weighted, eliminating the need for dimension-specific metrics. Secondly, the EIM methodology is capable of monitoring non-increasing and irregular accrual of incentives throughout the project, resulting in more effective project control.

The advantages of the proposed control technique have been tested using a computational experiment which simulated the control performance on 4,200 projects with varying size and contract structures. The results of these experiments showed that the EIM technique significantly outperforms conventional EVM/ES methods, and should therefore be preferred in projects where cost and time incentives play a significant role.

Several opportunities for future research have also been uncovered, such as the adoption of the model for highly specific incentive contracts. Another avenue for future research is the extension of the control signals to create probabilistic estimators, which could provide a more detailed risk analysis for an incentivised project. The model could also be extended to include incentives linked to dimensions other than time and cost such as scope, quality or safety. Another extension could be to apply the incentives logic to less clearly defined projects where making a very detailed a priori plan is impossible (as was done by Lévárdy and Browning (2009)).

6.A Overview of Notation

A comprehensive overview of the notation introduced in this research is given by table 6.9.

Index	**Description**
$i \in V = \{1, ..., n\}$	Activity index
$t \in \{1, ..., T\}$	Time index (control periods)
$r \in \{1, ..., R\}$	Contract region
Contract Variables	**Description**
$I^C(\cdot)$, $I^D(\cdot)$, I^{tot}	Cost, duration and total incentive awarded
C^t, D^t	Cost and duration target as specified in the contract
s_r^C	Cost sharing fraction for contract region r
v_r^D	Awarded incentive per unit of time in region r
B_r^C, B_r^D	Lower bound of region r in piecewise linear cost or duration contract
UB^C, UB^D	Upper bound of incentive spread in non-linear cost or duration contract
LB^C, LB^D	Lower bound of incentive spread in non-linear cost or duration contract
$I^C_{\max}$, $I^D_{\max}$	Maximal cost and duration incentive amount for non-linear contract
$I^C_{\min}$, $I^D_{\min}$	Minimal cost and duration incentive amount for non-linear contract
Project Control Variables	**Description**
β	Break-even schedule
π	Planned execution project schedule
α	Actual observed execution of the project
C^β, C^π, C^α	Total project cost for respective schedules
C_i^β, C_i^π, C_i^α	Cost associated with activity i for respective schedules
C_t^β, C_t^π, C_t^α	Total cost at a time interval t for respective schedules
D^β, D^π, D^α	Total project duration for respective schedules
D_i^β, D_i^π, D_i^α	Duration associated with activity i for respective schedules
$I^{C\pi}$, $I^{C\alpha}$	Cost incentive earned for planned and actual schedule respectively
$I^{D\pi}$, $I^{D\alpha}$	Duration incentive earned for planned and actual schedule respectively
PC_{it}^β, PC_{it}^π, PC_{it}^α	Percentage completed for activity i at time t in the respective schedules
PV_t^β	Planned value at time interval t in the break-even schedule
EV_t^π,EV_t^α	Earned value at time interval t of the planned and actual schedule respectively
ES_t^π,ES_t^α	Earned schedule at time interval t of the planned and actual schedule respectively
CV_t^π,CV_t^α	Cost variance at time interval t for the planned and actual schedule respectively
SV_t^π,SV_t^α	Schedule variance at time interval t for the planned and actual schedule respectively
$IV(C)_t$, $IV(D)_t$, $IV(tot)_t$	Cost, duration and total incentive variance at time interval t

Table 6.9: Overview of notation

6.B A Summary of Traditional Earned Value Management

Earned value management relies on a number of straightforward metrics to measure the time and cost performance of a project. The baseline against which the performance of the project is compared is dubbed the 'planned value' (PV_t). This is a time-phased budget baseline that equals the planned cumulative expenses at every time period. Plotting the planned value as a function of time results in the frequently encountered 'S-curve' (see top left panel in figure 6.4). The calculation of the planned value for time period t can be expressed as follows:

$$PV_t = \sum_{i=1}^{n} PC_{it}^{\pi} \cdot C_i^{\pi} \tag{6.35}$$

Where PC_{it}^{π} is the planned percentage of work completed for activity i at time period t and C_i^{π} is the planned total cost of activity i.

During the execution of the project two types of information are gathered. The first is the actually incurred cost (AC_t), which is the result of straightforward cost accounting. Secondly, the value of the amount of work which has been completed is tracked. This metric is dubbed 'earned value' (EV_t) and is calculated by multiplying the percentage completed by the planned cost for the activity. Using the notation used in the research this becomes:

$$EV_t = \sum_{i=1}^{n} PC_{it}^{\alpha} \cdot C_i^{\pi} \tag{6.36}$$

Where PC_{it}^{α} is the fraction of the work for activity i completed in time period t of the actual project execution, and C_i^{π} is the activity cost as initially planned.

These three measures (PV_t, AC_t and EV_t) form the basis for all metrics used in earned value management. The cost performance of the project can be expressed by the cost variance (CV_t) or the cost performance index (CPI_t) as follows:

$$CV_t = EV_t - AC_t \tag{6.37}$$

$$CPI_t = \frac{EV_t}{AC_t} \tag{6.38}$$

A positive value for the cost variance (CV_t) indicates better than expected cost performance, whereas a negative value indicates that the costs are higher than initially expected. The cost performance index (CPI_t) is interpreted in a similar manner, a value below unity indicating worse than expected cost performance, the inverse being true for a value above unity.

The time performance can be analysed in a similar manner, resulting in a schedule variance (SV_t) and schedule performance index (SPI_t) which can be calculated as follows:

$$SV_t = EV_t - PV_t \tag{6.39}$$

$$SPI_t = \frac{EV_t}{PV_t} \tag{6.40}$$

The interpretation of these metrics is analogous to the cost metrics. A negative value

for the schedule variance (SV_t) indicates that the project is behind schedule, a positive value that the project is progressing faster than originally planned. The same is true for values below and above unity for the SPI_t metric.

These traditional time performance metrics have two major shortcomings. Firstly, they measure time performance as a monetary amount, which can be quite hard to interpret correctly. Secondly, the schedule performance index (SPI_t) converges to unity at the end of the project, which may convey an incorrect signal that the time performance is improving when this is not the case in reality. To mitigate these issues, the earned schedule (ES_t) metric has been proposed by Lipke (2003). This approach translates the earned value to time units by calculating the fraction of the original schedule that has been earned at a specific point in time. The earned schedule metric at time period t is calculated as follows:

$$ES_t = \tau + \frac{EV_t - PV_\tau}{PV_{\tau+1} - PV_\tau} \tag{6.41}$$

where τ is an integer value which satisfies the following criterion:

$$PV_\tau \leq EV_t < PV_{\tau+1} \tag{6.42}$$

The schedule variance (SV_t) and schedule performance index (SPI_t) can then be recalculated using the earned schedule methodology as shown in equations 6.43 and 6.44. The conventional notation adds (t) to indicate that these metrics have been calculated using the earned schedule formulae. This is not to be confused with the index t that indicates the time period.

$$SV(t)_t = ES_t - t \tag{6.43}$$

$$SPI(t)_t = \frac{ES_t}{t} \tag{6.44}$$

Again, the schedule variance ($SV(t)_t$) is negative when the project's time performance is worse than expected and positive when the opposite is true. Similarly, $SPI(t)_t$ is below unity when the project is behind schedule and above unity when the time performance is better than expected.

The metrics which have been summarised in this appendix can be used as the basis for forecasting and control systems, as is evident from the substantial amount of literature on the subject (see section 6.2). The specifics of these applications fall outside the scope of this appendix, and the reader is referred to the following authors for a more comprehensive overview of the subject matter: Anbari (2003); Fleming and Koppelman (2005); Vanhoucke (2010b, 2014).

Part II

Schedule optimisation for weather-sensitive projects

7

Modelling the Weather

This chapter presents an overview of weather modelling principles, and proposes a new weather model which is stylised specifically for operational relevance. Specifically, this simulation model is capable of simulating both wind speed and wave height simultaneously. As such, this model can be employed to improve the quality of operational simulations, specifically for offshore operations, which is the subject of chapters 8 and 9.

7.1 Introduction

Accurately estimating the weather has been a goal in operations management for over a century. The majority of models which have been created for this purpose have focussed on predicting precipitation, which is of major importance to agriculture and many other industries (Wilks and Wilby, 1999).

For offshore construction there are two key weather factors which determine whether working is possible or not: the significant wave height and the average wind speed (Graham, 1982). Depending on the nature of an activity, an acceptable threshold for both these dimensions can be defined. If this threshold is exceeded, the activity has to be interrupted and resumed when weather conditions have improved. Such interruptions can of course be highly detrimental to project performance.

This chapter starts by giving an overview of the various types of weather models in section 7.2. This overview focuses on models that are capable of generating realistic weather patterns, rather than models with strictly meteorological objectives. Next, a novel weather model specifically tailored for use in offshore projects is presented in section 7.3.

7.2 A Brief Overview of Weather Modelling Techniques

7.2.1 Types of Weather Models

Two different kinds of weather models can be distinguished: weather simulation models and weather forecasting models. Simulation models can be viewed as a highly complex random number generators whose outputs resemble weather conditions at a certain location. However, these random sequences are in no way meant to be a forecast of what is going to happen; i.e. the weather pattern remains completely fictitious, but is assumed to have similar properties to weather at the location. Weather forecasting models on the other hand operate from a deterministic point of view and try to make short term predictions of what the weather will be like (Wilks and Wilby, 1999).

Both simulation and forecasting models are highly useful when managing offshore construction models. A good forecasting model can help project management with day-to-day operational decisions, allowing them to delay work when weather conditions are expected to be suboptimal. Weather simulation on the other hand can help to optimise the scheduling of a project in order to make it more robust with regards to weather influences. This robustness is highly important since offshore construction projects are highly capital intensive due to the use of expensive machinery. Impromptu leasing of this machinery is not possible, hence it is highly important to make accurate predictions of when this machinery will be needed as well as accurate estimates of the duration for which this machinery will

be needed.

Both simulation and forecasting models can be used for either meteorological or economical purposes. Where the focus of the former lies on providing an explanation for a physical phenomenon, the latter is interested in the impact of weather on a certain process or activity. This distinction obviously has a major impact on the way the model is constructed. Models wishing to explain meteorological phenomena usually use much smaller time steps (less than one hour) than models used for economical purposes, which almost always use time steps between one and six hours, and in some cases even more (Monbet et al., 2007). This different time step size also impacts the choice of modelling techniques. An example of this are Markov Chains which have been shown to be less adequate for weather models taking time steps of less than 40 minutes (Brokish and Kirtley, 2009).

Monbet et al. (2007) also highlight the importance of the trade-off between a model's ease of use and the accuracy of a model. A model's ease of use can be expressed using a number of qualitative criteria. Firstly, the robustness to the data source which may contain a considerable number of missing values. Secondly, the ease of use also depends on the robustness to the process itself. Some models are easier to use for a broader set of processes than others, especially for weather models where complex correlation between different elements may occur. Finally, the mathematical complexity and the time needed to implement such a model is also important. A model should not be needlessly complex thereby squandering scarce resources for its implementation when a simpler model would have sufficed. The accuracy of a model signifies the degree to which the statistical properties of the model correspond to the statistical properties of the original data. This includes various statistical properties such as the average, standard deviation, seasonality and autocorrelation.

7.2.2 Dealing with Non-stationary Data

Many traditional forecasting and simulation models such as Markov chains and autoregressive models are valid under the assumption that the underlying process is stationary[1]. Most weather parameters however are subject to strong seasonal influences. Moreover, some weather variables such as wind can differ depending on the time of day. To deal with this issue many authors have proposed methods to transform this non-stationary data into stationary or quasi-stationary processes.

There are two ways in which this issue can be resolved (Monbet et al., 2007). The first method uses statistical techniques to remove the seasonal component from the data, the resulting series is then stationary and the aforementioned techniques can then be used

[1] A stationary process is a process whose joint probability does not change when the process is shifted in time.

(Monbet et al., 2007; Stefanakos and Athanassoulis, 2002; Stefanakos and Belibassakis, 2005). A potential issue here is that these statistical techniques may not succeed in removing all the non-stationarity from the data. An alternative method is fitting a separate model for each season, assuming that stationarity can be assumed within a certain season. The most important drawback of this method is the need for substantially more data since a model has to be estimated for each season. Furthermore, this method also introduces artificial breaks in the data, which are of course not present in reality.

When using the first method for dealing with non-stationary data, the transformation which changes data into stationary data usually takes the following form (using the notation of Monbet et al. (2007)):

$$Y_t = m(t) + \sigma(t) Y_t^{stat} \tag{7.1}$$

Where Y_t is the original non-stationary process, $m(t)$ is a deterministic function describing the average value for time period t, $\sigma(t)$ is a deterministic function for the standard deviation in time period t. The Y_t^{stat} is then the underlying stationary process which can be used in a wide range of different modelling approaches. An important note is that weather data can exhibit non-stationary behaviour on both a seasonal and an intra-daily scale. To be able to assume a truly stationary process both of these have to be removed.

Cunha and Guedes Soares (1999) compare different methods for removing seasonality prior to fitting an autoregressive moving average (ARMA) model. For simulation purposes they advise using a Box-Cox transformation with $\lambda = 0$[2] to obtain variance stationarity and then correcting for changes in the average using a Fourier model.

7.2.3 Modelling Techniques

This section presents a short overview of weather modelling techniques, discussing their key advantages and drawbacks. These approaches are grouped using the categorisation used by Monbet et al. (2007), who distinguish Gaussian-based parametric techniques, non-Gaussian based parametric techniques and non-parametric techniques.

7.2.3.1 Gaussian-based Parametric Techniques

Gaussion-based parametric techniques remove seasonality from data using statistical techniques, such as Box-Cox and Fourrier transformations (Cunha and Guedes Soares, 1999). Once this transformation is complete, a wide variety of techniques such as autoregressive (AR) models or translated Gaussian processes are fitted to the deseasonalized pattern. Because all weather data is used to estimate a single model, relatively little data is

[2]This is in effect taking the natural logarithm $\ln(Y_t)$.

needed to estimate parameters when compared to techniques such as Markov chains which typically split the available data into seasons which are modelled separately. Moreover, Gaussian-based techniques are capable of identifying trends in potentially noisy weather data. Nevertheless, these techniques fall short when multiple weather components (i.e. wind, wave height, precipitation, etc.) have to be modelled simultaneously (Monbet et al., 2007).

7.2.3.2 Non-Gaussian-based Parametric Techniques

Several other parametric techniques have been used to model weather. Finite state space Markov chains and their variants are undoubtedly the most widely adopted for offshore weather simulation purposes. Rothkopf et al. (1974) present a first-order Markovian model which defines a number of sea states and define a transition matrix based on the locality and season in which operations are carried out. The authors also note that a more accurate representation of the weather conditions can be obtained by increasing the number of states considered in the Markov weather model, insofar as the quantity of available observations allows this.

The persistence of good and bad weather spells is of paramount importance to offshore construction projects. Because of this, a number of authors have focussed on modelling this persistence and incorporating it in simulation models. Graham (1982) describes how weather simulations can be made to take into account persistence by using a transition matrix in combination with two-parameter Weibull distributions which model persistence of a certain state.

Bowers and Mould (1994) specifically investigate weather risk for offshore construction projects. The model used to simulate weather conditions combines a Markov chain analogous to the one presented in Graham (1982) with a gamma distribution. In this model six different transition matrices are used to represent the six main seasons in which the simulation operates. Each of these transition matrices defines the possibility of transferring between one of 5 possible states. Because the persistence behaviour of the Markovian model in the first (calmest) of the five states did not correspond to the historic data, a gamma distribution was used to randomly draw the persistence upon arrival in the first state. Another example of the use of Markov chains to simulate wave height is presented by Norlund et al. (2015), who used the technique to create robust and more environmentally friendly schedules for offshore supply vessels.

Markov chains have also been used to simulate wind rather than significant wave heights. A study by Sahin and Sen (2001) shows that even a very simple first order Markov chain with eight states can attain an adequate level of accuracy. Ettoumi et al. (2003) used Markov chains in combination with a Weibull distribution to model both wind

speed and wind direction. For the wind speed each of the discrete states of the transfer matrix is linked to a certain Weibull distribution, from which the exact wind speed is then drawn at random. Second order Markov chains were used by Shamshad et al. (2005) to make short-term predictions of wind speed. The authors noted that there is a slight improvement in terms of autocorrelation when using a second order rather than first order Markov chain model.

Brokish and Kirtley (2009) make an important observation with regard to the use of Markov Chains used to simulate wind power, namely that this technique is often unable to match the autocorrelation of the original distribution when used for time steps which are shorter than 40 minutes. Hence, this technique should only be used for situations with longer time steps.

Hidden Markov models have also been used successfully to model weather conditions, one example of this is the Markov Switching Autoregressive model (MS-AR) which was proposed by Ailliot et al. (2006).

Another parametric technique which is often used for weather prediction weather is an artificial neural network (ANN). More and Deo (2003) used artificial neural networks to forecast wind speeds in coastal regions and found that these artificial intelligence techniques outperform traditional ARIMA techniques. The usefulness of this technique for weather prediction has also been illustrated by Arena and Puca (2004) who have used this technique to reconstruct significant wave heights based on correlated data from other measurement stations. Makarynskyy et al. (2005) also used ANN to predict wave height and period in order to deliver a warning signal up to 24 hours in advance. More recently Mandal and Prabaharan (2010) have also used the ANN technique to make short term predictions for significant wave height.

Non-linear autoregressive models are another technique which may be used rather than the traditional Gaussian techniques. Due to the fact that the underlying process is inherently non-linear, non-linear models can improve the estimation of the higher statistical moments such as skew and kurtosis (Scotto and Guedes Soares, 2000).

7.2.3.3 Non-parametric Techniques

The use of non-parametric techniques is considerably less widespread than that of parametric techniques. One technique which is sporadically used for making bootstrap estimations is re-sampling (Monbet et al., 2007). This technique consists of a nearest neighbour heuristic with a random component to create weather patterns which are similar to the patterns observed in the original data. The advantage of this technique being that no assumptions have to be made regarding the stationarity of the underlying data. Hence, it is unneeded to construct separate models for different seasons, nor is it required to subject the data to

transformations to remove seasonal patterns. The key disadvantage of this technique is the significant risk of overfitting, resulting in weather patterns which resemble the observed data too closely.

Resampling is the most frequently used non-parametric technique, but its use is generally limited to bootstrap estimations. The technique itself uses a nearest neighbour heuristic with a random component to create similar weather patterns as in the original data. The key advantage of this technique is that no assumptions have to be made regarding the stationarity of the original data.

7.2.4 Concluding Remarks

Based on this review of existing weather simulation models in section, a Markovian model capable of simulating both wave and wind conditions is proposed in this chapter. This capability to create realistic simulations of both weather parameters which impact the construction activities is the main motivation for choosing this type of model. The key disadvantage of such a model being that it needs a considerable amount of data to be fitted adequately. Nevertheless, since large datasets containing both wind and wave information this was not an issue for this research. Similarly, the limitations regarding the minimal step size imposed by this type of model were not operationally relevant for the offshore construction projects investigated in this research.

7.3 Weather Simulation Model

Using historic wind speed and significant wave height data collected off the Belgian coast in the North Sea[3], a simulation model was created. This model uses a combination of transition probability matrices and Weibull distributions to replicate a realistic combination of wind and wave conditions faced by offshore construction projects in the North sea.

The main difficulty encountered when creating this model was simultaneously guaranteeing the right correlation of wind speed and wave height, as well as the individual autocorrelation of both these variates. This was achieved by defining conditional probabilities that link these variables to each other as well as to past observations.

Because the main objective of this simulation is to determine the economical impact of the weather conditions, rather than a detailed description of meteorological phenomena, both wind and wave data have been transformed into discrete states. The evolution of these weather states is described using four-hour intervals. This interval is somewhat

[3]This data was collected by the Measurement Network Flemish Banks, specifically the data from the "Westhinder" measuring pile was used since this location most accurately reflects weather conditions for offshore wind farms off the Belgian coast. More information can be found at: `http://www.vliz.be/en/measurement-network-flemish-banks`

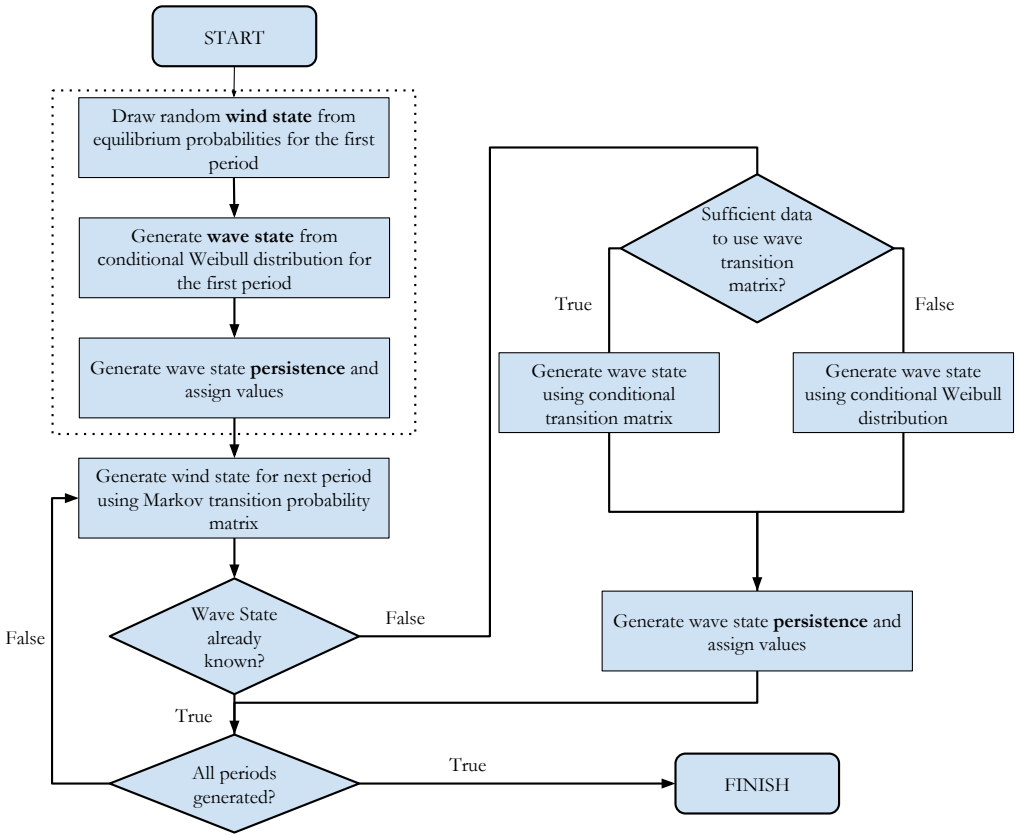


Figure 7.1: Flowchart of the weather generation model.

longer than that of models with meteorological objectives. The reason for this is dual: Firstly, very short time intervals are not relevant from an economic perspective. Secondly, using longer time intervals allows for a better approximation of autocorrelation.

Figure 7.1 shows the main flow of the weather simulation algorithm. Because substantially more wind data was available[4], the starting point is the generation of the wind speeds. These wind speeds then form the basis to generate associated wave heights.

7.3.1 Wind Speed

The simulation of wind states is done using separate first-order Markov chains for every month of the year, to account for the inherent non-stationarity of the data. Each of these models uses a transition matrix which defines a probability p_{ij}^{wind}, signifying the likelihood of moving from wind state i to wind state j during a certain season. Naturally these probabilities have to satisfy the following property:

[4] 4,897 days of wind data and 1,396 days of wave data were available at the time of this research.

$$\sum_{j=1}^{n} p_{ij}^{wind} = 1, \qquad \forall i \tag{7.2}$$

Where n signifies the total number of possible wind states. This property simply states that for every wind state i, the sum of all the transition probabilities must be exactly equal to 1. The exact value for these probabilities (p_{ij}^{wind}) has been calculated for every month based on the available weather data.

Given that these Markov models for the wind states are ergodic (i.e. any state can be reached from any other state in a finite number of steps), it is possible to calculate the equilibrium probabilities by means of simulation. These simulations start at a random point and perform a certain number of iterations (ι), after which the frequencies at which specific states were visited are transformed into probabilities. After the probabilities have been updated the steady state estimates are compared to the previous estimates, if the change exceeds a preset threshold the simulation continues for another ι iterations. For the model presented in this chapter a threshold of 0.1% and an $\iota = 100$ was used. The steady state probabilities obtained using this method are used to generate a starting point for the weather generation model (see figure 7.1).

A trade-off has to be made between the amount of information conveyed by the simulation by increasing the number of wind states, and the accuracy of the model, which decreases as the number of wind states increases because a larger number of parameters has to be estimated. The calibrations of the simulation model have indicated that the accuracy of the model starts to decrease when more than ten wind states are used. Hence, the total number of wind states has been set to ten ($n = 10$). Note that the bounds for these states are simply the n-th percentiles of the available data.

7.3.2 Significant Wave Height

The simulation of realistic wave heights to accompany the wind speeds is more complex for a number of reasons: (i) the wave state has to be correlated with the wind state, (ii) the wave state also has to have the right level of autocorrelation, (iii) the persistence of a wave state has to be realistic, and (iv) the limited amount of data points - especially for rare weather conditions - for fitting have to be taken into account.

To accommodate all these aspects, a combination of three techniques is used: two of which simulate realistic transitions between different wave states, and a third which simulates the persistence of a wave state once it is attained. The difference between the two techniques which simulate transitions is that the first one takes into account both the correlation with the wind state and the autocorrelation of the wave states, whereas the second technique only takes into account the correlation with the wind state. Which of

these techniques is used is a response to the amount of data which is available to estimate the transition probabilities; in case insufficient data is available to use the first model the second model is used.

The first model which simulates wave state transitions uses a variation on a second-order Markov chain. This variation defines the possibility of transitioning to another wave state in period t based on the preceding wave state from period $t-1$, as well as the already generated wind state for period t. Hence, a three-dimensional matrix containing transition probabilities p_{kli}^{wave} is created. The value of p_{kli}^{wave} represents the probability of moving from wave state k in period t to wave state l in period $t+1$, given that the wind state (which has already been simulated using the wind state Markov chain) in period $t+1$ is equal to i. Again, for every possible value of k and i, the sum over all possible l values ($l = 1, ..., m$) has to equal unity:

$$\sum_{l=1}^{m} p_{kli}^{wave} = 1, \qquad \forall k, i \tag{7.3}$$

Because the persistence of these wave states is modelled separately, the probability of moving to the same wave state is set to zero. Using the same notation this implies that:

$$p_{kki}^{wave} = 0, \qquad \forall k, i \tag{7.4}$$

By using this transition matrix, the correlation between the wind and wave data, as well as the autocorrelation of the wave data can be guaranteed. However, the drawback of this method is that it requires a lot ($n^2 \cdot m$) of probabilities to be estimated. Especially for rare scenarios where winds are relatively high and waves relatively are low or vice versa it may not be possible to estimate realistic probabilities by simply fitting them to the available data because no such observations exist. Nevertheless, the simulation model should be capable of dealing with these unlikely scenarios. This is the reason for the introduction of the second method for generating correlated wave states.

The second method for simulating transitions between wave states only takes the correlation between the wind and wave states into account. Depending on the wind state which has already been simulated, the accompanying wave state is drawn from one of n Weibull distributions. These n distributions have been fitted to subsets of the wave data, based on the wind state which accompanied the specific wave state. Assume for example that a wave state has to be simulated, given the fact that the wind state for the specific period has already been simulated and is equal to 2. Moreover, there is insufficient data to accurately estimate parameters using the first model. Hence, the wave state will be randomly drawn from a Weibull distribution whose parameters have been fitted to the wave heights which occurred when the wind state was also equal to 2. Note that this Weibull

model is both fitted to and simulates continuous data rather than the discrete wave states, requiring the wave state to be determined after a continuous Weibull distributed random variable has been returned.

Similarly to the first method of generating the wave state transitions, the Weibull model also does not allow the wave state to remain unchanged. Should an identical wave state be drawn from the distribution, the draw is discarded and a new Weibull distributed random number is drawn.

The persistence of good and bad weather conditions is one of the most important aspects of a weather simulation model (Bowers and Mould, 1994; Graham, 1982). Hence, one of the major goals of this situation model was to give an accurate representation of the persistence of weather conditions. Preliminary experiments which used the previously described models to simulate both transitions to different and the same wave state concluded that these techniques could not guarantee a realistic persistence of the wave states. This is caused by the correlation which is introduced between the wind and wave states.

This issue was solved by generating the persistence of a wave state separately. This is done by creating a persistence probability matrix for every wave state based on the available wave data. This matrix consists of probabilities $p_{kt}^{persist}$, which signify the probability that wave state k persists for t time periods. Naturally, the following must be true:

$$\sum_{t=1}^{t_{\max}} p_{kt}^{persist} = 1 \qquad \forall k \tag{7.5}$$

Where $t^{\max}$ represents the maximal number of time periods a wave state can persist. Using these conditional probabilities, the persistence of every wave state is modelled. Analysis shows that by using this method of simulation satisfactory results are obtained for the autocorrelation of both wind and wave states (see appendix 7.A).

The cost of this improved autocorrelation of the wave states is a slight decrease of the correlation between wind and wave states. In the original data a correlation of 0.8 was noted between wind and wave states. Due to the fact that the persistence of a wave state is assumed to be independent from the wind states, the global correlation between these two variates is reduced to 0.7. Insufficient data inhibits the use of a correction of the Weibull distribution without overfitting the available data points. Artificially inflating the correlation of the wind and wave states on the other hand results in a substantial reduction of the frequency of the wave states.

During the calibration of the simulation algorithm it was concluded that setting the number of available wave states to ten ($m = 10$) provided the best compromise between model complexity and accuracy in representing the weather patterns.

7.4 Conclusion

This section has presented an overview of weather modelling techniques designed to simulate realistic weather patterns and has proposed a novel weather model specifically tailored to the operational challenges in offshore projects. This model distinguishes itself from existing weather simulation models by its capability of simulating the correlation between wind- and wave states accurately with a long time horizon. Traditional weather models generally focus on a single weather dimension (such as precipitation), or on the prediction of the short term evolution of a number of different weather dimensions. For the latter case the autocorrelation of these measures plays a much stronger role than the correlation between the various dimensions. Hence, separate models are generally applied for separate dimensions. The model proposed in this chapter departs from this and is therefore capable of generating realistic long-term weather evolutions for operational scenarios where these elements have a joint impact on the process under study.

The usefulness of this model, and its integration in operationally and economically relevant simulation procedures will be discussed in chapters 8 and 9.

7.A Weather Model Persistence Plots

As was noted in the literature review (see section 7.2), the persistence of weather conditions is an important indicator of the quality of a weather model. Moreover, this aspect of the weather model is also of great importance to accurately estimate the influence of weather conditions on activity durations.

Figure 7.2 and 7.3 show how the persistence of wind- and wavestates of the weather generation model compare to the actual persistence observed from the empiric data gathered from the offshore measuring stations. Each of these figures shows a separate chart for each of the ten possible states for wind and waves respectively. These charts show the cumulative probability of remaining in a certain wind- or wavestate for a certain number of periods. From these charts it can be concluded that the simulation model is capable of modelling the persistence of both wind- and wave conditions with great accuracy.

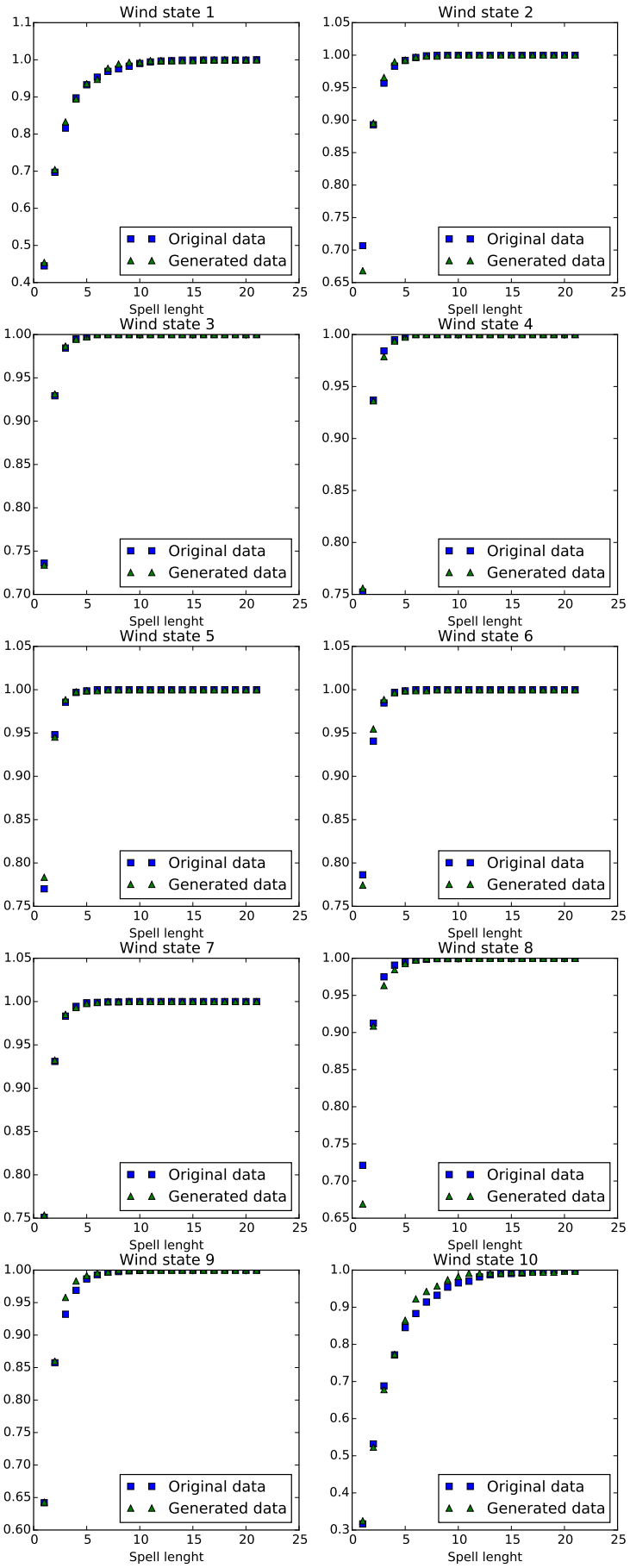


Figure 7.2: Wind state persistence

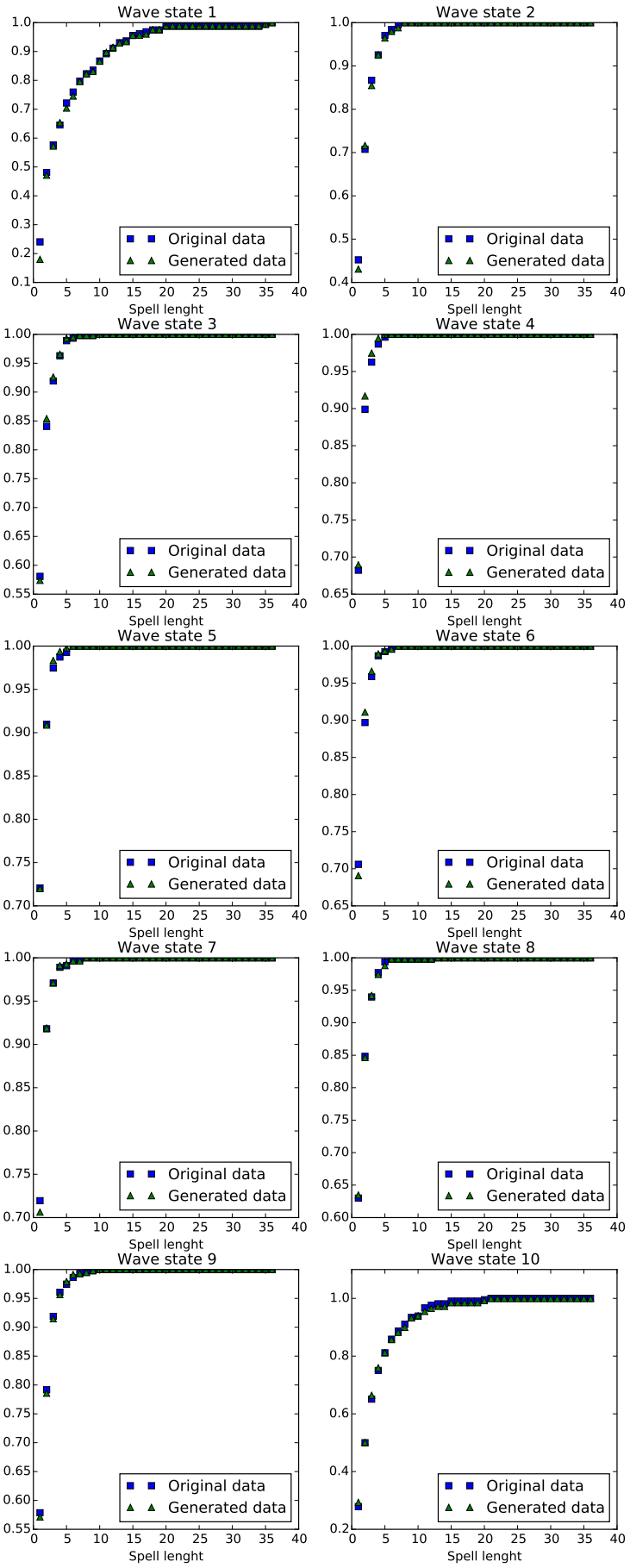


Figure 7.3: Wave state persistence

8

Optimised Scheduling for Weather Sensitive Offshore Construction Projects

This chapter presents a comprehensive methodology for the proactive scheduling of capital intensive projects that are sensitive to weather conditions. This is done by combining heuristic optimisation techniques and weather simulation models which are fitted to data gathered on the North Sea off the Belgian coast. The construction of an offshore wind farm will be used as an archetypical example of such a a capital-intensive and weather-sensitive project.

8.1 Introduction

The significant lead times and costs associated with materials and equipment in combination with intrinsic and weather related variability render the planning of offshore construction projects highly complex. Moreover, the way in which scarce resources are managed has a profound impact on both the cost and the completion date of a project. Hence, schedule quality is of paramount importance to the profitability of the project. A prerequisite to the creation of good schedules is the accuracy of the procedure used to estimate the project outcome when a given schedule is used. Because of the systematic influence of weather conditions, traditional Monte Carlo simulations that only include activity-related stochasticity fail to produce a reliable estimate of the project outcomes. Hence, the first objective of this research is to improve the accuracy of the project simulation by creating a procedure which includes both uncertainty related to the activities as well as an integrated model of the weather conditions. The weather model presented in chapter 7 has been designed to create realistically correlated wind- and weather conditions for operationally relevant time intervals.

The second objective of this research is to optimise the project planning itself by using both general meta-heuristic optimisation approaches and dedicated heuristics which have been specifically designed for the problem at hand. The performance of these heuristics is judged by the expected net present value of the project. The approach presented in this chapter is tested on real data from the construction of an offshore wind farm off the Belgian coast and weather data gathered by the Flanders Marine Institute using measuring poles in the North Sea.

8.2 Literature

This study combines techniques from project scheduling literature with insights from weather modelling, specifically stochastic weather generation models. This was motivated by the arguments made by Regnier (2008), who stated that recent advances in meteorological sciences have created substantial opportunities for quantitative modelling. Specifically, the OR/MS community could make significant contributions by tailoring meteorological products to the users' decision contexts. Incidentally, this is the main objective of this research.

A review of the weather modelling literature was given in chapter 7, where the weather model used for this research was presented. This section focuses on the contributions from the field of project management. This encompasses recent advances in dealing with uncertain activity durations, resource constraints and non-regular objectives which more

accurately represent the financial implications of project scheduling. This foundation is then used to construct a model for large-scale offshore construction projects in section 8.3.1. As was noted earlier, these theoretical foundations are supplemented with information from offshore wind projects (see section 8.5).

Offshore construction projects present one of the most challenging operational environments. Within this environment, the operational strategy can significantly impact the profitability, robustness and the environmental impact of a project (Norlund et al., 2015), and even prevent the loss of lives (Qian et al., 2012, 2015). This section highlights the specific challenges faced when scheduling offshore construction projects. This is done by presenting a concise review of relevant project scheduling literature. This literature forms the basis for the project model presented in section 8.3.1.

First and foremost, offshore projects are highly susceptible to adverse weather conditions. Rothkopf et al. (1974) and Graham (1982) highlight the importance of extended periods of good weather for the execution of certain activities, as well as the significant start-up cost which is incurred when an activity has to be interrupted due to harsh weather conditions. Hence, the project manager will try to optimise his project in light of the weather risks (s)he is facing. Bowers and Mould (1994) have analysed how this can be done by either delaying the start of the project, or by using more expensive equipment which is less susceptible to extreme weather conditions. Obviously, many other strategies can also be used to improve the expected performance of a project.

Traditional project scheduling literature usually analyses risk in a more general way, by considering activity costs and/or durations to be stochastic rather than deterministic. Although the majority of project scheduling research still uses the assumption that activity durations are deterministic (Herroelen and Leus, 2005), several authors have proposed methods for scheduling whilst taking stochasticity into account. An overview of these methods is presented in section 8.2.1.

A second challenge faced in the scheduling of offshore construction projects is the planning of expensive resources needed to perform the activities. The chartering of jack-up barges, transport ships, dumping vessels, crane vessels,etc. (European Wind Energy Association, 2011) is the key cost in the majority of offshore construction projects. Especially since many of these vessels have to be reserved up to two years beforehand, and significant mobilisation and demobilisation costs are incurred (Schönefeld, 2014). These issues can be related to resource constrained scheduling, as well as resource renting problems. These concepts are discussed in further detail in section 8.2.2.

The combination of weather sensitive activity durations and substantial variable costs associated with activities results in a significant financial risk. Hence, it makes sense to use a financial objective, rather than makespan-based targets. More specifically, the

main interest of the project manager in this case is the net present value of the complete project. The default formulation of this problem assumes that a fixed positive or negative cashflow is associated with the start of each activity, the goal being to speed up cash inflows and delay cash outflows. This basic formulation has been extended by considering time-dependent rather than static cashflows. Section 8.2.3 presents an overview of the most relevant publications which analyse this issue.

8.2.1 Dealing with Uncertainty

The techniques used for dealing with uncertainty that typifies projects in progress are frequently summarised using the term *dynamic scheduling* (Vanhoucke, 2012b). Within dynamic scheduling, three basic approaches can be distinguished: a reactive strategy, a proactive strategy or a mixed strategy (Fahmy et al., 2014). The literature review on stochastic scheduling by Herroelen and Leus (2005) provides even more detail and identifies six techniques which can be used to deal with stochasticity in a project scheduling environment.

Given that resources needed for the execution of offshore construction activities have to be reserved long beforehand, the focus of this research lies on the proactive approach. More specifically the construction of robust schedules, which are capable of dealing with the impact of adverse weather conditions in a satisfactory manner.

Practically, the creation of schedules for stochastic project scheduling problems is most frequently done by defining priority lists. Several different methods for creating these lists exist, each has advantages and disadvantages. Research by Stork (2000) has shown that linear restrictive policies can be expected to perform best in situations where resource restrictions (as defined by minimal forbidden sets) are limited. When this is not the case, and complex resource restrictions are present in the project, it is better to use activity-based priority rules. A key downside of this simple approach is that it is not possible to insert buffers at strategic moments in time when it may be ill-considered to proceed immediately with the next activity on the priority list.

Another approach which resolves this issue has been analysed by Ke and Liu (2005), who use a genetic algorithm in combination with a simulation procedure in order to determine the earliest start times of activities. This algorithm is tested on the three basic approaches for stochastic optimisation: the maximisation of the objective, the maximisation of the objective value attained with a certain probability α, or the maximisation of the probability of obtaining a certain objective function value. The latter two scenarios being of great significance when the aim of the project manager is protecting the downside risk associated with the project.

The method of improving schedule robustness by inserting time buffers at critical points

in the schedule is studied more explicitly by Herroelen and Leus (2004). The authors in this study analyse how the robustness of a schedule can be maximised for a given project deadline. The concept *float* is used to express the magnitude of the buffer inserted between activities. Three different types of float are identified for an activity: the *pairwise float* is equal to the amount of float between two activities which are connected to each other by a precedence relation. This float can be calculated by subtracting the planned finish time of the predecessor from the planned starting of the successor. The two other float metrics are derivations from this basic concept. The *free float* is the maximal delay of an activity, without affecting the start of it successors. Note that this metric takes all predecessors and successors into account. The *total float* is similar, but now the successors of the activity are assumed to start at their latest possible times. The origins of this technique can be traced back to earlier studies in machine and job-shop scheduling (Mehta and Uzsoy, 1998), where it was used to maximise a deterministic metric representing the robustness of a schedule.

Such deterministic proxies for robustness are hard to use in the situation studied in this chapter, where the stochasticity depends on a complex external process. An alternative for these deterministic metrics is presented by Huang and Ding (2011), who use a genetic algorithm combined with stochastic simulations to estimate the outcome of a specific strategy.

8.2.2 Dealing with Resources

The resource constrained project scheduling problem, commonly known as the $RCPSP$, is the most basic way of dealing with resources in a project environment (Hartmann and Briskorn, 2010). A variation on this approach is the resource renting problem, which does not only include renewable resources, but also takes into account the cost of renting these resources. Specifically, the costs associated with mobilisation and demobilisation of resources can inflate the total cost of a project to a project owner. This problem class was originally introduced by Nübel (2001), and more complex heuristic solution techniques have since been presented by Ballestín (2007, 2008). Extensions including more complex cost structures and the possibility of scheduling in overtime have also been presented in recent years (Kerkhove et al., 2015; Vandenheede et al., 2015). These more complex cost structures have a number of analogies with the chartering process used in offshore construction projects.

However, to the best of our knowledge literature on this problem has only studied the resource renting problem under the assumption of deterministic activity durations. Nevertheless, other authors have investigated methods of dealing with resources in a stochastic project setting. The CC/BM methodology presented by Goldratt (1997) is the most widely

known methodology to deal with similar situations.

A more specialised method for doing this is presented by Trietsch (2006), who uses *gates* to represent the moment in time when a resource has been made available for a certain activity. This methodology also defines two costs associated with these gates: a *holding cost*, which is incurred when an activity has to wait for resources, and a *shortage cost*, which is incurred when resources have already been made available, but the activity is still waiting for one or more predecessors to finish. A recent paper by Bendavid and Golany (2011) also uses this concept of gates in combination with a solution technique based on the cross entropy methodology in order to schedule a project with stochastic (exponential) activity durations.

8.2.3 Maximising Financial Returns

When cashflows can be determined for each individual activity, it is possible to use the project's net present value (NPV) as an objective measure of the outcome of the project, and by extension the comparative quality of different scheduling strategies. Traditionally, this is done within a deterministic setting where both the activity durations and the magnitude of the cashflows are assumed to be set with 100% accuracy beforehand. The problem described in this chapter however has a very pronounced stochastic component, which influences both the timing and the magnitude of the cashflows associated with the different activities. These variations are caused by stochasticity due to weather conditions, as well as the intrinsic variability of the activities. Although this specific scenario has not yet been investigated in existing literature, it can be linked to earlier problem formulations which have considered the implications of time-dependent cashflows. The remainder of this section focuses on authors which have investigated net present value optimisation whilst assuming some degree of variability of the cashflows associated with activities.

One such study is presented by Etgar et al. (1996), who investigate how projects consisting of activities with cashflows that increase or decrease over time can be scheduled to maximise net present value. An exact solution procedure for this problem was later presented by Vanhoucke et al. (2001). A variation on this problem where certain regions are defined wherein the cashflows remain constant is presented by Achuthan and Hardjawidjaja (2001).

The preceding authors all assumed that cashflows linked to activities could either increase, decrease or remain constant over time. However, for the problem studied in this chapter, it is possible that cashflows linked to activities display seasonal patterns, as variable costs are heavily influenced by weather conditions. This assumption was relaxed by Möhring et al. (2001), who defined a specific cost based on the starting time of an activity, but the evolution of this cost had the potential of fluctuating freely over time. In

a later paper Möhring and Schulz (2003) discovered that this problem formulation can be solved using fast minimum-cut operations.

The fluctuations in activity cashflows for the problem studied in this research are caused by stochastic variations in activity duration, depending on the time when the activity is executed. This is not completely identical to the problems studied in the aforementioned papers, which assumed that the cashflow itself is directly affected by the moment in time. Nevertheless, formulations which use time-dependent processing times have already been investigated in the context of machine scheduling (Cheng et al., 2004).

One key limitation of all preceding studies is that they consider time dependence within a deterministic rather than stochastic setting, something which some of these authors had already identified as a key limitation and possible avenue for future research (Etgar et al., 1996). Sobel et al. (2009) depart from this deterministic setting and maximise NPV for a project with stochastic activity durations using a reactive priority list approach. Such reactive strategies are, however, not adequate for the problem studied in this chapter. The reason for this being the significant lead times associated with both consumable and renewable resources required to execute activities. Hence, the focus of this research is to design a proactive approach in order to create robust solutions.

A recent paper by Huang and Ding (2011) presents such a proactive approach in a stochastic problem environment. Specifically, this is done through the combination of stochastic optimisation and a genetic optimisation algorithm. Nevertheless, the problem considered by Huang and Ding (2011) uses a cost objective rather than the net present value. Moreover, solely the duration of activities is assumed to be subjected by stochasticity - the actual cashflows are assumed to remain unaltered.

A maximisation of a project's NPV under the assumption of stochastic activity durations has been presented by Sobel et al. (2009), who used a reactive approach with priority lists to optimise the problem at hand. The use of these reactive strategies is however not advisable for the problem studied in this chapter because of the significant lead times needed to reserve the resources needed to execute the project. A more proactive approach to this problem is presented by Huang and Ding (2011), who also consider the duration of activities to be time-dependent stochastic variables, rather than solely stochastic. The solution approach used by Huang and Ding (2011) uses a genetic algorithm in combination with simulation techniques to find the optimal starting times for specific activities.

8.2.4 Concluding Remarks

The overview of the state of the art in related project management literature revealed that although attention has increased in recent years, a lot of potential remains in the area of stochastic project scheduling. Specifically, the maximisation of net present value in projects

where the cashflows are influenced by systematic stochasticity[1] has never been investigated. Building on works of earlier authors who have identified the major challenges in offshore construction projects (Graham, 1982; Rothkopf et al., 1974), this research proposes and tests the performance of novel techniques.

8.3 Problem Definition

This section introduces the components of the problem investigated in this research. First, a mathematical representation of the scheduling problem is presented in section 8.3.1. Next, the stochastic influences on this model are discussed in section 8.3.2. Finally, the problem is illustrated by means of a theoretical example in section 8.3.3.

8.3.1 The Scheduling Problem

The goal of this problem is to find the optimal 'gate times' (i.e. the time at which resources are made available for a specific activity) in order to maximise the net present value (NPV) of the complete project. These gate times are not only a theoretical representation of the schedule, but are actually required in order to guarantee the availability of the offshore construction vessels before the start of the project. The reasons for adopting the net present value as the objective are the following:

1. Offshore construction projects are highly capital intensive because of the high material cost and the need to charter large construction vessels. Hence, the objective used should take into account the magnitude and timing of these financial flows.
2. Due to the conditions in which these projects are executed, the timing and magnitude of cashflows can be volatile. Using the net present value allows for a more objective comparison of such volatile scenarios when compared to an optimisation of the total cost.
3. Linked to the preceding point is the fact that because of the inherent volatility, investors will require a larger return on investment for the project. Hence, the discount rate is larger than it would be for other projects. Again, this would mean a large discrepancy between the total costs incurred and the net present value of the project.
4. Because of the significant length (often multiple years) of offshore construction projects, it is required to discount both the cash outflows and the cash inflows to get an accurate image of the profitability of the project.

For projects with the aforementioned properties, there is a decided preference for the use of net present value as the optimisation objective (Herroelen et al., 1997). Other

[1] i.e. stochasticity originating from systemic weather influences.

objectives such as the minimisation of the project lead times are likely to result in reduced project profits.

Figure 8.1 visualises the various elements of the project which are relevant to the optimisation. The project can be seen as a collection of interrelated activities and resources. The activities are interconnected by precedence relationships and are also linked to the resources they require to be executed. Within the context of offshore construction these resources are the construction vessels.

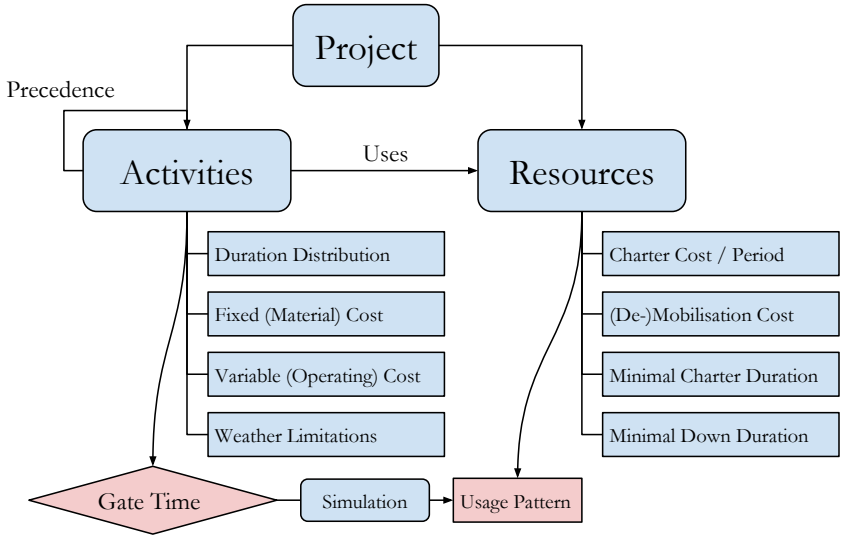


Figure 8.1: Schematic representation of the project model

An activity is defined by its duration (d_i), its fixed and variable cost (C_i^F and C_i^V) and the weather limitations under which it can operate. The fixed cost associated with an activity represents the cost of the materials which are used during the activity. For offshore wind projects, this constitutes (but is not limited to) the turbine itself, as well as its foundation and scour protection - naturally these are allocated to the respective activities. The variable cost is the operating cost incurred during the execution of the activity, this includes the fuel cost of operating the vessels as well as the costs for personnel operating at sea. However, this does not include the chartering cost of the vessel itself which has to be paid even when the vessel is temporarily not used but not decommissioned, this cost is linked to the specific resources. The activity's weather limitations are expressed in two dimensions: the maximal wave state and the maximal wind state at which the activity can be executed. Exceeding these thresholds will cause the activity to be temporarily halted until weather conditions improve. Note that it is of paramount importance to distinguish between these two dimensions for offshore operations since different types of activities will

have a different degree of wind- and wave-sensitivity. For ease of notation this is simplified in the deterministic model presented below, using the binary ω_{it} parameter to indicate if the weather threshold for activity i is not exceeded in time period t.

Separate costs are associated with the resources used by the activities (i.e. the installation vessels): the charter cost per period (ρ_r^V), the mobilisation and demobilisation cost (ρ_r^C and ρ_r^D), as well as the minimal duration of both the chartering ($\delta_{up}^{\min}$) and the downtime ($\delta_{down}^{\min}$). The basic cost of a vessel is the daily amount paid as rent, however at the start of the renting period the operator also has to pay to deploy the vessel at the desired location (ρ_r^C). Similarly, costs are also associated with the decommissioning of these vessels (ρ_r^D). These costs vary depending on the vessel type but are often equal to 4 to 8 times the associated dayrate of the vessel (Schönefeld, 2014). Related to these mobilisation and demobilisation charges are the minimal durations for both chartering and downtime, which are required by the company which leases the vessels to ensure better work continuity (Vanhoucke, 2006). The minimal charter duration guarantees that the vessel will be chartered for a sufficiently long period. Similarly it is undesirable to decommission and recommission vessels with only very short downtimes, simply because it is unlikely that the vessel's owner can find a client for this small time-frame.

Because of this (de)mobilisation decision, the chartered resources are not simply viewed as a variable cost that is directly linked to an activity. Once a schedule is constructed, the demand for a certain vessel can be determined and the periods when the vessel needs to be chartered can be fixed. An important note here is that regardless of the fact that the acquisition of these resources has a very long lead time, the contractual agreements generally allow for a scheduling contingency, meaning that resources will not be deallocated until the work for which they have been chartered has been completed.

The schedule itself is defined by setting *gate times* for all activities, these times indicate when the resources to execute an activity will be made available. Hence, the time at which an activity can start is the maximum of this gate time and the latest finish time amongst its predecessors. This principle of gate times is similar to the methodology used by Trietsch (2006) and Bendavid and Golany (2011) (see section 8.2.2). Practically, this means that the gate time is directly linked to the fixed cost associated with the activity. Hence, this fixed cost will always be incurred at precisely the gate time of the activity. The variable activity costs, as well as the costs associated with chartering vessels will be incurred when the activity is actually executed.

The simplest representation of the problem studied in this chapter is a deterministic scheduling problem as described by equations 8.1 until 8.10. The mathematical model below assumes that both the duration of the activities and the weather pattern is completely deterministic, an assumption which will be relaxed in section 8.3.2. The notation used in

this scheduling optimisation model is summarised in table 8.1.

$$\max NPV = \sum_{i=1}^{NI}\left[\frac{-C_i^F}{(1+ROI)^{g_i}} - \frac{C_i^V\cdot(f_i-s_i)}{(1+ROI)^{f_i}}\right] + \sum_{r=1}^{NR}\sum_{t=1}^{NT}\left[\frac{-\rho_r^V\cdot u_t^r}{(1+ROI)^t} - \frac{\rho_r^C\cdot v_t^r}{(1+ROI)^t} - \frac{\rho_r^D\cdot w_t^r}{(1+ROI)^t}\right] + \frac{\pi}{(1+ROI)^{f_{NI}}} \quad (8.1)$$

subject to:

$$s_i = \max\left[g_i, \max_{j\in P_i}(f_j)\right] \quad \forall i\in I \quad (8.2)$$

$$\sum_{t=s_i}^{f_i-1}\omega_{it} = d_i \quad \forall i\in I \quad (8.3)$$

$$u_{rt} = \sum_{i|s_i\le t<f_i}\omega_{it} \quad \forall r\in R \quad \forall t\in T \quad (8.4)$$

$$\varrho_{rt} \ge u_{rt} \quad \forall r\in R \quad \forall t\in T \quad (8.5)$$

$$\varrho_{r1} = v_{r1} \quad \forall r\in R \quad (8.6)$$

$$\varrho_{rt} = \varrho_{r(t-1)} + v_{rt} - w_{rt} \quad \forall r\in R \quad \forall t = 2,...,NT \quad (8.7)$$

$$w_{rt} \le M\cdot\left[1-\min\left(1, \sum_{t'=\max(1,t-\delta_{up}^{\min})}^{t} v_{rt'}\right)\right] \quad \forall r\in R \quad \forall t\in T \quad (8.8)$$

$$v_{rt} \le M\cdot\left[1-\min\left(1, \sum_{t'=\max(1,t-\delta_{down}^{\min})}^{t} w_{rt'}\right)\right] \quad \forall r\in R \quad \forall t\in T \quad (8.9)$$

$$g_i, s_i, f_i \in \mathbb{N}_0 \quad \varrho_{rt}, u_{rt}, v_{rt}, w_{rt} \in \mathbb{N} \quad (8.10)$$

The main decision variable in the optimisation model is the gate time g_i, which represents the time period at which the resources for an activity have been made available. The NPV-objective is calculated by equation 8.1, which is a function of the various cash-

Indices	
i	Activity index: $i \in I = \{1, ..., NI\}$
r	Resource index: $r \in R = \{1, ..., NR\}$
t	Time interval index: $t \in T = \{1, ..., NT\}$
Decision variables	
g_i	Gate time of activity i
Assisting variables	
s_i	Starting time of activity i
f_i	Finish time of activity i
u_{rt}	Resource usage of r at time t
v_{rt}	Commissioned amount of r at time t
w_{rt}	Decommissioned amount of r at time t
ϱ_{rt}	Amount of resources r rented in period t
Stochastic parameters	
ω_{it}	Binary = 1 if suitable weather for act i at t
d_i	Duration of activity i
Fixed parameters	
ROI	Discount rate per period
C_i^F	Fixed cost of activity i
C_i^V	Variable cost of activity i
ρ_r^V	Variable chartering cost for resource r
ρ_r^C	Commissioning cost for resource r
ρ_r^D	Decommissioning cost for resource r
π	The cashflow associated with the completed project
μ_{ir}	Binary = 1 if act i requires resource r
P_i	Set of predecessors of activity i
$\delta_{up}^{\min}$	Minimal renting period required for resources.
$\delta_{down}^{\min}$	Minimal downtime period required for resources.
M	A large number.

Table 8.1: Overview of notation.

flows associated with the project. The fixed cost (C_i^F) represents the building materials required to perform an activity. Since the gate time g_i represents the moment in time when these resources have been made available this is the discount factor for the fixed cost. As explained earlier, the variable costs (C_i^V) are incurred in all time periods as soon as the activity has actually started. These costs are assumed to be payable upon completion of the activity. The second part of equation 8.1 represents the costs associated with the resources (i.e. the chartering of the vessels). This includes the variable operating costs ρ_r^V incurred when the resource is actually used, as well as the commissioning (ρ_r^C) and decommissioning (ρ_r^D) costs associated with the vessel. Finally, the positive cashflow associated with completing the project is also added to the NPV.

The earliest possible starting time of an activity is either the gate time g_i, or the time period after the finish of its latest predecessor, whichever is greater (equation 8.2). The activity finishes (f_i) as soon as the number of time periods with acceptable weather conditions since the start of the activity is equal to the duration of the activity d_i (equation 8.3). In order to calculate the variable operating cost of the resources the variable u_{rt} is introduced. This variable is equal to the resource use for resource r during period t, and is assigned by considering all active activities during time period t as is shown in equation

8.4. In order to calculate the commissioning and decommissioning costs correctly, the variable ϱ_{rt} is introduced. This variable is equal to the available amounts of resource type r during period t. Naturally, the available amount of resources must always exceed the amount of resources used (equation 8.5). The evolution of the ϱ_{rt} variable is tracked by the u_{rt} and w_{rt} which track the commissioned and decommissioned resources during each time period respectively. The correct value for these variables is enforced by equations 8.6 and 8.7. Finally, equations 8.8 and 8.9 guarantee the minimal number of periods for the (de-)commissioning as explained above.

This model is problem specific in that it only allows a single resource to be associated with an activity. These resources correspond to the specific installation vessels associated with offshore construction projects. As a side note, this type of project generally only has one resource of every type available because of the scarcity of these vessels on the market.

In the next section stochasticity is introduced into the scheduling problem, prior to presenting a small theoretical example in section 8.3.3 in order to familiarise the reader with the concepts which have just been introduced.

8.3.2 Stochastic Influences

The optimisation model presented in section 8.3.1 assumed that all model parameters are deterministic. In reality however, there are significant sources of uncertainty in offshore construction projects. Traditional stochastic project scheduling (see section 8.2.1) defines probability distributions for each activity to express this risk.

For offshore construction projects such a model fails to capture the systematic uncertainty introduced by seasonal weather conditions. Hence, this research considers both uncertainty inherent to specific activities and the uncertainty introduced due to weather conditions.

The activity-dependent uncertainty is modelled by defining a triangular distribution for each of the activity durations, as is conventional in project scheduling literature. For the weather uncertainty, a novel simulation model is proposed based on recent advances in weather simulation techniques. The details of this weather simulation model are explained in detail in section 8.4.2.1.

The implications for the scheduling optimisation model described in section 8.3.1 are that both the activity durations (d_i) and the weather conditions (ω_{it}) are now stochastic variables. Hence, the objective is not to optimise the gate times NPV for a single value of these parameters, but rather to optimise the expected value of the net present value ($E[NPV]$).

8.3.3 Example Problem

This section presents a highly simplified example in order to illustrate the basic workings of the scheduling model. Assume a simple project consisting of two activities: 1 and 2, which are scheduled in series. The basic properties of these activities can be summarised as follows:

	1	2
Duration Distribution	$U(9,11)$	$U(8,12)$
Fixed Cost (C_i^F)	€ 100	€ 175
Variable Cost (C_i^V)	€ 5	€ 10

Table 8.2: Example project information

Both these activities depend on the same installation vessel for their execution. This vessel has a charter cost (ρ_r^V) of € 10 per period, and a commissioning (ρ_r^C) and decommissioning (ρ_r^D) cost of € 15. For this simple example it is assumed that there are no minimal periods required for chartering or vessel decommissioning.

The actual duration of both these activities also depends on the weather conditions. For this theoretical example, a simplified weather model which consists of only two states - good weather and bad weather - is used. Both activities are only capable of operating in the good weather conditions and are interrupted if weather conditions are bad. These weather states can be simulated using Markov chains defined by two transition matrices: one for the 'summer' season, and one for the 'winter' season. Furthermore it is assumed that each of these seasons lasts 15 time periods.

Summer transition matrix		
	good	bad
good	0.9	0.1
bad	0.7	0.3

Winter transition matrix		
	good	bad
good	0.6	0.4
bad	0.5	0.5

Table 8.3: Example weather model

The actual value of the project itself (i.e. the cash inflow received upon project completion π) is equal to € 1,250, and the discount rate per period (ROI) is equal to 1%.

Based on this information two basic strategies are immediately apparent. The first option is to start all activities as soon as possible. From the point of view of the decision variables this means that the first gate is set at time 1 (i.e. activity 1 can start immediately)

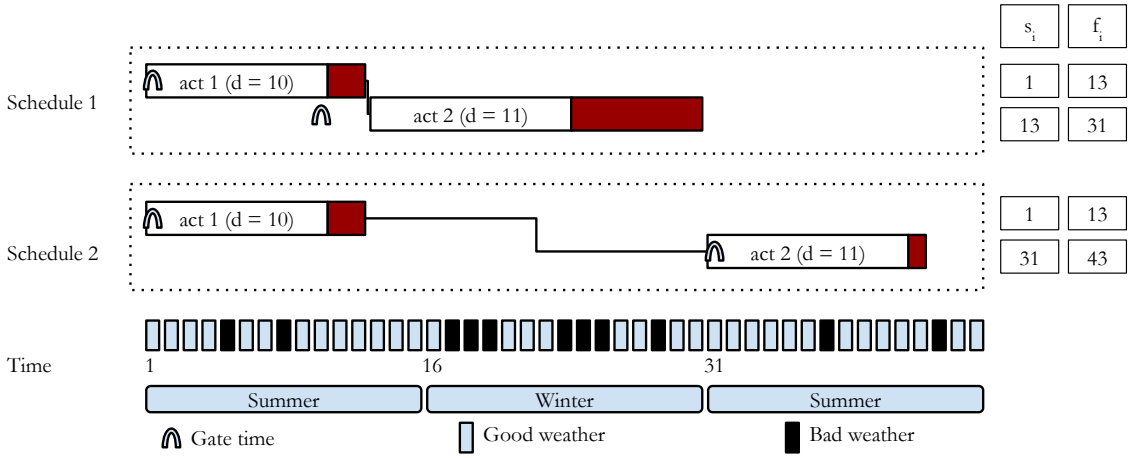


Figure 8.2: Comparison of scheduling approaches for the sample problem.

and the second gate is set at the earliest possible starting time of activity 2. Assuming that activity 1 has the shortest possible duration of 9 time periods and the weather is continually favourable. This solution can be noted as: $(g_1, g_2) = (1, 10)$.

This strategy is likely to result in considerable delays because the second activity is executed at least partly in the winter period. Even in the best case scenario where there is not a single period of bad weather during summer and all activity durations are minimal. Hence, an alternative strategy is to delay the start of the second activity until the end of the winter period (i.e. $g_2 = 31$). This is likely to reduce the net working time during the project, but does delay the project's completion and results in a commissioning and decommissioning charge to be incurred, inflating the non-discounted cost of the project by €30.

Analysing the impact of these strategies requires simulating the execution of the project. A visual example of a single simulation run of the example problem is shown in figure 8.2. The simulation procedure has generated a specific duration for the activities from the uniform distributions in the table above: activity 1 has a duration of 10, and activity 2 has a duration of 11. Moreover, the simulation procedure has also simulated a weather pattern which is represented by the light and dark coloured blocks at the bottom of figure 8.2. A dark coloured block represents a period with bad weather, during which activity progress is interrupted. These interruptions extend the duration of the activities, as shown by the additional duration added at the end of activities for both schedules. An important note here is that making up for delays during the activity progress can only be done when weather conditions are good. Hence, bad weather conditions after the planned finish can further extend the duration of the activity as well - as can be seen from the

duration of the second activity when using the first schedule.

The simulation shown in figure 8.2 indicates that weather conditions cause significant inflation of the duration of the activities. The effective activity durations are 12 and 18 for the first and second activity respectively when the first schedule is used, and both activities have a duration of 12 when the second scheduling procedure is used.

The scheduling strategies are represented visually by the 'gates' that indicate the earliest starting times of the activities. When comparing the performance of both schedules, it can be seen that the second schedule has set the gate time for the second activity much further, at a time when good weather conditions were expected. This has resulted in a much shorter duration for the second activity, at the cost of inflating the duration of the complete project. To judge which of these schedules performed best during this simulation, the net present value of the costs and return of the project have to be analysed in detail, as is shown in tables 8.4 and 8.5 which show the different components of the net present value as calculated using equation 8.1.

	Cashflow	Timing	NPV (€)
Activity 1 fixed cost	-100	0	-100.0
Activity 1 variable cost	-60	12	-53.2
Activity 2 fixed cost	-175	10	-158.4
Activity 2 variable cost	-180	30	-133.5
Chartering cost	-300	15	-258.4
Mobilisation	-15	0	-15.0
Demobilisation	-15	30	-11.1
Terminal value	1250	30	927.4
		TOTAL	197.7

Table 8.4: NPV calculation for first scheduling strategy

	Cashflow	Timing	NPV (€)
Activity 1 fixed cost	-100	0	-100.0
Activity 1 variable cost	-60	12	-53.2
Activity 2 fixed cost	-175	29	-131.1
Activity 2 variable cost	-120	41	-79.8
Chartering cost	-240	15	-206.7
Mobilisation 1	-15	0	-15.0
Mobilisation 2	-15	29	-11.2
Demobilisation 1	-15	12	-13.3
Demobilisation 2	-15	41	-10.0
Terminal value	1250	41	831.3
		TOTAL	210.8

Table 8.5: NPV calculation for second scheduling strategy

When comparing the results in detail it can be seen that the first schedule strategy

results in a considerably higher variable cost for the second activity, as well as a greater chartering cost, since the total duration during which the vessels are used is longer (30 time periods). The second schedule on the other hand is able to present much lower costs, but decreases the value of the project by extending its duration, resulting in a lower net present value of the terminal value of the project.

Naturally, it must not be assumed that the results from a single simulation run are representative and more simulation runs are needed in order to get accurate estimates of the relative performance of the different scheduling strategies for this project.

8.4 Solution Approach

This section presents the proposed solution approach for the problem defined in section 8.3. Finding a solution for this problem is a complex problem because of two aspects. First and foremost the number of possible schedules is very large, as will be discussed in section 8.4.1 where the solution space is discussed. Secondly, the stochastic nature of the problem requires simulations in order to correctly evaluate the quality of the schedule (see section 8.4.2).

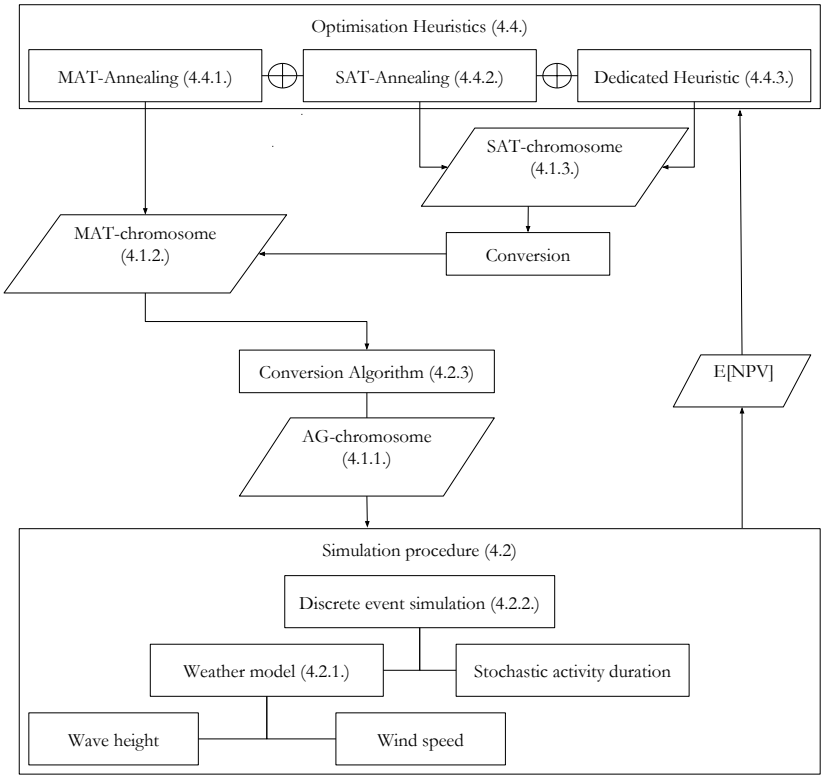


Figure 8.3: Schematic overview of the solution approach

Figure 8.3 gives a schematic overview of the proposed solution approach, including references to the sections in which the various elements are discussed. The two main building blocks of the solution procedure are the optimisation heuristics (section 8.4.4) and the simulation procedures (section 8.4.2). The optimisation heuristics create schedules for the construction project, which are expressed as different types of chromosomes depending on the optimisation heuristic used. These chromosomes are then passed to the simulation procedure which calculates the expected net present value ($E[NPV]$). This information is then returned to the optimisation heuristic.

Three different optimisation heuristics are proposed in this research: two meta-heuristic approaches (section 8.4.4.1 and 8.4.4.2) and a dedicated heuristic (section 8.4.4.3). The chromosome encodings used by these algorithms have been created to reduce the size of the solution space, and by extension the performance of the solution heuristic, as discussed at length in section 8.4.1. The simulation model uses discrete event based simulation (section 8.4.2.2), and includes both stochastic activity durations and realistically generated weather patterns (section 7.3). The weather model itself generates realistically correlated wind and wave states for every time period.

8.4.1 Solution Representation

As was stated in section 8.3.1, the operational schedule is defined by a set of gate times (g_i) for each activity i. This variable can have any positive integer value within the time interval taken into consideration $(1, ..., NT)$ and represents the moment in time when the materials needed for the execution of the activity have been made available. The model assumes that the payment for these materials has to be made at the moment they are delivered, meaning that the net present value of these outgoing cashflows is based on the gate times which have been set for these materials.

Based on this principle, several ways of constructing a solution representation (chromosome) can be proposed. When constructing such a solution representation it is important to balance the coverage of the total solution space with the total possible number of solutions. An encoding which is too detailed will increase the computation time needed to adequately search the solution space beyond practical value. A solution which is too restrictive on the other hand may cause severely suboptimal solutions due to the limited set of solutions which is considered. Sections 8.4.1.1, 8.4.1.2 and 8.4.1.3 present three different solution encodings at different points in this spectrum. These encodings are then used by the heuristic solution procedures presented in section 8.4.4.

8.4.1.1 Activity Gate Chromosome

The simplest way of representing a solution is simply to use the gate times for the individual activities as alleles in the chromosome encoding. Hence, each allele (g_i) is an integer gate time, indicating the specific time period in which the resources for activity i are made available:

$$[g_1, g_2, ..., g_{NI}] \qquad g_i \in \{1, ..., NT\} \tag{8.11}$$

The first schedule of the example presented in section 8.3.3 can be represented by the following activity gate chromosome: $[1, 10]$. This implies that the resources for the first activity are available immediately ($g_1 = 1$), and the resources for the second activity are available at the start of the 10-th time period ($g_2 = 10$). Practically this implies that the first activity will be able to start immediately, and the second activity will start at the 10-th time period, or once the first activity has been completed, whichever is later: $s_2 = \max(g_2, f_1)$.

The disadvantage of this solution representation is the large size of the solution space, even for problems of limited size. The exact number of possible solutions can be calculated as NT^{NI}, where NI is the total number of activities and NT is the total number of time periods considered. Depending on the precedence relations in the project and the manner

in which the durations of the activities are specified, a subset of these combinations can be eliminated from the pool of rational solutions. This is done by preventing gate times from taking values smaller than the earliest possible start of the respective activity.

8.4.1.2 Monthly Activity Type Chromosome

Besides eliminating irrational solutions, the size of the solution space can also be reduced by using other chromosome definitions. One way of doing this is by taking advantage of the repetitive nature of offshore projects: many types of activities, and even sequences of activities are repeated within the same project. Hence, rather than specifying when specific activities are allowed to start, it is possible to define during which time periods activity types are allowed to be executed.

The size of the solution space can be reduced even further by aggregating the time periods into longer intervals. Given that the weather model uses months as climatological units, and the motivation for allowing or not allowing certain activity types is likely motivated by expected weather conditions, it makes sense to aggregate the time periods per month.

The term 'monthly activity type' or MAT chromosome will be used to describe this type of chromosome. The chromosome itself consists of a set of binary variables $x_{j\tau}$, which signify whether (1) or not (0) activity type j ($j \in [1, 2, ..., NJ]$) is allowed to be executed in month τ ($\tau \in [1, 2, ..., N\tau]$). The chromosome can be visualised as a matrix where the rows represent the activity numbers and the columns represent the time periods (months):

$$\begin{bmatrix} x_{11} & ... & x_{1,N\tau} \\ ... & x_{jm} & ... \\ x_{NJ,1} & ... & x_{NJ,N\tau} \end{bmatrix} \tag{8.12}$$

Again this chromosome can be illustrated using the simple example which has been defined in section 8.3.3. For this example the 'summer' and 'winter' seasons can be seen as the relevant time periods indexed $\tau = 1, 2, 3$. The two activities in the example project are assumed to be of different types: $j = 1, 2$. This results in a total of six binary variables which can be placed in a matrix structure to represent the MAT chromosome encoding:

$$\begin{bmatrix} x_{11} = 1 & x_{12} = 0 & x_{13} = 0 \\ x_{21} = 0 & x_{22} = 0 & x_{23} = 1 \end{bmatrix} \tag{8.13}$$

It can be noted that the the values of x_{12} and x_{13} could also be set to 1 without impacting the way in which the project is executed. The underlying execution strategies for both these chromosomes are indeed different. However these differences held no relevance with regard to the outcome of the project as expressed by the MAT chromosome. Hence,

the fact that these variables can take different values should simply be seen as an ex aequo between two different solutions[2].

The substantial reduction of the solution space when using the MAT chromosome encoding, rather than the gate times encoding is already apparent from this small example. Using the MAT encoding, the total number of combinations can be calculated as $2^{NJ \cdot N\tau}$, for the example from section 8.3.3 this means that there are a total of $2^{2 \cdot 3} = 64$ possible combinations. Whereas, when using the original chromosome the total number of combinations was equal to $NT^{NI} = 45^2 = 2,025$.

In order to determine the expected net present value when using a schedule defined by a MAT-chromosome the activity specific gate times have to be determined (see section 8.3.1). Effectively, the higher-level MAT-chromosome has to be translated into an AG-chromosome (see section 8.4.1.1) encoding. This is done using an iterative simulation procedure which is explained in detail in section 8.4.2.3.

8.4.1.3 Seasonal Activity Type Chromosome

A further reduction of the solution space can be obtained by reducing the project schedule to a combination of two key decisions. The first decision being when to start the project, and the second decision being how to deal with the weather sensitivity for each activity type.

Similarly to the MAT-chromosome encoding, this chromosome uses the climatological periods as they are defined in the weather simulation model (months for the model used in this research). However, whereas the MAT-chromosome defined a variable for each occurrence of these time periods (i.e. January of 2015, January of 2016, ...), the encoding scheme only defines a single variable for each climatological period (i.e. January, February, ...). In effect, the seasonality of the weather pattern is used to decrease the complexity of the scheduling decision. Hence, this type of chromosome encoding is dubbed a seasonal activity type (SAT) chromosome.

The SAT-chromosome is a list of integer variables in the range $[0, NP - 1]$ where NP is equal to the number of climatological periods used in the weather model, in this case 12 months. This list includes one variable s which indicates the number of months the start of the project is delayed. This allele can be used to align periodic weather sensitivity of the project with periods of more favourable weather. A value of $s = 0$ indicates that the project starts immediately, whereas a value of $s = 3$ indicates that the start of the project is postponed with three months (i.e. assuming the earliest start would be in January, a value $s = 3$ will result in the project starting in April).

[2]One could also argue that any MAT chromosome corresponds to a set of activity gate chromosomes.

The second part of the SAT-chromosome consists of a variable (d_j) for each of the activity types (j). This variable represents the number of months during which activities of type j will not be performed to avoid weather related delays. Naturally, the months during which the execution is halted are the months during which the worst average weather conditions are expected. Which time periods are the worst on average is determined using the steady state probability matrices of the weather model (see section 8.4.2.1). The relative weather quality is simply defined as the average of the wind- and wavestates during the time period.

This results in the following representation of the project schedule:

$$[s, d_1, d_2, ..., d_j, ..., d_{NJ}], \qquad s, d_j \in [0, ..., NP - 1] \tag{8.14}$$

By using this encoding, the number of possible schedules in the solution space is further reduced to $(1 + NJ)^{NP}$. Again the use of this chromosome can be illustrated using the example from section 8.3.3. Again assuming that the two activities in the example are of different types, the following SAT encoding can be used to represent the second schedule of the example:

$$[s = 0, d_1 = 0, d_2 = 1] \tag{8.15}$$

Equation 8.15 shows that the schedule starts immediately during the first period ($s = 0$). Also the first activity is allowed to be executed during every time period, whereas the second activity cannot be executed during the period with the worst weather conditions. Which period has the best or worst weather or average is determined by using the steady state probabilities, as described in section 7.3.1. For the simple example the steady state probability for good weather during the summer season is 87.8%, whereas the probability for good weather is equal to only 55.6% during the winter season. Hence, it is clear that the winter season has the worst weather conditions and the second activity type should be suspended during the winter rather than the summer season.

In order to obtain the specific gate times associated with the schedule represented by the SAT-chromosome, the SAT-encoding is simply translated into a MAT-chromosome, after which the procedures used for the latter can be used to determine the activity specific gate times.

8.4.1.4 Implications of Aggregated Solution Encodings

The first key implication of the aggregated encoding is the significant reduction in the size of the solution space. This reduction is illustrated in figure 8.4. This example assumes a project with a five-year timespan using four-hour time periods for the simulation procedure.

It is clear that the chromosome used has an impact of several orders of magnitude regarding the size of the solution space. Moreover, this impact increases as the size of the project increases.

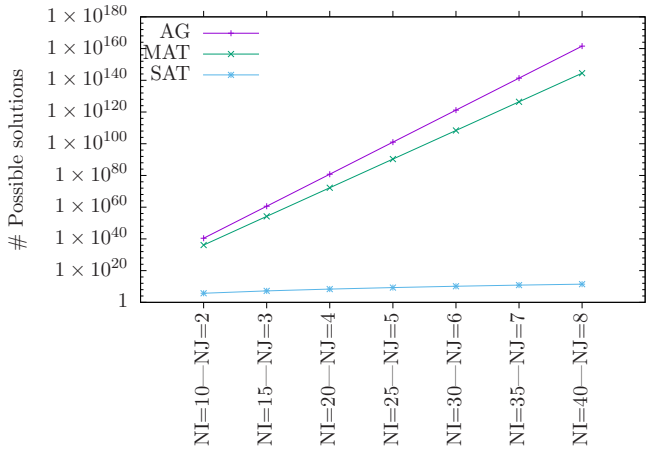


Figure 8.4: Evolution of the size of the solution space when using different chromosomes.

A second side note which has to be made regarding these aggregated encodings is that these are best applied to projects with primarily serial networks, which ideally also have sufficiently long activity durations. This type of network is however very typical for large offshore projects due to the limited availability of construction vessels. While these encodings can be applied to serial networks, a network with a large amount of merging paths may cause excessive buffering when the chromosome is translated into an activity gate encoding.

8.4.2 Simulation Procedure

8.4.2.1 Modelling the Weather

The weather is simulated using the simulation model described in chapter 7.

8.4.2.2 Simulation of *AG*-chromosome Solutions

The goal of the simulation procedure is to obtain a value for the start and finish times for every activity based on a *AG*-chromosome schedule representation (see section 8.4.1.1), hereby allowing the net present value (NPV) of the project to be calculated (see section 8.3.1). The simulation procedure itself takes into account two distinct sources of uncertainty: uncertainty with respect to the duration of activities and uncertainty with respect to the influence of weather conditions.

Algorithm 7 Simulation Procedure

```
1: procedure RunSimulation(AG-chromosome, weatherPattern)
2:     agenda ← ∅
3:     ∀i : agenda ← agenda ∪ {g_i}
4:     ∀i : released_i ← False
5:     ∀i : ρ_i ← # predecessors_i
6:     while agenda ≠ ∅ do
7:         e ← agenda.pop()
8:         ι ← e.index()
9:         if e.type() = gate then
10:            released_ι ← True
11:            if ρ_ι = 0 then
12:                s_ι ← e.time()
13:                agenda ← agenda ∪ {f_ι}
14:        else if e.type() = finish then
15:            for s in successors_ι do
16:                ρ_s ← ρ_s − 1
17:                if ρ_s = 0 and released_s = True then
18:                    s_ι ← e.time()
19:                    agenda ← agenda ∪ {f_s}
20:    return {s_i, f_i} ∀i
```

Algorithm 7 is a pseudocode representation of the discrete-event based simulation procedure used to evaluate the performance of a specific schedule. The input for the simulation procedure is an *AG*-chromosome. Such a chromosome consists of a gate time g_i for each of the activities in the project. The outcome of the simulation procedure is a starting (s_i) and finishing time (f_i) for each of the activities. This information is sufficient to calculate the net present value for this specific scenario. Naturally in order to obtain an estimate of the expected net present value ($E[NPV]$) multiple simulations will have to be performed.

The simulation procedure starts by defining an empty event agenda, and then adds the gate times (g_i) for all the activities as events to the agenda (line 3). For each activity a boolean variable $released_i$ is defined which is used to indicate if the gate time of activity i has already occurred. Another variable ρ_i is introduced and initialised to track the number of unfinished predecessors remaining for every activity.

Once these variables have been declared, the procedure starts looping over the events in the agenda (line 6). At each iteration the next event e is removed from the event agenda. The variable ι is used to represent the index of the activity to which the event is linked. Two types of events are considered in the simulation procedure: a gate time of an activity and the finishing time of an activity.

If the event e is a gate time of an activity (line 9), the $released_\iota$ variable is set to $True$. This indicates that the activity can start as soon as all its predecessors have been completed. Next, the procedure tests if there are no remaining predecessors (line 11). If this is the case, the finishing time of the activity can be calculated, taking into account a randomly drawn activity duration, as well as the simulated weather conditions

in combination with the operational restrictions of the specific activity. This finishing time is then added to the agenda as a new event since it may indicate the time at which the number of predecessors of an activity has to be decreased.

If the event e is a finishing time of an activity (line 14), the algorithm iterates over all the successors of activity ι, decrementing the number of remaining predecessors (ρ_s) for each of these successors. For each successor, the procedure also tests if the total number of predecessors is now zero. If this is the case and the activity has already been released the starting and finishing times of the activity can be determined. The finishing time of the activity is then also added as a new event to the agenda.

8.4.2.3 Chromosome Re-encoding Simulation

As was mentioned in section 8.4.1.2, the higher-level chromosome encodings (MAT and SAT) do not directly specify gate times for all individual activities. Nevertheless, simulating the execution of the project using the procedure described in section 8.4.2.2 requires explicit gate times to be defined for each activity. Naturally, due to the stochastic nature of the project it is infeasible to specify gate times which are guaranteed to respect the conditions set by the higher order MAT or SAT chromosomes. Because of this, an iterative satisficing procedure to set the gate times is proposed. The outline of this procedure is visualised by figure 8.5.

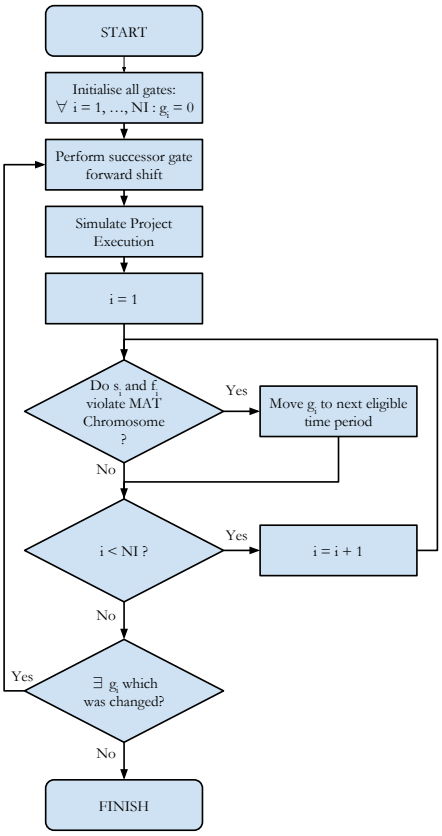


Figure 8.5: Iterative Procedure to Set Gate Times

The procedure starts by setting all gate times (g_i) at the start of the project ($g_i = 1$). Next, the precedence relations are used to shift a fraction of these gate times forward in time based on the release times and shortest possible durations of their predecessors (assuming no weather delays). This set of gate times is then simulated using the procedure which was described in detail in section 8.4.2.2. The outcome of this simulation is a set of starting (s_i) and finishing times (f_i), which are compared with the conditions imposed by the MAT or SAT chromosome. In case the execution of an activity falls within a period which does not allow execution of its activity type, the gate time of the activity is shifted forwards.

Once this has been tested for all the activities, the algorithm verifies if any gates were shifted since the last simulation run. If this is the case, the forward shift procedure for all successors is called again, and a new simulation is performed. When no more gate times have been adjusted after performing a simulation, the restrictions imposed by the higher-order chromosome are assumed to be satisfied.

8.4.3 Comparing Solutions

Due to the stochastic nature of the scheduling problem, it is important to verify that differences in performance between two schedules are statistically significant. This is especially important when comparing alternatives in a (meta-)heuristic procedure. However, constructing independent confidence intervals for both solutions is computationally expensive. When using unrelated samples the variance of the difference between two estimators can be calculated as:

$$\text{var}\left[N\hat{P}V(s_1) - N\hat{P}V(s_2)\right] = \text{var}\left[N\hat{P}V(s_1)\right] + \text{var}\left[N\hat{P}V(s_2)\right] \tag{8.16}$$

Where $N\hat{P}V$ is an estimator of the NPV for a specific schedule s. Naturally, the statistical significance of the difference between two solution can be improved by increasing the number of simulations. However, this increased statistical significance comes at a considerable computational expense.

Constructing correlated samples by using common random numbers is an alternative technique which can also increase statistical significance at no increased computational expense. By comparing the performance difference between two schedules using the same random estimates for the weather conditions and the activity durations, a strong correlation of the samples can be obtained. Whereas the covariance between the two estimators could be assumed to be zero in equation 8.16, given the independent samples used, the covariance will now become strictly positive. This means the variance of the difference between the two estimates now becomes:

$$\begin{aligned}\text{var}\left[N\hat{P}V(s_1) - N\hat{P}V(s_2)\right] = \\ \text{var}\left[N\hat{P}V(s_1)\right] + \text{var}\left[N\hat{P}V(s_2)\right] \\ - 2 \cdot \text{cov}\left[N\hat{P}V(s_1), N\hat{P}V(s_2)\right]\end{aligned} \tag{8.17}$$

Given that the covariance will always be strictly positive, this will always be a reduction of the variance of the difference between the two estimators. Hence, this technique enables a more efficient one-on-one comparison of different schedules, increasing the number of schedules which can be tested in a given time-frame. The latter being of paramount importance to the performance of local-search based meta-heuristic procedures since it allows a larger number of solutions to be investigated in a given time-frame.

8.4.4 Optimisation Heuristics

Creating optimal schedules is complicated by both the size of the solution space (see section 8.4.1), and the need for simulation in order to estimate the NPV associated with a specific schedule. Especially since these simulations are relatively computationally expensive, enumerative procedures which iterate over all possible schedules are eliminated as viable solution methods. Moreover, exact solution techniques are also ill-suited due to the stochastic nature of the problem.

Hence, the scheduling optimisation uses heuristic techniques to make an efficient scan of the most promising areas of the solution space. In the remainder of this chapter three different heuristics to solve this problem are proposed and tested. Two of these are local-search-based meta-heuristics, the first of which uses the MAT-chromosome encoding, and the second of which uses the higher-level SAT-chromosome encoding. Note that the choice for local search rather than population based meta-heuristics was made because of the efficiency gains which are possible when comparing solutions one-on-one using common random numbers (see section 8.4.3). These efficiency gains can no longer be obtained when selecting the best solutions out of a large pool of candidates.

Rather than using general purpose techniques on a specific solution encoding, the third solution technique is a dedicated heuristic which tests the most 'rational' choices in the solution spectrum. The subsequent sections provide more information on the specifics of the meta-heuristics which are used, after which the performance of these heuristics is tested in sections 8.5 and 8.7.

8.4.4.1 MAT-chromsome Annealing

The first meta-heuristic is an implementation of the simulated annealing meta-heuristic (Talbi, 2009), which uses the MAT-chromosome encoding presented in section 8.4.1.2. A basic outline of the annealing procedure is presented by algorithm 8.

The neighbourhood moves are performed by flipping a random binary value in the MAT-chromosome from 1 to 0 or vice versa. Next, the iterative procedure (see section 8.4.2.3) is used to create an AG-chromosome which can be used by the simulation procedure which estimates the NPV of the solution.

The meta-heuristic also verifies that infeasible solutions are discarded. This can occur when a certain solution does not allocate sufficient time to a certain activity type in order to complete it within the timeframe of the simulation procedure. In case such an infeasible solution is detected, the solution is discarded and a new neighbourhood move is tested.

As shown in algorithm 8, the heuristic starts by initialising the temperature (t) and the iteration counter (i). The current best solution is initialised randomly, and the in-

Algorithm 8 Basic Annealing Procedure

```
1: procedure SIMULATED ANNEALING(problem instance)
2:     t ← t_initial
3:     i ← 0
4:     sol_best ← random init
5:     sol_incu ← sol_best
6:     while t > t_end and i < maxiter do
7:         NeighMove(sol_incu)
8:         δ = significantDifference(sol_best, sol_incu)
9:         if δ < 0 then
10:            sol_best ← sol_incu
11:            t ← t · cooldown
12:        else if δ = 0 then
13:            Insignificant difference:
14:            sol_incu ← sol_best
15:        else if e^(−δ/t) < r then
16:            sol_best ← sol_incu
17:            t ← t · cooldown
18:        else
19:            sol_incu ← sol_best
20:        i ← i + 1
21:    return sol_best
```

cumbent is set to be equal to this initial solution (lines 3 and 4). Next the iterations of the simulated annealing heuristic start, and continue until the temperature has become lower than the predefined end temperature (t_{end}) or the maximal number of iterations has been reached (line 5). During each iteration a random neighbourhood move is performed on the incumbent solution (line 6). Next, the difference between the best (sol_{best}) and incumbent solution (sol_{incu}) is analysed using the procedure described in section 8.4.3. This comparison yields one of four outcomes: (i) the solution is better and replaces the current best solution (line 9), or (ii) there is no significant difference between the current best and the incumbent and the move is discarded (line 12), or (iii) the solution is significantly worse but is still accepted according to a certain random probability expressed as a function of the current temperature (line 15), or (iv) none of the above is true and the solution is discarded (line 18). A non-traditional element of this simulated annealing meta-heuristic is that it discards any moves which do not yield a statistically significant difference between the current and incumbent solution.

8.4.4.2 SAT-chromsome Annealing

A variation on the simulated annealing procedure presented in section 8.4.4.1 can be created by using the SAT chromosome rather than the MAT chromosome. The neighbourhood operator for this meta-heuristic increments or decrements a single allele of the chromosome encoding, naturally taking into account the bounds of the individual alleles ($s, d_j \in [0, 11]$). Similarly to the previous meta-heuristic, infeasible solutions due to too little available production time are discarded. All other elements of the heuristic remain

unchanged.

8.4.4.3 Dedicated Heuristics

For problems where evaluating a solution takes a considerable amount of computation time, dedicated heuristics can provide a valuable alternative. Rather than using a random approach to explore the solution space, dedicated heuristics use as much problem information as possible to explore the most promising regions. This section presents a dedicated heuristic which takes a subset of the solution space as defined by the SAT-chromosome and uses a full-factorial approach to explore all solutions therein. This subset can be defined as the SAT-chromosomes (see equation 8.14) for which the following condition holds:

$$d_j = d_k \qquad \forall k, j = 1, ..., NJ | j \neq k \tag{8.18}$$

Equation 8.18 simply states that the number of months during which activity types are halted because of averse weather conditions is equal across all activity types. This effectively reduces the SAT-chromosome to the following expression:

$$[s, d], \qquad s, d \in [0, ..., NP - 1] \tag{8.19}$$

Where d represents the number of months of downtime for adverse weather conditions for all activities. Hence, the total number of combinations is reduced to NP^2. Given that the number of seasons distinguished by weather models rarely exceeds the twelve months of the year. Hence, the computational effort needed to calculate all these combinations is unlikely to exceed computational limitations, even for realistically sized projects.

8.5 Case Study: the Construction of an Offshore Wind Farm

The objective of this research is to present tools which can be used for realistically sized problems which are subjected to substantial weather risks. An example with increasing relevance in recent years is the construction of offshore wind farms. This section presents a project consisting of 54 wind turbines which have been installed on the Thorntonbank off the Belgian coast, having a total capacity of 325 megawatt.

Given that the objective is to optimise the scheduling of the project taking into account the weather conditions, the activity breakdown structure should be sufficiently detailed to incorporate all differences in weather sensitivity between activities. Practically this means that 12 different activity types are distinguished. The properties of these activity types

Activity Type	Activity Subtype	Costs in 000 € C_i^F	C_i^V	Δ-duration (h)	Weather restrictions Wave (m)	Wind (m/s)	Wavest.	Windst.
Scour protection	Scour protection	64.87	3.17	(20, 24, 28)	1.5	-	5	-
Foundation	Loading	3,192.69	9.72	(5, 6, 8)	-	10	-	4
	Transport	0.00	9.72	(4, 6, 8)	2	14	7	7
	Positioning	0.00	9.72	(2, 4, 5)	1.75	10	6	4
	Installation	0.00	9.72	(11, 15, 22)	1.75	9	6	3
	Return to port	0.00	9.72	(4, 6, 8)	2	14	7	7
Turbine	Loading	8,779.91	9.72	(9, 10, 12)	-	10	-	4
	Transport	0.0 0	9.72	(4, 6, 8)	2	14	7	7
	Positioning	0.00	9.72	(2, 4, 5)	1.75	10	6	4
	Installation	0.00	9.72	(24, 27, 34)	1.75	8	6	2
	Return to port	0.00	9.72	(4, 6, 8)	2	14	7	7
Inner array cable	Inner array cable	508.84	28.50	(40, 80, 160)	1.75	-	6	-

Table 8.6: Costs, durations and weather restrictions associated with the different activity types.

Resource	Costs in 000 € ρ_r^V	ρ_r^C	ρ_r^D	Renting restrictions in days min rent period	min down period
Stone dump vessel	13.5	600	600	30	90
Jack-up barge	41.445	1000	1000	90	90
Trenching vessel	20.25	900	900	30	90

Table 8.7: Properties of the chartered vessels for offshore windfarm construction.

are listed in table 8.6. The following paragraphs provide a more detailed insight in the properties of these activities, as well as their interrelations.

In order to construct an offshore wind farm, four basic tasks must be performed. First and foremost, the place where the foundation will be placed has to be prepared by installing scour protection (S). This is done by using a stone dumping vessel to place rocks that prevent ocean current from digging out the monopile foundations out from under the wind turbine. Secondly, the foundation of the turbine (F) has to be installed using a jack-up barge which hammers the foundations into the seabed. Once this has been completed, the third activity is installing the turbine (T) itself, including the tower, navette and blades. Finally the fourth task is to dig a cable (C) into the seabed which connects each of the turbines to the transformer station. This final installation is done using a specially

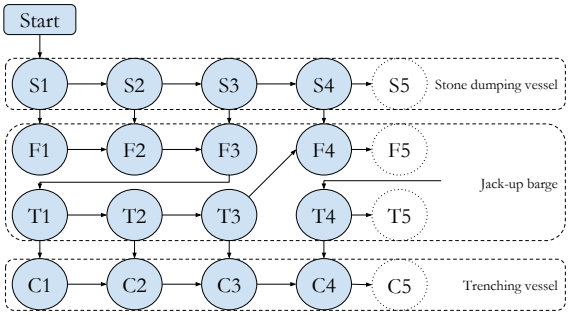


Figure 8.6: Wind farm project structure

equipped trenching vessel.

Figure 8.6 presents a high-level overview of the interrelations between the different activities. Naturally, the order which has been described has to be followed for each individual turbine: first the scour protection is installed, then the foundation of the turbine is installed, next the turbine itself is placed on op of the foundation and finally the cables are installed. However, because of the technical capabilities of the vessels used for these installations the installing of the foundation and the placement of the turbine are grouped. The reason for this is that the jack-up barge is capable of transporting three foundations or three turbines simultaneously. Hence the pattern as is shown by figure 8.6 is created.

A more detailed view is shown by table 8.6, which shows that both the installation of the foundation and turbine can be split up into more detailed sub-activities: loading the barge with the materials (incurring a substantial fixed material cost C_i^F), transporting the materials to the installation site, positioning of the barge for the installation, the installation of the turbine or foundation and the return to shore. There is also an operating cost associated with each of these activities (C_i^V), which represents the variable operating cost of the vessel and its crew, this is the surplus cost compared to the costs which are incurred when a chartered vessel is idle. The chain of events also respects the order which is shown by figure 8.6: first materials for three foundations or three turbines are loaded, after which these are transported to the site. Next the barge needs to be positioned for each turbine location after which the associated installation takes place. When all materials which were transported have been installed the barge returns to the dock for its next shipment.

Each of these activities has an associated triangular distribution[3] which represents the duration of the associated activity. Table 8.6 also shows the weather restrictions for each of the activities. It can be seen that the installation of the turbine is most susceptible to averse weather conditions, followed by the installation of the foundation, placement of the inner-array cable and the scour protection respectively.

The scheduling procedure as defined in the preceding sections is implemented on the level of the activity sub-types. The reason for this being the more realistic representation of weather sensitivity as well as the uncertainty related to each of the sub-types. Hence, 12 activity types are distinguished in the scheduling algorithms used to solve this case study.

Table 8.7 provides more information on the properties of the vessels used to execute the activities. As indicated in figure 8.6, the scour protection is installed by a stone dump vessel, the installation of both the foundation and the turbine is done by a general-purpose jack-up barge, and the installation of the cables on the seabed is done by a trenching vessel. Each of these vessels has an associated chartering cost ρ_r^V which is expressed per time

[3]These triangular distributions are defined as: (min, modulo, max).

period of 4 hours. Moreover a substantial cost is also incurred for the deployment (ρ_r^C) and decommissioning (ρ_r^D) of these vessels. Also related to the (de)commissioning of the vessels are the minimal time periods for both renting and downtime. These limitations are imposed by the firm which rents out the vessel in order to guarantee that vessel occupancy rates are satisfactory. The periods during which vessels have to be rented as well as how often these vessels are (de)commissioned are determined during the simulation procedure (see section 8.4.2).

8.6 Evaluating Computational Performance

In order to evaluate the performance of the solution techniques proposed in section 8.4, two benchmarks are proposed in this section. The first is an upper bound on the net present value, which is based on a best-case scenario execution of the project (section 8.6.1). The second benchmark is the current deterministic scheduling approach used when scheduling similar projects (section 8.6.2).

8.6.1 Upper Bound for the NPV

An upper bound for the net present value earned can be calculated by making a number of assumptions regarding the manner in which the project is executed. Based on these assumptions the activity starting (s_i) and finishing (f_i) times can be obtained, which can then be used to make an estimation of the maximal possible value of the project using equation 8.1.

Assumption 1: All activities have their shortest possible duration. As explained in section 8.3.2 activities are inherently variable, however for the upper bound it will be assumed that all activities are executed according to their most optimistic time estimate. This guarantees a minimal amount of variable costs.

Assumption 2: There are no delays due to weather conditions, i.e. weather conditions are always sufficiently good for the relevant activities to be executed.

Assumption 3: The project is executed using an as late as possible scheduling strategy without extending the critical path as defined by the shortest possible durations for all activities. By doing so all negative cashflows are shifted backwards as much as possible, whilst earning the positive cashflow at the end of the project as soon as possible.

Note that given the structure of the offshore project studied in this chapter (see section 8.5) these three assumptions guarantee continued use of the resources employed. As such, both the size and the timing of the cash outflows linked to resources is a best case scenario.

8.6.2 Deterministic Scheduling

A second benchmark for the performance of the heuristics is the scheduling approach which assumes a deterministic project. This strategy uses the average durations of the activities to create a project schedule. However, noting that there may be some disruptions during the execution of the project the non-critical activities are scheduled as soon as possible (as opposed to the ALAP schedule used for the upper bound, see section 8.6.1).

The performance of such a schedule can be tested using an *AG*-chromosome (section (8.4.1.1). The gate times (g_i) for this chromosome are simply set to the starting times of this deterministic schedule. The performance as expressed by the expected net present value of the project can then be calculated using the simulation procedure described in section 8.4.2.

8.7 Computational Experiments

Intensive computational experiments have been carried out to investigate both the impact of specific strategies as well as the general performance of the different solution heuristics for the case study outlined in section 8.5, as well as variations on the specific scenario with increased and decreased project size. Specifically the impact of varying the starting date of the project as well as the amount of months during which work is interrupted is investigated in section 8.7.1, and more extensive experiments which compare the performance of the different algorithms for projects of various sizes are presented in section 8.7.2.

8.7.1 Impact of Project Start and Downtime

This section presents a qualitative analysis of the impact of the two main decisions by the project manager: when to start the project and how long to interrupt the project in anticipation of inclement weather. These two decisions form the basis of the dedicated heuristic algorithm which was presented in section 8.4.4.3. The aim of this section is to provide insight into the impact these variables can have on the net present value of the project, based on the case study scenario with 54 wind turbines as presented in section 8.5.

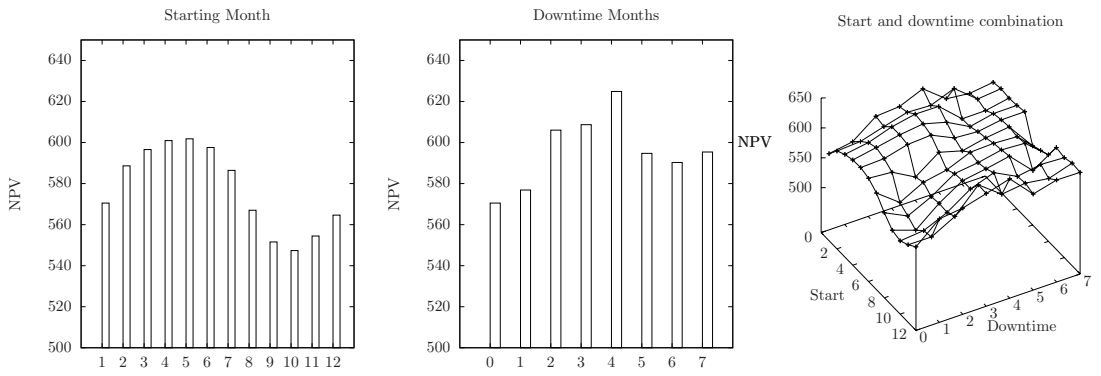


Figure 8.7: Comparison of the impact of varying strategies on the case study project (NPV in mio €).

Naturally, the number of downtime months can only be incremented until the duration of the project extends beyond the maximal allowable deadline (which was set at five years). The experiments showed that setting the winter downtime to more than 7 months resulted in infeasible schedules, hence a total of 96 viable combinations[4] have been tested.

The net present value (NPV) estimate for the various strategies is summarised in figure 8.7. Looking at the leftmost panel of this figure reveals that the NPV displays a clear cyclical pattern when the start of the project is varied. More specifically it appears to be more beneficial to start around the month May, whereas the period surrounding October seems to be the least attractive. The difference between these two months is highly significant: starting in the right month can make a 10% difference in the net present value of the project.

The centre panel reveals a more complex pattern for the number of months the operations are halted during the winter months. The complexity of this pattern is mainly caused by the degree to which chartering costs can be decreased by not working during the winter months, taking into account the restrictions on the minimal time periods during which vessels need to be chartered (see table 8.7). These results do however show that a detailed analysis of the impact of these decisions on a specific project is worthwhile since even a one month difference can have very substantial implications on the net present value of the project. Similarly to the starting month decision, the choice of the number of downtime months can result in a 10% difference in the project's net present value.

A more complete picture of the solution space is presented by the surface plot in the rightmost panel of figure 8.7. This pattern clearly shows that the general patterns which have been uncovered with regard to the number of downtime months as well as the starting

[4] 12 starting months x 8 downtime month possibilities, including zero.

months remain valid when both variables are subject to variations. Nevertheless, it appears that in order to find the optimal combination it pays to perform a full factorial analysis, rather than simply basing the decision on a separate analysis of the starting month and downtime decision. Based on the separate analysis the optimal starting month would have been the 10th month (October), and the optimal number of winter downtime months would be four. However, when looking at the joint analysis it becomes apparent that the combination of these two decisions is sub-optimal, resulting in an estimated NPV of 607 million €, whereas starting in February with five months of winter downtime results in an expected NPV of 626 million €, an increase in NPV of 3.1%.

8.7.2 Comparison of Algorithm Performance

A second computational experiment was carried out to compare the performance of the different heuristic strategies for varying project sizes. The dataset used contained projects of a similar structure to the case study presented in section 8.5 with the number of turbines to install as the determinant of project size. The number of turbines was varied between 1 and 60, each heuristic being used 20 times to get an accurate reading of the average relative performance of these heuristics. A total of 4,500 hours of processing time on 2.5 GHz processors was used to conduct these experiments.

Figures 8.8 and 8.9 compare the performance of the solution approaches in absolute and relative terms respectively. The spread between the upper bound and the solution heuristics clearly increases as the problem becomes larger and systematic weather influences become more problematic.

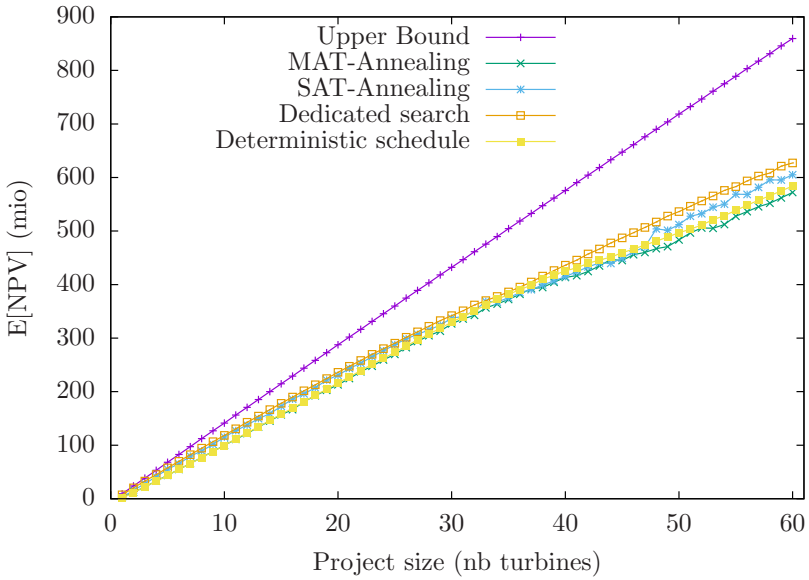


Figure 8.8: Performance comparison in absolute terms

The solution techniques which use the SAT-chromosome (dedicated search and SAT-annealing) outperform the other solution approaches, as evidenced by figure 8.9. Specifically the dedicated search significantly outperforms all other solution methods.

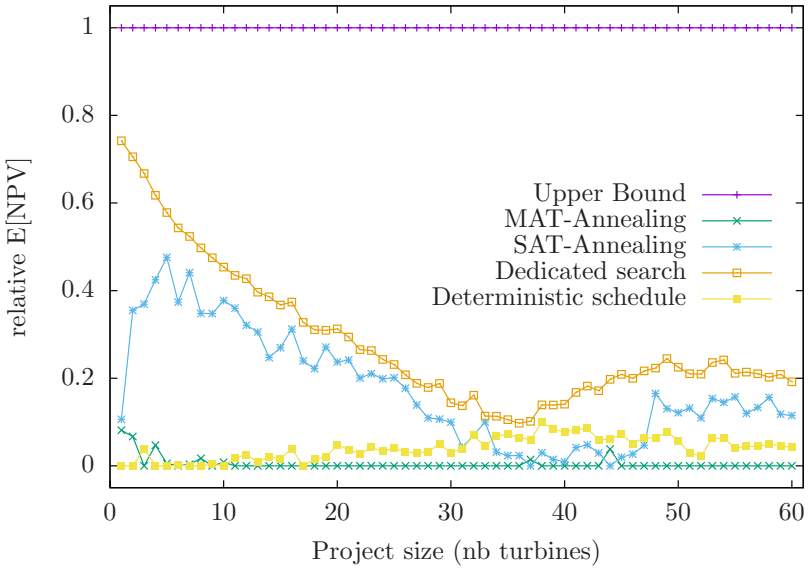


Figure 8.9: Performance comparison in relative terms: The performance of the heuristics is illustrated within the upper and lower bound for the expected net present value, a higher value representing better performance

Interestingly, the MAT-annealing heuristic performs worse than the deterministic scheduling approach in the majority of the cases. This clearly shows that the solution space of this type of problem is too large to be efficiently searched at such a level of detail. This proves the usefulness of the techniques for aggregating the chromosome encoding into higher level encodings as proposed in this research. An important factor here being that these higher-level chromosome encodings still have to be translated into detailed operational guidelines (gate times g_i) since these are required beforehand to guarantee the availability of the installation vessels needed to complete offshore construction projects.

The relative gains from using the SAT-annealing technique rather than deterministic scheduling techniques is highly substantial for smaller projects. As the size of the projects increase the relative impact declines before stabilising at an increase 7 to 10% in expected net present value for projects with 25 turbines or more.

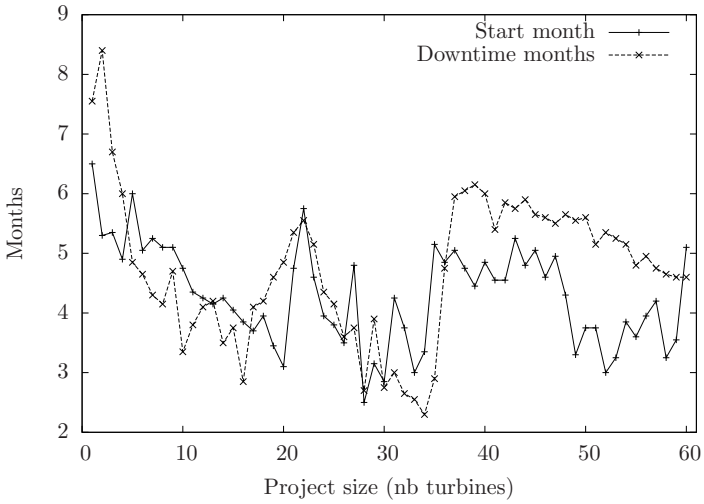


Figure 8.10: The average properties of the optimal solution provided by the dedicated search heuristic for varying problem sizes.

Both the SAT-annealing and the dedicated search heuristic show a dip in their relative performance for project sizes ranging from 35 to 45 turbines. The cause of this dip in performance can be seen when looking at the properties of the average solution provided by the dedicated heuristic (figure 8.10), which clearly shows that the properties of the optimal solution change significantly within this problem size range. This implies that the increasing size of the project has caused an increased sensitivity to the seasonality of the weather patterns. For example a project spanning more than a full year will be guaranteed to be influenced by a period of bad weather, whereas a project of a couple of months can conceivably be planned around worse time periods. Moreover, as the length of the project increases the impact of discounting cashflows to calculate the net present value becomes greater.

8.8 Conclusions

This chapter has presented a novel methodology to optimise scheduling for projects subjected to both inherent and weather-related stochasticity. This methodology consists of a novel weather model which generates both wave height and wind speed, as well as a modelling approach for the project itself.

Computational experiments indicate that the best results are obtained when using dedicated heuristics on a systematically reduced solution space. Hence, future research should focus on efficient ways of identifying the most interesting areas of the solution space

for this type of problem. Moreover, future research should extend the analysis to include other case studies, as well as creating a dataset containing benchmark problems on which future heuristics can be compared objectively. The discussion of other case studies as well as climates could also be a valuable extension of this research. For projects where the lead times for resources are short or non-existent a dynamic scheduling approach may also be an interesting extension of this research.

9

Weather-Based Maintenance Strategies for Offshore Wind Farms

This chapter presents a number of novel strategies for the preventive maintenance of offshore wind farms. These strategies take into account seasonal weather patterns and their impact on electricity production. By optimising the timing of the maintenance the losses due to relinquished electricity production can be reduced.

9.1 Introduction

Whereas chapter 8 studied the influence of weather conditions on the construction of offshore wind turbines, this chapter leverages the weather model presented in chapter 7 to optimise the preventive maintenance activities in offshore wind parks.

The operations and maintenance costs of offshore wind turbines represent a significant fraction of the lifetime cost of offshore wind farms (20 - 35%, Ortegon et al. (2013)). Specifically, the offshore location of these turbines greatly increases the costs associated with the maintenance. This is due to the need for specialised vessels and the fact that the turbines can not always be serviced when weather conditions are unfavourable. The significance of these costs has fostered a number of publications on this subject in recent years. A good overview of this literature is presented by Shafiee (2015).

The main purpose of the models presented in this section is to aid decision makers facing operational go/no-go decisions regarding preventive maintenance. In order to perform this type of maintenance, the turbine has to be switched off, resulting in lost energy production. Hence, it is advised to perform this maintenance when the weather conditions are unfavourable for electricity production (specifically at wind speeds below the rated speed of the turbine). Nevertheless, determining the right thresholds at which to perform this maintenance is a complex task due to the inherent uncertainty of weather conditions.

The nature of the maintenance process as well as the settings of the various parameters in this operational model are based on real case study information. Specifically, a wind farm in the North Sea off the Belgian coast.

9.2 The Nature of Preventive Wind Farm Maintenance

In his recent review of maintenance logistics literature for offshore wind farms Shafiee (2015) distinguishes three different levels of decision making. Strategic decisions that impact the complete life cycle of the wind farm, tactical decisions that are made multiple times during the lifespan of the wind park, and operational decisions which govern the day to day operations of the wind farm. The research presented in this chapter is related to the organisation of maintenance support task within the tactical level, and the scheduling of maintenance tasks situated in the operational level.

Different types of maintenance are required throughout the life of an offshore turbine. Tavner (2012) distinguishes two main categories: preventive maintenance and corrective maintenance. Preventive maintenance can be further categorised as calendar, condition or weather-based maintenance. Corrective maintenance can in turn be either planned or unplanned. Several other authors also use a similar categorisation for the mainte-

nance activities (Karyotakis, 2011; Shafiee, 2015). The research in this chapter specifically aims to decrease the cost associated with weather-based maintenance activities. For the current generation of offshore wind turbines approximately 80 hours of preventive maintenance is recommended, although this amount is frequently substantially greater in practice (Van Bussel et al., 2001).

The weather conditions have a dual impact on the maintenance activities. Firstly, averse weather conditions may prevent maintenance operations altogether for safety reasons. Secondly, the weather conditions during maintenance operations determine the lost income due to stopping the wind turbine. Typically safety thresholds for maintenance activities are set at 150 cm wave height and at a wind speed of 12 meters per second (Maples et al., 2013).

The production output of a wind turbine can be plotted as a function of the wind speed. For this research, a simplified power output curve as used by Gundegjerde and Halvorsen (2012) is employed. Figure 9.1 illustrates the curve that is used for the experiment in section 9.5. Naturally, the nature of this curve depends on the type of turbine.

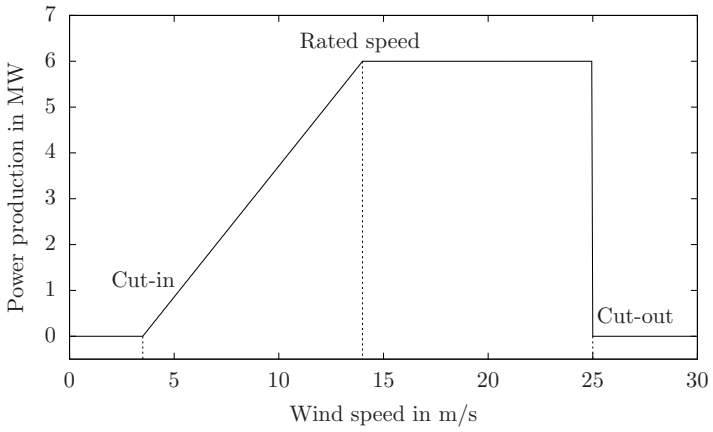


Figure 9.1: Turbine power production as a function of the wind speed.

On this curve three important wind speeds are shown. The cut-in wind speed is the wind speed at which there is sufficient wind to turn the blades of the turbine. Below this wind speed there is no electricity production. As the wind speed increases, so does the power production, here approximated by a linear curve. This increase continues until the rated speed of the turbine is reached, here the power production levels off at its maximum capacity (6MW for this specific type of turbine). For safety reasons, the turbine is shut off when the wind speeds exceed the cut-out rate, at which point the power production falls back to zero.

9.3 Problem Definition

This section defines the preventive maintenance problem starting from the total cost function of the optimising actor. To improve readability, the notation in this chapter is based on a situation with a single turbine. This section starts by analysing the problem under deterministic assumptions in subsection 9.3.1. Next, the assumption of deterministic weather is relaxed and the more realistic stochastic problem faced by the decision maker is defined in subsection 9.3.2.

9.3.1 Deterministic Analysis

This research takes the perspective of the actor responsible for performing the preventive maintenance on a collection of offshore wind turbines. Each of these turbines requires a predefined number of maintenance shifts per annum. A penalty is associated with non-completion of required maintenance shifts. Effectively, the actor wishes to minimize the following total cost (TC) function:

$$TC = CC^{tot} + MC^{tot} + PC^{tot} \tag{9.1}$$

Where CC^{tot} signifies the total cost of lost capacity, MC^{tot} the total actual cost for executing the maintenance and PC^{tot} the total penalty cost for uncompleted maintenance shifts. These three cost components are explored in detail in the following paragraphs.

The cost of lost capacity is calculated based on the power production of the wind turbine (see section 9.2 and figure 9.1). This power production is defined by the following equation:

$$P(WS) = \begin{cases} 0 & \text{if } WS < WS^{\text{cut-in}} \\ P^{\max} \frac{WS - WS^{\text{cut-in}}}{WS^{\text{rated}} - WS^{\text{cut-in}}} & WS^{\text{cut-in}} \leq WS < WS^{\text{rated}} \\ P^{\max} & \text{if } WS^{\text{rated}} \leq WS < WS^{\text{cut-out}} \\ 0 & \text{if } WS \geq WS^{\text{cut-out}} \end{cases} \tag{9.2}$$

Where P is the power production in Megawatt, as a function of the wind speed (WS). $P^{\max}$ is the maximal power production of the turbine. For the turbines studied in this research this is equal to $6MW$. As shown on figure 9.1, there is no power production below the cut-in speed ($WS^{\text{cut-in}}$) and above the cut-out speed ($WS^{\text{cut-out}}$). Moreover, the power production reaches its peak at the rated wind speed of the wind turbine (WS^{rated}) beyond which further increases in wind speed no longer increase the power production.

Using this function, the cost due to lost capacity (CC) can be calculated by multiplying the power production (P) by the duration of the standstill (in hours) and the price received

received per MWh on the electricity market (E). Assuming the turbine is stopped for 8 hours during each of the maintenance shifts, the potential lost capacity cost during shift t can be calculated as:

$$CC_t(WS_t) = P(WS_t) \cdot 8h \cdot E \tag{9.3}$$

The total lost capacity cost (CC) can simply be calculated by summing the lost capacity cost for the time periods during which maintenance was executed:

$$CC^{tot} = \sum_{t=1}^{|T|} x_t \cdot CC_t \tag{9.4}$$

Where x_t is a binary variable equal to 1 when the contractor has decided to perform maintenance, and equal to zero otherwise. The same variable can be used to calculate the total cost of maintenance (MC), as shown in equation 9.5. Where MC^{shift} is the maintenance cost for a single shift. Note that this cost is not time dependent.

$$MC^{tot} = \sum_{t=1}^{|T|} x_t \cdot MC^{\text{shift}} \tag{9.5}$$

The third component of the cost represents the penalty for unfinished maintenance. Given that a total of S maintenance shifts per turbine are contractually required within the time period under consideration, the total penalty cost can be represented by equation 9.6.

$$PC^{tot} = \max\left[0, \left(S - \sum_{t=1}^{|T|} x_t\right)\right] \cdot PC^{\text{shift}} \tag{9.6}$$

Using this information and assuming that the observed wind speed is a deterministic parameter to the model, the maintenance scheduling problem can be defined as a simple model as described by expressions 9.7 until 9.13.

Minimize:

$$\sum_{t=1}^{|T|} \left[(CC_t + MC^{shift}) \cdot x_t\right] + PC^{tot} \tag{9.7}$$

subject to:

$$PC^{tot} \geq \left(S - \sum_{t=1}^{|T|} x_t\right) \cdot PC^{\text{shift}} \tag{9.8}$$

$$PC^{tot} \geq 0 \tag{9.9}$$

$$WH^{\max} - WH_t + M\alpha_t \geq 0 \quad \forall t \in T \tag{9.10}$$

$$WH_t - WH^{\max} + M(1 - \alpha_t) \geq 0 \quad \forall t \in T \tag{9.11}$$

$$x_t + \alpha_t < 2 \tag{9.12}$$

$$x_t, \alpha_t \in \{0, 1\} \quad t \in T \tag{9.13}$$

In this linear model, an additional constraint related to the maximal allowable wave height to conduct maintenance is represented by equations 9.10 until 9.12. These equations introduce a new binary variable α_t which is equal to 1 if the maximal allowable wave height is exceeded. Equation 9.12 guarantees that maintenance will not be allowed during time periods where the maximal wave height is exceeded.

This deterministic linear model can provide a useful lower bound for the stochastic optimisation of the maintenance operations. Moreover, by making the assumption that the penalty cost per shift exceeds the maximal cost of capacity loss and the cost of maintenance itself, the lower bound for the optimal solution of the stochastic procedure can be determined very easily. This assumption can be noted as follows:

$$PC^{shift} > CC_t(WS^{\text{rated}}) + MC^{\text{shift}} \tag{9.14}$$

If this condition is true, the optimal solution is always to perform all the maintenance during the decision period - insofar as this is not inhibited by excessive wave heights. Hence, the set of optimal time periods for maintenance T^{opt} can be defined as follows:

$$T^{\text{feasible}} = \{t \in T : WH_t \leq WH^{\max}\} \tag{9.15}$$

$$\exists T^{\text{opt}} \subseteq T^{\text{feasible}} : |T^{\text{opt}}| = S, WS_t \leq WS_k | t \in T^{\text{opt}}, k \in T^{\text{feasible}} \setminus T^{\text{opt}} \tag{9.16}$$

under this assumption the optimal solution is simply:

$$x_t = 1 \quad \forall t \in T^{\text{opt}}, \quad x_k = 0 \quad \forall k \in T \setminus T^{\text{opt}} \tag{9.17}$$

9.3.2 Stochastic Extensions

The initial analysis in section 9.3.1 assumed the observed wind speed to be deterministic. In reality such long term weather predictions are impossible and form the main complicating factor in this optimisation problem. Operationally, the decision to perform preventive maintenance is made on a daily basis. Moreover, there are no formal rules in place that govern this go/no-go decision. Hence, the creation of a tool to optimise this process is one of the key factors where the preventive maintenance process can be improved.

In order to evaluate the performance of the proposed maintenance strategies, a simulation engine is required. The basis of this simulation engine is a weather simulation that generates realistic wind speeds and wave heights for the annual time horizon of the preventive maintenance contract. In order to create these weather patterns, the weather simulation model presented in chapter 7 is used. This model is extended in order to output specific wind speeds and wave heights, rather than simply a category for these values, as was sufficient for the experiments in chapter 8. The motivation for this exension is that this allows for a more detailed view on the cost of lost capacity (see section 9.3.1). In order to approximate the actual distributions of wind speeds and wave heights as good as possible, the recently proposed approach proposed by Tang et al. (2015b) was used. Using this approach, the wind speeds and wave heights are drawn at random from the actual observations in the training data, given the simulated wind or wave state as presented by the simulation model from chapter 7.

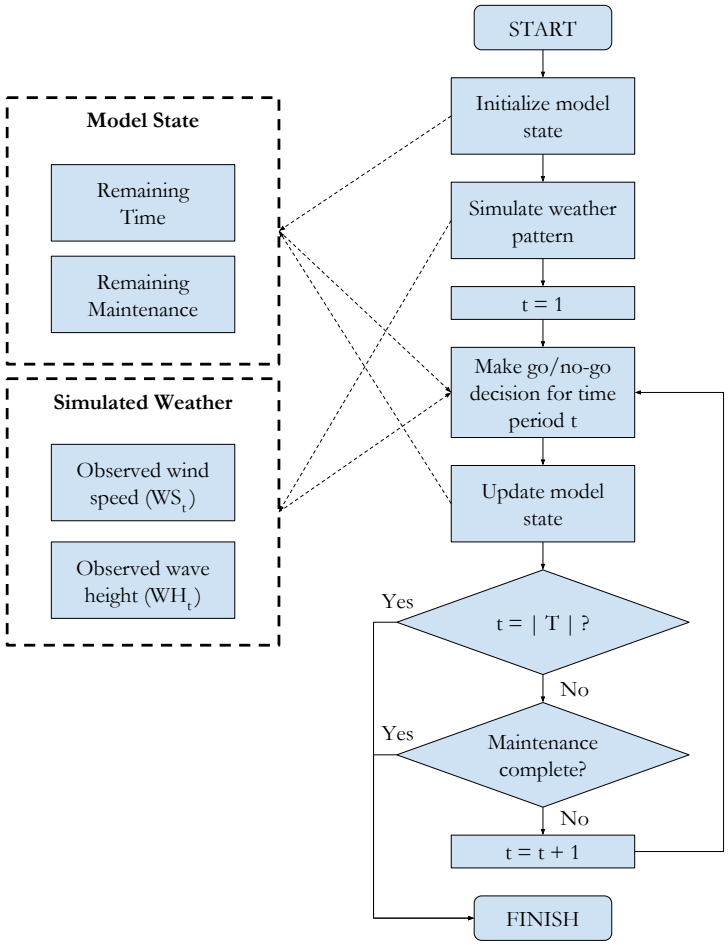


Figure 9.2: Outline of the decision making process.

The logic of the stochastic process is outlined in figure 9.2. The two major components of the process are the simulated weather conditions and the state of the maintenance operations (*model state*). The simulation procedure starts by initialising the model state, i.e. setting the remaining number of time periods equal to the total number of available time periods in the contract and setting the remaining number of maintenance shifts equal to the total number of maintenance shifts required. Next, the weather simulation engine is called and wind speeds as well as wave heights are simulated for the complete time period. The time period index t is then initialised to 1 and the simulation loop is started. At every iteration of the loop a go/no-go decision regarding the maintenance operations is made. At this position in the simulation loop a number of different decision making models can be inserted in the simulation (see section 9.4). Depending on the nature of the decision the state of the model is updated. Next, the stopping conditions are tested. The simulation

halts when the last time period is finished, or when all the necessary maintenance has been executed. If the stopping conditions are not yet satisfied, the time period counter is incremented and the loop continues.

9.4 Maintenance Strategies

As described in section 9.3.2 any type of logic can be used to make the go/no-go decision regarding the maintenance of offshore wind turbines. In this section a number of different approaches are presented prior to comparing their performance using computational experiments in section 9.5.

9.4.1 Static Threshold

The simplest technique which can be employed is a simple static threshold for the wind speed. The decision maker simply selects a specific wind speed and if the observed wind speed for the next shift does not exceed this threshold, maintenance is executed. Naturally, there are two preconditions to this. Firstly, the wave height must not exceed the maximal allowable wave height (i.e. performing maintenance has to be technically feasible). Secondly, the total number of maintenance shifts will of course never exceed the number of contractually required maintenance shifts.

The value of the threshold is optimised by testing a wide range of values on a set of training data. The value of the threshold is varied in the range $[WS^{\text{cut-in}}, WS^{\text{rated}}]$ since a lower threshold could never be economically relevant (it does not make sense to delay maintenance when the turbine has already stopped). A higher threshold is also not relevant since the output of the wind turbine is already at maximal capacity when the wind speed is equal to the rated wind speed.

The training procedure for this technique is very simple and uses the number of values tested as the only parameter. Based on this parameter a number of equidistant points are selected in the range $[WS^{\text{cut-in}}, WS^{\text{rated}}]$. Each of these candidate thresholds is tested on all of the members of the training dataset and the average total cost is reported for each of the candidate values. The threshold value with the lowest average value is then selected as the static threshold to be used.

This technique corresponds most closely to the method currently employed by the scheduler. The key difference being that the technique presented here uses a set of training data, whereas in practice this value is based on intuition.

9.4.2 Relative Workload Dependent Threshold

The main drawback to the static threshold approach is that it does not take into account all information available to the scheduler. Specifically, the information on the amount of maintenance already completed and the amount of time remaining to perform this maintenance is neglected. Nevertheless, this information is highly valuable when making the go/no-go decision.

The relative workload dependent threshold ($RWDT$) presented in this section mitigates this issue by including these elements in a simple expression, as shown by equation 9.18. This equation contains two parameters that have to be tuned to the specific problem. These parameters represent the intercept (p_1) and the importance of the workload (p_2) respectively. The workload itself is included as the ratio of the total number of shifts remaining versus the total number of time periods remaining. This ratio is then multiplied with the second parameter p_2 in equation 9.18.

$$\tau_t = p_1 + p_2 \cdot \frac{S - \sum_{t=1}^{|T|} x_t}{|T| - t} \tag{9.18}$$

A wide variety of techniques can be used to tune the two parameters in this decision model. The simplest of which is a simple full-factorial approach that attempts to cover all possible combinations of parameter values over a reasonable range. Alternatively, more advanced heuristic approaches can be used to train this model. Because of the high cost associated with evaluating the performance of these heuristics, local-search based rather than memetic algorithms are preferred for this task. The main reason for this is the advantage these techniques can obtain by using common random numbers to reduce the number of evaluations needed to determine that a move is significantly better or worse than another solution (see section 8.4.3).

The tuning method employed in this research starts with a grid search of the most likely parameter values of the $RWDT$ measure. Once this initial search has been completed, the best solution is used as a starting point for a simulated annealing heuristic which explores the neighbourhood of the solution by changing one of the two parameter values of the $RWDT$ equation. The annealing procedure itself is highly similar to the one described in section 8.4.4.1.

The grid search for the initial best solution is carried out by varying the p_1 value in the range $[0, WS^{\text{rated}}]$, and the p_2 value in the range $[0, \frac{WS^{\text{rated}}}{5}]$. In both these ranges 10 equidistant values have been selected, and all possible combinations for these two ranges are then tested on the complete training dataset. The solution resulting in the lowest total cost is taken as the initial seed solution for the $RWDT$ parameter optimisation. Where

Algorithm 9 Simulated Annealing Procedure for *RWDT*

```
1: procedure SIMULATED ANNEALING(problem instance)
2:     t ← t_initial
3:     i ← 0
4:     sol_best ← best grid search solution
5:     sol_incu ← sol_best
6:     while t > t_end and i < maxiter do
7:         NeighMove(sol_incu)
8:         δ = significantDifference(sol_best, sol_incu)
9:         if δ < 0 then
10:            sol_best ← sol_incu
11:            t ← t · cooldown
12:        else if δ = 0 then
13:            Insignificant difference:
14:            sol_incu ← sol_best
15:        else if e^(−δ/t) < r then
16:            sol_best ← sol_incu
17:            t ← t · cooldown
18:        else
19:            sol_incu ← sol_best
20:        i ← i + 1
21:    return sol_best
```

the grid search is only capable of covering a limited fraction of the solution space, the simulated annealing heuristic uses a move that is theoretically capable of reaching every point in the solution space.

During the simulated annealing procedure, a very simple neighbourhood move is used. This neighbourhood move changes one of the two parameters with equal probability. The new parameter value is obtained by multiplying the current value with a random draw from a uniform distribution $U[0.5, 1.5]$. New solutions are then evaluated based on their being significantly different from the current best solution, taking common random numbers into account. Based on extensive parameter tuning, it was found that the following parameters yielded the best results for the simulated annealing training heuristic.

- $t_{initial}$: 10,000
- t_{end}: 10
- cooldown rate: 0.99

These values were obtained by using a set of 250 weather patterns, which are naturally different from the patterns used in the actual experiments in order to avoid overfitting.

9.4.3 Support Vector Machine

Defining a workload-dependent threshold already includes substantially more information. However, this approach implicitly assumes a specific nature of the relation between these inputs and the best possible threshold values. This relation is reflected in the specific form of equation 9.18.

An alternative which mitigates this issue is to use supervised learning techniques to make the go/no-go decision. A support vector machine (SVM) is one of the most frequently employed techniques capable of making a binary classification. This technique was originally introduced by Cortes and Vapnik (1995). The main objective of these support vector machines is to find a hyperplane that is capable of separating observations in their respective classes. The remainder of this chapter assumes a basic knowledge of the workings of support vector machines, the reader who is not familiar with these techniques is referred to the excellent introductory works by Burges (1998); Mangasarian (2003); Smola and Schölkopf (2004).

In order to train the support vector machines, the same training data as used with the other techniques has been used. The 'correct' classification of the data points as either 'go' (perform maintenance) or 'no-go' (don't perform maintenance) was determined using the method described in section 9.3.1. The support vector machine is presented with four input parameters in order to make its decision: the number of maintenance shifts that still have to be completed, the number of shifts remaining during which this maintenance can be carried out, the observed wind speed and the observed wave height. Conforming to standard practice, these value shave been rescaled in the range $[0, 1]$.

The settings of the macro-parameters for the support vector machines is done using a grid-search methodology employing a 3-fold test set-up to avoid over fitting the training data. Beyond the basic parameters, the grid search was also used to test the performance of linear, polynomial and radial basis function kernels.

9.4.4 Artificial Neural Networks

Artificial neural networks (ANN) are an alternative to the support vector machine technique. As many techniques in this domain, this technique is inspired by biology, specifically the central nervous system. The technique itself was pioneered in the 1940s by McCulloch and Pitts (1943), and has experienced a significant surge in popularity in recent years, with a substantial number of variations and specialised adaptations for specific challenges. Readers who are unfamiliar with this technique are referred to the excellent introductions by Picton (2000) and Hastie et al. (2005).

Similarly to the support vector machines, the ANN technique does not require any explicit assumptions about the underlying nature of the process. Hence, it can be applied to a wide range of problems. In this research a three-layered perceptron was used, as is visualised in figure 9.3. This perceptron uses the same four input parameters as presented to the support vector machine method.

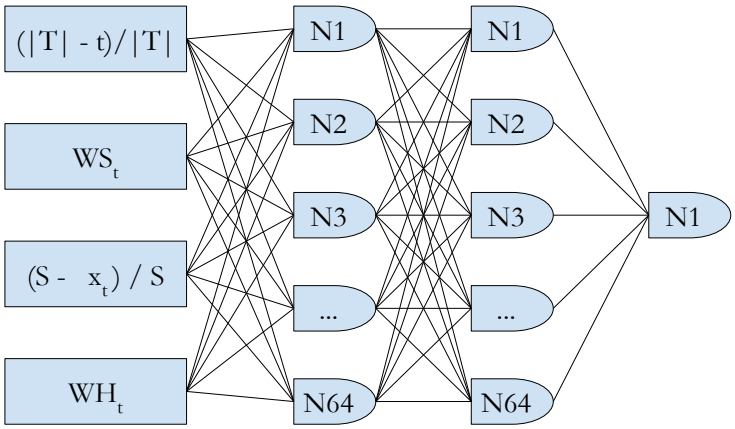


Figure 9.3: Visual representation of the neural network structure.

The perceptron uses two 64-neuron layers and one single neuron layer as the output layer. The former two layers both use a rectifier activation function, which allow for faster and more effective training than traditional sigmoid functions. A 50% drop-out for the interconnections was used in order to prevent overfitting, in combination with a 3-fold test design similar to the one used by the support vector machine training procedure. The performance of the neural network was evaluated using a binary cross-entropy loss function (De Boer et al., 2005), and root-mean-square propagation was used to optimise the network's performance (Tieleman and Hinton, 2012).

9.5 Computational Experiments

The performance of the different solution methods has been evaluated by means of computational experiments. These experiments are based on real life data, gathered from a wind turbine park off the Belgian coast. The specific parameters derived from this case are summarised in section 9.5.1. Next, the training of the different solution techniques is discussed in section 9.5.2. Finally, the performance of the methods is compared in section 9.5.3.

9.5.1 Parameters

The computational experiments presented in this section are based on real information from a wind turbine park off the Belgian coast in the North Sea. Based on information received from professionals, the parameters have been set at the appropriate levels for the maintenance scheduling problem faced by this specific company. Naturally, several of these

parameters can be significantly different for wind parks in other regions or parks that use other technology. An overview of these parameters is presented in table 9.1.

Parameter	Value
$P^{\max}$	$6MW$
$WS^{\text{cut-in}}$	$3.5m/s$
WS^{rated}	$14m/s$
$WS^{\text{cut-out}}$	$25m/s$
E	$101.29EUR/MWh$
$WH^{\max}$	$150cm$
S	$10/turbine$
$\|T\|$	365
MC^{shift}	$1,000.0$
PC^{shift}	$7,500.0$

Table 9.1: Parameter values

The wind turbine park consists of 60 wind turbines in total, and five of these wind turbines can be serviced simultaneously during any given time period since there are five crews available. Note that the actual costs have been changed, but the order of magnitude of these costs remains correct.

9.5.2 Training

In order to train the parameters of the three proposed solution approaches (see section 9.4), 250 weather patterns have been used. Each of these weather patterns describes the weather during a complete year (365 days), which is equal to the time horizon during which the preventive maintenance has to be executed. Hence, the training data set consists of a total of $91,250$ go/no-go decisions. Naturally, the same training data is used for all the solution methods in order to provide a fair comparison of the performance of the different techniques. The values reported here are of course specific to the nature of the wind park, as well as the meteorological conditions. Hence, for other scenarios this training should be repeated since they are likely to yield significantly different results.

9.5.2.1 Static Threshold

The training results for the static threshold technique are visualised in figure 9.4. A clear-cut optimum can be defined for this technique; setting the threshold at a wind speed of $6.75m/s$.

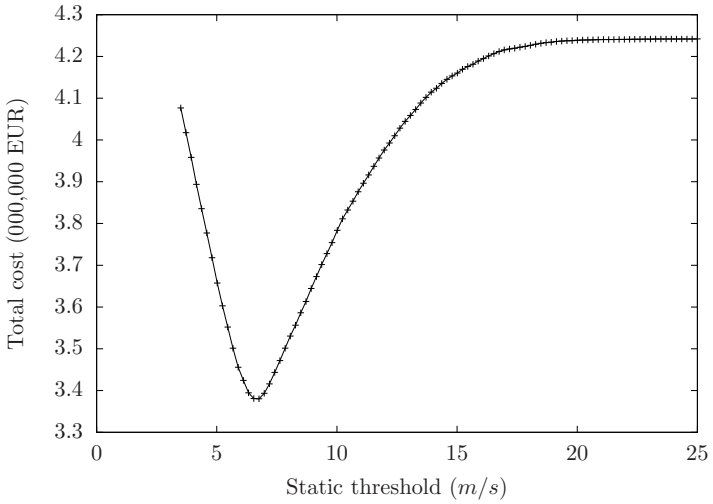


Figure 9.4: Training results for the static threshold technique.

9.5.2.2 Relative Workload Dependent Threshold

The *RWDT* has been trained using the method outlined in section 9.4.2. Using this technique the best parameter values for equation 9.18 have been identified as 4.67 for p_1, and 1.24 for p_2 respectively.

9.5.2.3 Support Vector Machine

The grid-search training procedure for the support vector machine identified the second degree polynomial kernel as the most effective for the specific setting investigated in this computational experiment.

9.5.2.4 Neural Network

The 3-fold approach to training the neural network indicated that the model started overfitting after approximately 80 epochs. Hence, this threshold has been used to train the network described in section 9.4.4 using the complete set of training data.

9.5.3 Computational Results

The performance of the different solution techniques is tested on a dataset containing 2,500 weather patterns, each representing a full year of weather data. Hence a total of 912,500 go/no-go decisions have to be made. The optimal solution for each of these scenarios can easily be calculated based on the logic explained in section 9.3.1. This

deterministic optimum can then be used as a lower bound for the stochastic optimisation. The time required to make individual go/no-go decisions is negligible for each of the solution techniques, and are therefore not reported.

Figure 9.5 shows the average total cost for each of the solution techniques. It is immediately apparent that the simple static threshold performs remarkably well. Nevertheless, taking into account the relative workload remaining ($RWDT$) does succeed in making a significant improvement upon the static method. Both machine learning techniques perform roughly on-par with the static threshold method. The under par performance of the machine learning techniques indicate that at least in their current form they are unable to capture the inherent logic that is used by the $RWDT$. However, it is possible that by experimenting further with other kernels, network structures and other meta-parameters the performance of these techniques can be improved substantially.

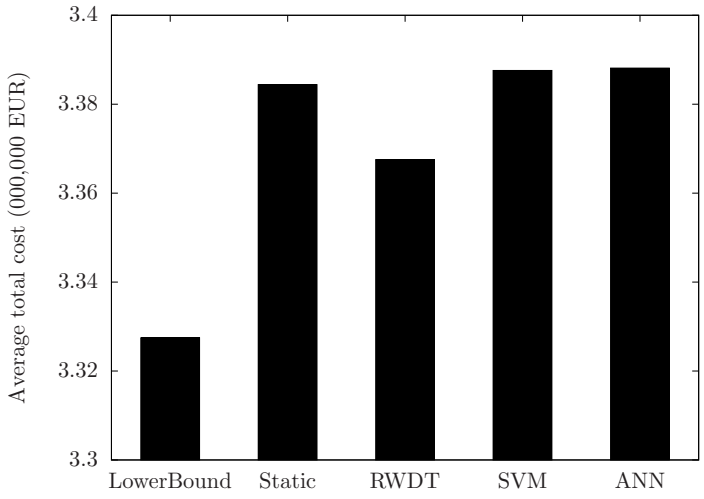


Figure 9.5: Comparison of the average total costs when using different solution techniques.

An important note here is that the actual static threshold used in practice is set based on intuition rather than extensive simulations. Hence, it is likely that the performance of the threshold used in practice is considerably worse than the performance reported here (see figure 9.4, which shows a steep increase in average cost even for minor deviations from this optimal value.)

Figure 9.6 gives insight in the cost components when using the different approaches. As described in section 9.3.1, the total costs are a combination of the lost capacity cost (CC^{tot}), the direct cost of maintenance (MC^{tot}) and the penalty costs (PC^{tot}).

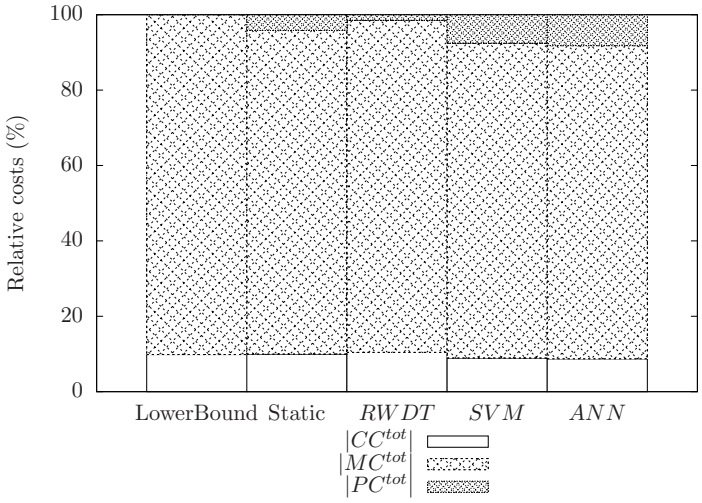


Figure 9.6: Detailed look at the composition of the costs when using different solution techniques.

The lower bound does not include any penalty cost, since this solution is based on a deterministic analysis and under full information it is never advisable not to execute maintenance. A key observation when comparing the different solution techniques is that the $RWDT$ includes a substantially smaller fraction of penalty costs when compared to other solutions. This is another indication that the technique is to be preferred over the static threshold technique, since a solution with a lower penalty cost should be preferred over a solution with a higher penalty cost, even when total costs are identical. Moreover, the results also show that both SVM and ANN solution techniques result in substantially larger penalty cost fractions when compared to the other solution techniques.

Figure 9.7 shows the standard deviation of the total cost for the different solution techniques. This graph shows that the inherent variability of the outcomes is not inflated by using a static threshold. The variability of the results increases slightly when moving to a $RWDT$ decision technique, and even further when moving to SVM or ANN methods. Nevertheless, it appears that this increase in variability is rather limited.

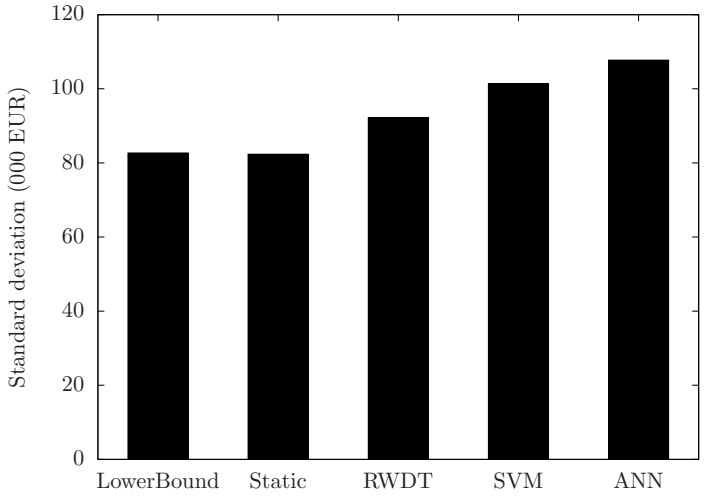


Figure 9.7: Comparison of the standard deviation of the outcomes for the different solution techniques.

9.6 Conclusions

This chapter has presented a new methodology for the optimisation of preventive maintenance of offshore wind turbines. Using computational experiments, the merits of a number of different techniques to govern the go/no-go decision have been analysed. These experiments tested the performance of a simple static threshold, a workload-dependent threshold as well as support vector machines and artificial neural networks.

The results of these experiments indicate that using a workload-dependent threshold can significantly improve the profitability of the preventive maintenance activities. Moreover, for the specific case study analysed in this chapter the static threshold performed remarkably well. The more advanced machine learning techniques were unable to provide convincing results for this specific problem.

This study has also opened a number of interesting avenues for future research. Firstly, using this approach in another operational setting with more volatile weather conditions could lead to another solution technique being preferred. Secondly, this model could also be adjusted for use in tactical rather than operational decision making. Specifically, deciding on the capacity and availability of maintenance crews.

9.A Overview of Notation

Table 9.2 presents an overview of the most important notation used in this chapter.

Variable/Parameter	Description
τ	Wind threshold for maintenance operations
t	Index of the time periods
T	Set of time periods considered
$\|T\|$	Total number of time periods considered
TC	Total cost associated with maintenance
CC	Cost of lost capacity due to stopping turbines for maintenance
MC	Direct costs associated with the maintenance activities
PC	Penalty cost incurred for not completed maintenance shifts
P	Power production (in MW)
$P^{\max}$	Maximal power production of a turbine
WS	Wind speed
WS_t	Observed wind speed in time period t
$WS^{\text{cut-in}}$	Wind turbine cut-in wind speed
WS^{rated}	Wind turbine rated wind speed
$WS^{\text{cut-out}}$	Wind turbine cut-out wind speed
WH	Wave height
WH_t	Observed wave height at time period t
$WH^{\max}$	Maximal allowable wave height for maintenance operations
E	Electricity price
S	Number of required maintenance shifts in the time period
α_t	Binary variable =1 if WH_t exceeds $WH^{\max}$

Table 9.2: Overview of notation

10

Conclusions and Future Research Avenues

This dissertation has extended two sub-domains of operations research that were but sparsely considered in pre-existing literature. The first part of this dissertation has presented a number of novel techniques to deal with the adoption of incentive contracts in a project-based context. The focus of this research has been the quantifiable aspects of these contracts. These aspects were analysed from the perspective of both the owner and the contractor hired to execute the project. This includes the optimisation of the contract design, as well as the optimisation of project schedules and improvements for project control.

The second part of this dissertation investigated how the scheduling of weather sensitive projects could be improved by making use of recent advances in weather modelling techniques, as well as the improved availability of weather data. This was specifically applied to offshore projects which are highly sensitive to wind speeds and wave heights during construction and maintenance operations.

This chapter presents an overview of the key findings of this research, as well as the potential avenues for future research that were uncovered by this research.

10.1 Research Summary

10.1.1 Part I: A Quantitative Analysis of Incentive Contracts for Projects

In chapter 2, an overview of pre-existing literature on incentive contracting has been presented. This review uncovered a large body of literature, but few publications that were able to provide prescriptive and quantitative guidance to either the owner or contractor. Rather, the majority of literature remains rather qualitative and case-based. The identification of this research opportunity was the motivation for the first part of this dissertation.

Further investigation revealed significant opportunities for cross fertilisation of literature on incentive contracting and the more quantitatively oriented areas of project management research, such as project scheduling and project control. These domains were able to provide a solid foundation on which prescriptive models for incentivised projects could be based.

Chapter 3 presented a novel high-level model of the relationship between the project owner and contractor, as visualised in figure 10.1. This model consists of three components: the contract model, the trade-off model and the evaluation model. The contract model is a generalisation of the majority of incentive contracts that have been analysed in the literature (as was investigated in chapter 2). This model can represent linear, piecewise linear as well as quadratic pay-off equations, as well as lump sum awards related to specific project deadlines. The trade-off model is the second component of this novel methodology. This component builds on existing project management literature, specifically the quantification of the trade-offs faced in a project scheduling. This pre-existing research was extended in order to create a four-dimensional trade-off capable of representing the dynamic of a project underlying an incentive agreement. The nature of this trade-off is based on a set of six basic axioms that describe the relationship between these four dimensions: project cost, duration, scope and the effort invested by the contractor. Contractor effort in this context comprises any action taken by the contractor to improve the performance of the project (in terms of cost, duration or scope) to which (s)he is not contractually obliged and which cannot be directly monitored by the owner of the project. Finally, the third component of the model proposed in this chapter is the evaluation model which translates the outcome of a project into a financial valuation for both the owner and contractor.

The prosed model is then used to investigate the contract design problem from the perspective of the project owner. The latter having two objectives: firstly, the maximisation of his/her profits, and secondly, the alignment of the contractor's pay-offs with his/her own preferences. Such an alignment effectively limits the risk of agency conflicts where it is in the contractor's best interest to act against the desires of the project owner. As

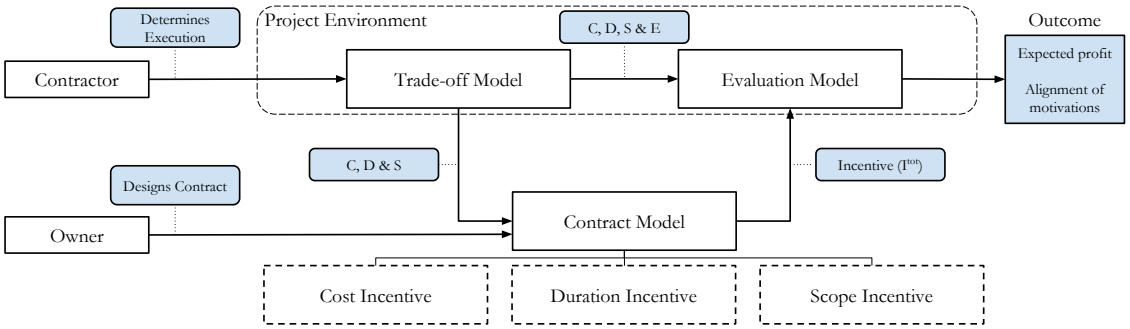


Figure 10.1: Strategic model of the contract design problem.

such, a contract can only be considered objectively better than another contact when it dominates the other contract on both dimensions.

An extensive computational experiment was set up in order to provide prescriptive guidelines to project owners when selecting incentive contracts for specific environments. The performance of various sets of contracts being evaluated based on (i) the distance to the global Pareto frontier as calculated by the average Hausdorff distance, (ii) the overall spread of the solutions and (iii) the number of elements in the set of solutions. Based on these experiments a number of specific guidelines for contract design from the owner's perspective can be made, which are summarised the table below.

	Experiment	
	Uni-Dimensional	**Multi-Dimensional**
Contract type	Non-linear and Piecewise contract outperform linear contracts. Avoid transfer of all risk to one party. Preferred types are robust to imposed restrictions. Skewness persists within contract types.	More dimensions improve performance. Duration > Scope > Cost. Avoid contract components transferring risk to a single party.
Environment	Strong variation of impact across environment parameters. Duration and scope contracts are more volatile. Preferred types are more robust.	More incentivised dimensions improves robustness. Preferred multidimensional types are more robust. Including certain components increases sensitivity.

Where the goal of chapter 3 was to provide general guidance regarding the design of contracts, The aim of the 4-th chapter was to create a dedicated optimization method which is capable of constructing robust contracts for a given project. This research question also takes the the project owner's perspective, but moves towards a more operational application area. This is done by designing optimization techniques that provide answers to specific cases, rather than providing broad managerial guidelines. Specifically, a parallel multi-objective scatter search meta-heuristic has been developed to optimise the contract structure.

This multi-objective scatter search uses strongly problem-specific elements to enforce the diversity of the population, in order to evolve towards a better set of contracts for

the project owner to take into consideration. The actual selection of a specific contract of course being dependent on the specific risk-preferences of the project owner. Again using extensive computational experiments, the added value of this optimisation approach when compared to the brute-force approach taken in chapter 3, as well as the results obtained when using commercial solver software is proven.

Next, the perspective of the contractor was analysed in greater detail (chapter 5). The goal of this chapter was to analyse how the scheduling of a project subjected to a given incentive contract can be optimized. To this end, the traditional multi-mode scheduling approach was extended to include the four-dimensional trade-off modelled in chapter 3.

Moreover, this meant moving from a trade-off on the level of the complete project to a trade-off on the level of specific activities, since this is the level where the key decisions of the contractor are situated. The scheduling problem itself was then modelled using mixed integer programming, and implemented using Gurobi's optimisation framework. Computational experiments prove that the proposed formulation compares favourably to heuristic optimisation models, even for larger project instances.

Another key task of the contractor that succeeds the scheduling phase is monitoring the manner in which the project is being executed. The key goal of chapter 6 was to adjust existing project control techniques in order to enhance the relevance of the signals they produce for an incentivized project context. This is especially important since the profit margin of the contractor, and hereby also the objective of the contractor, are inextricably linked to the incentive structure of the projects.

Chapter 6 introduced the concept of 'earned incentive management', which is a technique based on the traditional earned value and earned schedule methodology. The key difference being that this novel methodology actively monitors the accrual of the incentive amounts rather than the incurred cost. By using this methodology, two key issues can be avoided. Firstly, the separate impact of the cost and duration dimensions can be correctly weighted, eliminating the need for dimension-specific metrics. Secondly, the EIM methodology is capable of monitoring non-increasing and irregular accrual of incentives throughout the project, resulting in more effective project control.

The advantages of the proposed control technique have been tested using a computational experiment which simulated the control performance on 4,200 projects with varying size and contract structures. The results of these experiments showed that the EIM technique significantly outperforms conventional EVM/ES methods, and should therefore be preferred in projects where cost and time incentives play a significant role.

10.1.2 Part II: Schedule Optimisation for Weather-Sensitive Projects

The second part of this dissertation investigated the usefulness of novel weather simulation techniques for managing weather-sensitive offshore projects. The motivation for this research was provided by the substantial advances in weather simulation techniques as well as the high capital intensity of such projects, making their optimisation highly relevant from an economical perspective.

Chapter 7 presented a novel weather simulation model, specifically tailored to offshore projects sensitive to both wind and wave conditions. The novelty of this model lies in it being designed for operational rather than meteorological objectives. As such, the model is capable of simulating weather using operationally relevant time intervals, and for the complete duration of long term projects.

The capabilities of this model were leveraged in chapter 8, where the scheduling of offshore construction projects was investigated. Specifically, the construction of offshore wind turbines has been investigated. The scheduling of projects was optimised by designing a number of novel chromosome encodings that represent different levels of abstraction in the scheduling of offshore construction projects. These encodings were optimised using meta-heuristic as well as dedicated solution techniques. Computational experiments indicate that the best results are obtained when using dedicated heuristics on a systematically reduced solution space.

Next, chapter 9 extended pre-existing literature on the maintenance of offshore wind turbines by considering the operational go/no-go decision regarding preventive maintenance. The key consideration here being that preventive maintenance is best executed at times when wind speeds, and therefore energy production, are low. Again leveraging the capabilities of the weather simulation model presented in section 7, several strategies for taking this decision have been compared in a computational experiment. Specifically the performance of a static threshold, a workload dependent threshold, support vector machines and an artificial neural networks were compared. The results indicate that using a workload dependent threshold to take this decision can significantly improve the quality of decision making, reducing the gap with the lower bound.

10.2 Future Research Avenues

10.2.1 Part I: A Quantitative Analysis of Incentive Contracts for Projects

There still remain a substantial amount of challenges regarding the design and implications of incentive contracts for projects. The following research topics could provide interesting

insights and extend the current state of the art.

10.2.1.1 Data Collection

Current literature includes a large number of cases but there could be substantial value in the creation of a larger database of incentive contracts as well as the nature of the projects to which they have been applied. As of now, the available information is either solely on the type of contract employed, or on the nature of the project. Combining this information could provide valuable insights in the actual impact these contracts have on project performance.

10.2.1.2 Stochastic Project Models

Due to the inherent complexity of the subject matter, the research in this dissertation has focussed on deterministic problem formulations. Nevertheless, project environments are frequently subjected to substantial amounts of stochasticity. Hence, there could be substantial academic and practical value to an analysis which is capable of realistically modelling the uncertainty associated with this four-dimensional trade-off and the implications of this uncertainty for the performance of incentive contracts.

10.2.1.3 Resource Constrained Scheduling

The scheduling methodology proposed in this research could be extended by including resource constraints on top of the four-dimensional trade-offs. By doing so it is unlikely that the problem could still be solved to optimality. Nevertheless, the resource constrained project scheduling problem is a classic staple of operational research, with a high degree of relevance in practice. As such, this is certain to be a valuable extension of this research.

10.2.2 Part II: Schedule Optimisation for Weather-Sensitive Projects

10.2.2.1 Extending to Other Types of Weather Dependencies

The chapters in the second part of this dissertation have added to the literature combining weather modelling with operational decision making. However, the cross section between these two research domains still contains a lot of virgin territory. One possible avenue for future research could be the exploration of projects that are susceptible to different types of weather conditions such as precipitation or temperature.

10.2.2.2 Datasets of Benchmark Problems

Another possible extension of this research is the creation of a dataset of benchmark problems of varying type and structure. These benchmark problems could provide a more in-depth knowledge of the performance of various abstraction levels used when optimising the scheduling for weather-sensitive projects.

10.2.2.3 Case Studies in Other Industries

The research in the paper has been applied to offshore wind turbines in particular. Extending this type of research to other industries could uncover specific issues associated with various industries. In the long term such a body of research could result in a better knowledge of the most adequate techniques to address specific operational challenges. In extremis such research could lead to a formal classification system for these problems, as is already in place for various problems in operational research, such as machine scheduling.

10.3 Acknowledgements

We acknowledge the support provided by the “Nationale Bank van België” (NBB) and by the “Bijzonder Onderzoeksfonds” (BOF) for the project with contract number BOF12GOA021. This work was carried out using 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.

References

Abu-Hijleh, S., Ibbs, C., 1989. Schedule-based construction incentives. Journal of Construction Engineering and Management 115 (3), 430–443.

Acebes, F., Pereda, M., Poza, D., Pajares, J., Galán, J. M., 2015. Stochastic earned value analysis using monte carlo simulation and statistical learning techniques. International Journal of Project Management 33 (7), 1597–1609.

Achuthan, N., Hardjawidjaja, A., 2001. Project scheduling under time dependent costs–a branch and bound algorithm. Annals of Operations Research 108 (1-4), 55–74.

Afshar, A., Kaveh, A., Shoghli, O., 2007. Multi-Objective optimization of time-cost-quality using multi-colony ant algorithm. Asian Journal of civil engineering (Building and Housing) 8 (2), 113–124.

Ailliot, P., Monbet, V., Prevosto, M., 2006. An autoregressive model with time-varying coefficients for wind fields. Environmetrics 17 (September 2005), 107–117.
URL http://onlinelibrary.wiley.com/doi/10.1002/env.753/abstract

Al-Subhi Al-Harbi, K. M., 1998. Sharing fractions in cost-plus-incentive-fee contracts. International Journal of project management 16 (2), 73–80.

Alba, E., Luque, G., Nesmachnow, S., 2013. Parallel metaheuristics: recent advances and new trends. International Transactions in Operational Research 20 (1), 1–48.

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

Anvuur, A. M., Kumaraswamy, M. K., 2010. Promises, pitfalls and shortfalls of the guaranteed maximum price approach: a comparative case study. In: Association of Researchers in Construction Management, ARCOM 2010-Proceedings of the 26th Annual Conference. pp. 1079–1088.

Arditi, D., Yasamis, F., 1998. Incentive/disincentive contracts: perceptions of owners and contractors. Journal of Construction Engineering and Management 124 (5), 361–373.

Arena, F., Puca, S., 2004. The Reconstruction of Significant Wave Height Time Series by Using a Neural Network Approach. Journal of Offshore Mechanics and Arctic Engineering 126 (3), 213.
URL http://offshoremechanics.asmedigitalcollection.asme.org/article.aspx?articleid=1455334

Bajari, P., Lewis, G., 2009. Procurement contracting with time incentives: Theory and evidence. Tech. rep., National Bureau of Economic Research.

Bajestani, M. A., Rabbani, M., Rahimi-Vahed, A., Khoshkhou, G. B., 2009. A multi-objective scatter search for a dynamic cell formation problem. Computers & operations research 36 (3), 777–794.

Ballestín, F., 2007. A genetic algorithm for the resource renting problem with minimum and maximum time lags. Lecture Notes in Computer Science 4446, 25–35.

Ballestín, F., 2008. Different codifications and metaheuristic algorithms for the resource renting problem with minimum and maximum time lags. In: Cotta, C., van Hemert, J. (Eds.), Recent Advances in Evolutionary Computation for Combinatorial Intelligence. Vol. 153 of Studies in Computational Intelligence. Springer-Verlag Berlin Heidelberg, Ch. 12, pp. 187–202.

Batselier, J., Vanhoucke, M., 2015. Evaluation of deterministic state-of-the-art forecasting approaches for project duration based on earned value management. International Journal of Project Management 33 (7), 1588–1596.

Bayiz, M., Corbett, C., 2005. Coordination and incentive contracts in project management under asymmetric information. Available at SSRN 914227.

Beausoleil, R. P., 2006. Moss multiobjective scatter search applied to non-linear multiple criteria optimization. European Journal of Operational Research 169 (2), 426–449.

Bendavid, I., Golany, B., Aug. 2011. Setting gates for activities in the stochastic project scheduling problem through the cross entropy methodology. Annals of Operations Research 189 (1), 25–42.
URL http://link.springer.com/10.1007/s10479-011-0940-1

Bendoly, E., Swink, M., 2007. Moderating effects of information access on project management behavior, performance and perceptions. Journal of Operations Management 25 (3), 604–622.

Berends, K., 2000. Cost plus incentive fee contracting - experiences and structuring. International Journal of Project Management 18, 165–171.

Boukendour, S., Bah, R., Oct. 2001. The guaranteed maximum price contract as call option. Construction Management and Economics 19 (6), 563–567.

Bower, D., Ashby, G., Gerald, K., Smyk, W., 2002. Incentive mechanisms for project success. Journal of Management in Engineering 18 (1), 37–43.

Bowers, J., Mould, G., 1994. Weather risk in offshore projects. Journal of the Operational Research Society 45 (4), 409–418.
URL http://www.jstor.org/stable/2584212

Bożejko, W., Wodecki, M., 2008. Parallel scatter search algorithm for the flow shop sequencing problem. In: Parallel Processing and Applied Mathematics. Springer, pp. 180–188.

Bresnen, M., Marshall, N., 2000. Partnering in construction: a critical review of issues, problems and dilemmas. Construction Management and Economics 18 (2), 229–237.

Brokish, K., Kirtley, J., 2009. Pitfalls of modeling wind power using markov chains. In: Power Systems Conference and Exposition, 2009. PSCE'09. IEEE/PES. IEEE, pp. 1–6.

Broome, J., Perry, J., Jan. 2002. How practitioners set share fractions in target cost contracts. International Journal of Project Management 20 (1), 59–66.

Browning, T. R., 2010. On the alignment of the purposes and views of process models in project management. Journal of Operations Management 28 (4), 316–332.

Browning, T. R., 2014. A quantitative framework for managing project value, risk, and opportunity. IEEE Transactions on Engineering Management 61 (4), 583–598.

Bubshait, A. a., Jan. 2003. Incentive/disincentive contracts and its effects on industrial projects. International Journal of Project Management 21 (1), 63–70.

Burges, C. J., 1998. A tutorial on support vector machines for pattern recognition. Data mining and knowledge discovery 2 (2), 121–167.

Burke, E. K., Kendall, G., 2005. Search methodologies. Springer.

Chan, D., Lam, P., Chan, J., 2012. A Comparative Study of the Benefits of Applying Target Cost Contracts Between South Australia and Hong Kong. Project Management Journal 43 (2), 4–20.

Chan, D. W., Chan, A. P., Lam, P. T., Wong, J. M., 2010a. Identifying the critical success factors for target cost contracts in the construction industry. Journal of Facilities Management 8 (3), 179–201.

Chan, D. W., Chan, A. P., Lam, P. T., Wong, J. M., Jul. 2011a. An empirical survey of the motives and benefits of adopting guaranteed maximum price and target cost contracts in construction. International Journal of Project Management 29 (5), 577–590.

Chan, D. W., Lam, P. T., Chan, A. P., Wong, J. M., 2010b. Achieving better performance through target cost contracts: The tale of an underground railway station modification project. Facilities 28 (5/6), 261–277.

Chan, J. H., Chan, D. W., Lam, P. T., Chan, A. P., 2011b. Preferred risk allocation in target cost contracts in construction. Facilities 29 (13/14), 542–562.

Chapman, C., Ward, S., Nov. 1994. The efficient allocation of risk in contracts. Omega 22 (6), 537–552.

Chapman, C., Ward, S., Jun. 2008. Developing and implementing a balanced incentive and risk sharing contract. Construction Management and Economics 26 (6), 659–669.

Chen, H. L., Chen, W. T., Lin, Y. L., 2016. Earned value project management: Improving the predictive power of planned value. International Journal of Project Management 34 (1), 22–29.

Chen, T., Klastorin, T., Wagner, M. R., 2015. Incentive contracts in serial stochastic projects. Manufacturing & Service Operations Management.

Cheng, T., Ding, Q., Lin, B., Jan. 2004. A concise survey of scheduling with time-dependent processing times. European Journal of Operational Research 152 (1), 1–13. URL http://linkinghub.elsevier.com/retrieve/pii/S0377221702009098

Choi, K., Kwak, Y. H., Jan. 2012. Decision support model for incentives/disincentives time cost tradeoff. Automation in Construction 21, 219–228.

Choi, K., Kwak, Y. H., Yu, B., 2010. Quantitative model for determining incentive/disincentive amounts through schedule simulations. Simulation Conference (WSC), Proceedings of the 2010 Winter, 3295–3306.

Cohen, I., Iluz, M., 2014. When cost–effective design strategies are not enough: Evidence from an experimental study on the role of redundant goals. Omega.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine learning 20 (3), 273–297.

Cukierman, A., Shiffer, Z., 1976. Contracting for optimal delivery time in long-term projects. The Bell Journal of Economics 7 (1), 132–149.

Cunha, C., Guedes Soares, C., Jun. 1999. On the choice of data transformation for modelling time series of significant wave height. Ocean Engineering 26 (6), 489–506. URL http://linkinghub.elsevier.com/retrieve/pii/S0029801898000146

Davis, E., 1975. Project network summary measures constrained-resource scheduling. AIIE Transactions 7, 132–142.

Dayanand, N., Padman, R., 2001. Project contracts and payment schedules: the client's problem. Management Science 47, 1654–1667.

De, P., Dunne, E., Ghosh, J., Wells, C., 1995. The discrete time-cost tradeoff problem revisited. European Journal of Operational Research 81, 225–238.

De Boer, P.-T., Kroese, D. P., Mannor, S., Rubinstein, R. Y., 2005. A tutorial on the cross-entropy method. Annals of operations research 134 (1), 19–67.

De Wit, A., 1988. Measurement of project success. International journal of project management 6 (3), 164–170.

Deb, K., 2001. Multi-objective optimization using evolutionary algorithms. Vol. 16. John Wiley & Sons.

Ebrahim, R. M., Razmi, J., Haleh, H., 2009. Scatter search algorithm for supplier selection and order lot sizing under multiple price discount environment. Advances in Engineering Software 40 (9), 766–776.

Ederer, F., Manso, G., 2013. Is pay for performance detrimental to innovation? Management Science 59 (7), 1496–1513.

El-Rayes, K., 2001a. Optimim planning of highway construction under A + B bidding method. Journal of Construction Engineering and Management 127 (4), 261–269.

El-Rayes, K., 2001b. Optimum planning of highway construction under A+B bidding method. Journal of Construction Engineering and Management 127, 261–269.

El-Rayes, K., Kandil, A., Apr. 2005. Time-Cost-Quality Trade-Off Analysis for Highway Construction. Journal of Construction Engineering and Management 131 (4), 477–486.

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.

Etgar, R., Shtub, A., LeBlanc, L., 1996. Scheduling projects to maximize net present value-the case of time-dependent, contingent cash flows. European Journal of Operational Research 96, 90–96.

Ettoumi, F., Sauvageot, H., a. E.-H Adane, Sep. 2003. Statistical bivariate modelling of wind using first-order Markov chain and Weibull distribution. Renewable Energy 28 (11), 1787–1802.
URL http://linkinghub.elsevier.com/retrieve/pii/S0960148103000193

European Wind Energy Association, 2011. Wind in our sails-the coming of europe's offshore wind energy industry. Tech. rep.

Fahmy, A., Hassan, T., Bassioni, H., 2014. What is Dynamic Scheduling? PM World Journal III (V), 1–9.
URL http://pmworldjournal.net/wp-content/uploads/2014/05/pmwj22-may2014-Fahmy-Hassan-Bassioni-What-is-dynamic-scheduling-FeaturedPaper.pdf

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

Gangwar, M., Goodrum, P. M., 2005. The effect of time on safety incentive programs in the us construction industry. Construction Management and Economics 23 (8), 851–859.

Garcıa-López, F., Melián-Batista, B., Moreno-Pérez, J. A., Moreno-Vega, J. M., 2003. Parallelization of the scatter search for the p-median problem. Parallel Computing 29 (5), 575–589.

Ghodsi, R., Skandari, M. R., Allahverdiloo, M., Iranmanesh, S. H., 2009. A new practical model to trade-off time, cost, and quality of a project. Australian Journal of Basic and Applied Sciences 3 (4), 3741–3756.

Gibbons, R., 2005. Incentives between firms (and within). Management Science 51 (1), 2–17.

Glover, F., Laguna, M., Marti, R., 2000. Fundamentals of scatter search and path relinking. Control and Cybernetics 29, 653–684.

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

Graham, C., 1982. The Parameterisation and Prediction of wave height and wind speed persistence statistics for oil industry operational planning purposes. Coastal Engineering 6 (4), 303—-329.
URL http://www.sciencedirect.com/science/article/pii/0378383982900059

Gundegjerde, C., Halvorsen, I. B., 2012. Vessel fleet size and mix for maintenance of offshore wind farms: A stochastic approach.

Gupta, D., Snir, E. M., Chen, Y., 2015. Contractors,Äô and agency decisions and policy implications in a+ b bidding. Production and Operations Management 24 (1), 159–177.

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.

Hastie, T., Tibshirani, R., Friedman, J., Franklin, J., 2005. The elements of statistical learning: data mining, inference and prediction. The Mathematical Intelligencer 27 (2), 83–85.

Herbsman, Z., 1995. A + B Bidding Method - Hidden success story for Highway Construction. Journal of Construction Engineering and Management 4.

Herroelen, W., De Reyck, B., 1999. Phase transitions in project scheduling. Journal of the Operational Research Society 50, 148–156.

Herroelen, W., Leus, R., 2004. The construction of stable project baseline schedules. European Journal of Operational Research 156 (3), 550–565.

Herroelen, W., Leus, R., 2005. Project scheduling under uncertainty: Survey and research potentials. European Journal of Operational Research 165, 289–306.

Herroelen, W., Van Dommelen, P., Demeulemeester, E., 1997. Project network models with discounted cash flows a guided tour through recent developments. European Journal of Operational Research 100, 97–121.

Herten, H., Peeters, W., 1986. Incentive contracting as a project management tool. International Journal of Project Management 4 (1), 34–39.

Hiller, J., Tollison, R., 1978. Incentive versus cost-plus contracts in defense procurement. The Journal of Industrial Economics 26 (3), 239–248.

Hu, W., He, X., 2014. An innovative time-cost-quality tradeoff modeling of building construction project based on resource allocation. The Scientific World Journal 2014.

Huang, W., Ding, L., May 2011. Project-Scheduling Problem With Random Time-Dependent Activity Duration Times. IEEE Transactions on Engineering Management 58 (2), 377–387.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5585739

Iranmanesh, H., Skandari, M., Allahverdiloo, M., 2008. Finding pareto optimal front for the multi-mode time, cost quality trade-off in project scheduling. International Journal of Computer, Information, and Systems Science, and Engineering 2 (2), 118–118.

Jaafari, A., 1996. Twinning time and cost in incentive-based contracts. Journal of Management in Engineering.

Jaraiedi, M., Plummer, R. W., Aber, M. S., Mar. 1995. Incentive/Disincentive Guidelines for Highway Construction Contracts. Journal of Construction Engineering and Management 121 (1), 112–120.

Kandil, A., El-Rayes, K., May 2006. Parallel Genetic Algorithms for Optimizing Resource Utilization in Large-Scale Construction Projects. Journal of Construction Engineering and Management 132 (5), 491–498.

Karyotakis, A., 2011. On the optimisation of operation and maintenance strategies for offshore wind farms. Ph.D. thesis, UCL (University College London).

Ke, H., Liu, B., Sep. 2005. Project scheduling problem with stochastic activity duration times. Applied Mathematics and Computation 168 (1), 342–353.
URL http://linkinghub.elsevier.com/retrieve/pii/S0096300304006083

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

Keren, B., Cohen, Y., 2012. Optimising project performance: the triangular trade off optimisation approach. International Journal of Engineering Management and Economics 3, 152–170.

Kerkhove, L.-P., Vanhoucke, M., 2015a. Incentive contract design for projects: The owner's perspective. Omega Article in Press.

Kerkhove, L.-P., Vanhoucke, M., 2015b. A parallel multi-objective scatter search for optimising incentive contract design in projects. Working Paper at Ghent University, currently under journal submission (Check www.projectmanagement.ugent.be for an update).

Kerkhove, L.-P., Vanhoucke, M., Maenhout, B., Vandenheede, L., 2015. On the resource renting problem with overtime. Working Paper at Ghent University, currently under journal submission (Check www.projectmanagement.ugent.be for an update).

Kerr, S., 1975. On the folly of rewarding a, while hoping for b. Academy of Management journal 18 (4), 769–783.

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.

Khang, D. B., Myint, Y. M., 1999. Time, cost and quality trade-off in project management: a case study. International journal of project management 17 (4), 249–256.

Lee, E., Thomas, D., 2007. State-of-practice technologies on accelerated urban highway rehabilitation: I-15 California experience. Journal of Construction Engineering and Management.

Lévárdy, V., Browning, T. R., 2009. An adaptive process model to support product development project management. IEEE Transactions on Engineering Management 56 (4), 600–620.

Lipke, W., 2003. Schedule is different. The Measurable News Summer, 31–34.

Lippman, S. a., McCardle, K. F., Tang, C. S., Sep. 2013. Using Nash bargaining to design project management contracts under cost uncertainty. International Journal of Production Economics 145 (1), 199–207.

López, F. G., Torres, M. G., Batista, B. M., Pérez, J. A. M., Moreno-Vega, J. M., 2006. Solving feature subset selection problem by a parallel scatter search. European Journal of Operational Research 169 (2), 477–489.

Love, P. E., Davis, P. R., Chevis, R., Edwards, D. J., 2010. Risk/reward compensation model for civil engineering infrastructure alliance projects. Journal of Construction Engineering and Management 137 (2), 127–136.

MacCormack, A., Mishra, A., 2015. Managing the performance trade-offs from partner integration: Implications of contract choice in r&d projects. Production and Operations Management 24 (10), 1552–1569.

Mackley, A. R., 2012. Use of incentive/disincentive contracting to mitigate work zone traffic impacts. Ph.D. thesis, University of Missouri–Columbia.

Makarynskyy, O., a.a. Pires-Silva, Makarynska, D., Ventura-Soares, C., May 2005. Artificial neural networks in wave predictions at the west coast of Portugal. Computers & Geosciences 31 (4), 415–424.
URL http://linkinghub.elsevier.com/retrieve/pii/S0098300404002158

Mandal, S., Prabaharan, N., Jan. 2010. Ocean Wave Prediction Using Numerical and Neural Network Models. The Open Ocean Engineering Journal 3 (1), 12–17.
URL http://benthamopen.com/tooej/articles/V003/12TOOEJ.htm

Mangasarian, O. L., 2003. Data mining via support vector machines. In: System Modeling and Optimization XX. Springer, pp. 91–112.

Maples, B., Saur, G., Hand, M., van de Pietermen, R., Obdam, T., 2013. Installation, operation, and maintenance strategies to reduce the cost of offshore wind energy. NREL, Denver.

Markowitz, H., 1952. Portfolio selection*. The journal of finance 7 (1), 77–91.

Marques, G., Gourc, D., Lauras, M., 2011. Multi-criteria performance analysis for decision making in project management. International Journal of Project Management 29 (8), 1057–1069.

Mastor, A., 1970. An experimental and comparative evaluation of production line balancing techniques. Management Science 16, 728–746.

McAfee, R., McMillan, J., 1986. Bidding for contracts: a principal-agent analysis. The RAND Journal of Economics 17 (3), 326–338.

McCall, J., 1970. The simple economics of incentive contracting. The American Economic Review 1, 28–29.

McCulloch, W. S., Pitts, W., 1943. A logical calculus of the ideas immanent in nervous activity. The bulletin of mathematical biophysics 5 (4), 115–133.

Mehta, S., Uzsoy, R., 1998. Predictable scheduling of a job shop subject to breakdowns. Robotics and Automation, IEEE Transactions on 14 (3), 365–378.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=678447

Meier, C., Yassine, A. A., Browning, T. R., Walter, U., 2016. Optimizing time–cost tradeoffs in product development projects with a multi-objective evolutionary algorithm. Research in Engineering Design, 1–20.

Meinhart, W., Delionback, L., 1968. Project Management: An Incentive Contracting Decision Model. Academy of Management Journal, 427–435.

Meng, X., Gallagher, B., Apr. 2012. The impact of incentive mechanisms on project performance. International Journal of Project Management 30 (3), 352–362.

Messac, A., 2015. Optimization in practice with MATLAB for engineering students and professionals. Cambridge University Press, New York, NY.

Mihm, J., 2010. Incentives in new product development projects and the role of target costing. Management Science 56 (8), 1324–1344.

Möhring, R., Schulz, A., 2003. Solving project scheduling problems by minimum cut computations. Management Science 49 (3), 330–350.
URL http://pubsonline.informs.org/doi/abs/10.1287/mnsc.49.3.330.12737

Möhring, R., Schulz, A., Stork, F., Uetz, M., 2001. On project scheduling with irregular starting time costs. Operations Research Letters 28 (664), 149—-154.
URL http://www.sciencedirect.com/science/article/pii/S0167637701000645

Monbet, V., Ailliot, P., Prevosto, M., Apr. 2007. Survey of stochastic models for wind and sea state time series. Probabilistic Engineering Mechanics 22 (2), 113–126.
URL http://linkinghub.elsevier.com/retrieve/pii/S0266892006000361

More, A., Deo, M., Jan. 2003. Forecasting wind with neural networks. Marine Structures 16 (1), 35–49.
URL http://linkinghub.elsevier.com/retrieve/pii/S0951833902000539

Müller, R., Turner, J. R., 2005. The impact of principal–agent relationship and contract type on communication between project owner and manager. International Journal of Project Management 23 (5), 398–403.

Naeni, L. M., Shadrokh, S., Salehipour, A., 2014. A fuzzy approach for the earned value management. International Journal of Project Management 32 (4), 709–716.

Nebro, A. J., Luna, F., Alba, E., Dorronsoro, B., Durillo, J. J., Beham, A., 2008. Abyss: Adapting scatter search to multiobjective optimization. Evolutionary Computation, IEEE Transactions on 12 (4), 439–457.

Norlund, E. K., Gribkovskaia, I., Laporte, G., 2015. Supply vessel planning under cost, environment and robustness considerations. Omega 57, 271–281.

Nübel, H., 2001. The resource renting problem subject to temporal constraints. OR Spektrum 23, 359–381.

Okabe, T., Jin, Y., Sendhoff, B., 2003. A critical survey of performance indices for multi-objective optimisation. In: Evolutionary Computation, 2003. CEC'03. The 2003 Congress on. Vol. 2. IEEE, pp. 878–885.

Ortegon, K., Nies, L. F., Sutherland, J. W., 2013. Preparing for end of service life of wind turbines. Journal of Cleaner Production 39, 191–199.

Osei-Bryson, K.-M., Ngwenyama, O. K., 2006. Managing risks in information systems outsourcing: An approach to analyzing outsourcing risks and structuring incentive contracts. European Journal of Operational Research 174 (1), 245–264.

Paquin, J., Couillard, J., Ferrand, D., 2000. Assessing and controlling the quality of a project end product: the earned quality method. IEEE Transactions on Engineering Management 47 (1), 88–97.

Pascoe, T., 1966. Allocation of resources - CPM. Revue Française de Recherche Opérationnelle 38, 31–38.

Perry, J. G., Barnes, M., 2000. Target cost contracts: an analysis of the interplay between fee, target, share and price. Engineering, Construction and Architectural Management 7 (2), 202–208.

Picton, P., 2000. Neural Networks. Grassroots (Lagos, Nigeria). Palgrave.
URL `https://books.google.be/books?id=QWakQgAACAAJ`

Pollack-johnson, B., Liberatore, M. J., 2006. Incorporating quality considerations into project time/cost tradeoff analysis and decision making. IEEE Transactions on Engineering Management 53 (4), 534–542.

Pour, N., Modarres, M., Aryanejad, M., Moghadam, R., 2010. The discrete time-cost-quality trade-off problem using a novel hybrid genetic algorithm. Applied Mathematical Sciences 42, 2081–2094.

Qian, F., Gribkovskaia, I., Laporte, G., Halskau sr, Ø., 2012. Passenger and pilot risk minimization in offshore helicopter transportation. Omega 40 (5), 584–593.

Qian, F., Strusevich, V., Gribkovskaia, I., Halskau, Ø., 2015. Minimization of passenger takeoff and landing risk in offshore helicopter transportation: Models, approaches and analysis. Omega 51, 93–106.

Rahimi, M., Iranmanesh, H., 2008. Multi Objective Particle Swarm Optimization for a Discrete Time, Cost and Quality Trade-off Problem. World Applied Sciences Journal 4, 270–276.

Rahimi-Vahed, A., Rabbani, M., Tavakkoli-Moghaddam, R., Torabi, S. A., Jolai, F., 2007. A multi-objective scatter search for a mixed-model assembly line sequencing problem. Advanced Engineering Informatics 21 (1), 85–99.

Ramón, J., Cristóbal, J. S., 2009. Time, cost, and quality in a road building project. Journal of Construction Engineering and Management 135 (11), 1271–1275.

Ranjbar, M., De Reyck, B., Kianfar, F., 2009. A hybrid scatter-search for the discrete time/resource trade-off problem in project scheduling. European Journal of Operational Research 193, 35–48.

Regnier, E., Feb. 2008. Doing something about the weather. Omega 36 (1), 22–32. URL http://linkinghub.elsevier.com/retrieve/pii/S0305048305001805

Rosandich, R. G., 2007. Choosing the best contract type for a given situation: An agent-based simulation study. Tech. rep., Working Paper, Department of Mechanical and Industrial Engineering, University of Minnesota Duluth.

Rose, T., Manley, K., 2010. Financial incentives and advanced construction procurement systems. Project Management Journal 41, 40–50.

Rose, T., Manley, K., Jul. 2011. Motivation toward financial incentive goals on construction projects. Journal of Business Research 64 (7), 765–773.

Rose, T. M., 2008. The impact of financial incentive mechanisms on motivation in australian government large non-residential building projects. Ph.D. thesis, Queensland University of Technology.

Rosenfeld, Y., Geltner, D., 1991. Cost-Plus and incentive contracting: some false benefits and inherent drawbacks. Construction Management and Economics 9, 481–492.

Rothkopf, M., McCarron, J., Fromovitz, S., 1974. A Weather Model for Simulating Offshore Construction Alternatives. Management Science 20 (10), 1345–1349. URL http://pubsonline.informs.org/doi/abs/10.1287/mnsc.20.10.1345

Ryan, P. J., Henin, C. G., Gandhi, D. K., Jan. 1986. Financial incentives for cost control under moral hazard. Omega 14 (3), 221–231.

Sacks, R., Harel, M., 2006. An economic game theory model of subcontractor resource allocation behaviour. Construction Management and Economics 24 (8), 869–881.

Sahin, A. D., Sen, Z., Mar. 2001. First-order Markov chain approach to wind speed modelling. Journal of Wind Engineering and Industrial Aerodynamics 89 (3-4), 263–269. URL http://linkinghub.elsevier.com/retrieve/pii/S0167610500000817

Sarker, R., Coello, C. A. C., 2002. Assessment methodologies for multiobjective evolutionary algorithms. In: Evolutionary Optimization. Springer, pp. 177–195.

Schönefeld, P., 2014. private communication.

Schutze, O., Esquivel, X., Lara, A., Coello Coello, C. A., 2012. Using the averaged hausdorff distance as a performance measure in evolutionary multiobjective optimization. Evolutionary Computation, IEEE Transactions on 16 (4), 504–522.

Scotto, M., Guedes Soares, C., Jul. 2000. Modelling the long-term time series of significant wave height with non-linear threshold models. Coastal Engineering 40 (4), 313–327.
URL http://linkinghub.elsevier.com/retrieve/pii/S0378383900000168

Shafiee, M., 2015. Maintenance logistics organization for offshore wind energy: current progress and future perspectives. Renewable Energy 77, 182–193.

Shahsavari Pour, N., Modarres, M., Tavakkoli-Moghaddam, R., Najafi, E., 2010. Optimizing a multi-objectives time-cost-quality trade-off problem by a new hybrid genetic algorithm. World Applied Sciences Journal 10 (3), 355–363.

Shamshad, a., Bawadi, M., Wanhussin, W., Majid, T., Sanusi, S., Apr. 2005. First and second order Markov chain models for synthetic generation of wind speed time series. Energy 30 (5), 693–708.
URL http://linkinghub.elsevier.com/retrieve/pii/S0360544204002609

Shr, J., Chen, W., 2004. Setting maximum incentive for incentive/disincentive contracts for highway projects. Journal of Construction Engineering and Management 130 (1), 84–93.

Shr, J., Ran, B., Sung, C., 2004. Method to Determine Minimum Contract Bid for A+B+I/D Highway Projects. Journal of Construction Engineering and Management 130 (4), 509–516.

Shr, J. F., Chen, W. T., Nov. 2003. A method to determine minimum contract bids for incentive highway projects. International Journal of Project Management 21 (8), 601–615.

Sillars, D. N., 2007. Establishing guidelines for incentive/disincentive contracting at odot. Tech. rep., Oregon Department of Transportation, Research Unit.

Smola, A. J., Schölkopf, B., 2004. A tutorial on support vector regression. Statistics and computing 14 (3), 199–222.

Sobel, M. J., Szmerekovsky, J. G., Tilson, V., Nov. 2009. Scheduling projects with stochastic activity duration to maximize expected net present value. European Journal of Op-

erational Research 198 (3), 697–705.
URL http://linkinghub.elsevier.com/retrieve/pii/S0377221708008291

Sommer, S. C., Loch, C. H., 2009. Incentive contracts in projects with unforeseeable uncertainty. Production and Operations Management 18 (2), 185–196.

Song, L., AbouRizk, S. M., 2005. Quantifying engineering project scope for productivity modeling. Journal of construction engineering and management 131 (3), 360–367.

Stefanakos, C., Athanassoulis, G., 2002. Multiscale time series modelling of significant wave height. In: Proceedings of The Twelfth (2002) International Offshore and Polar Engineering Conference. Vol. 3. pp. 66–73.
URL http://e-book.lib.sjtu.edu.cn/isope2002/pdffiles/Volume3/3011p066.pdf

Stefanakos, C. N., Belibassakis, K. A., 2005. Nonstationary stochastic modelling of multivariate long-term wind and wave data. In: ASME 2005 24th International Conference on Offshore Mechanics and Arctic Engineering. American Society of Mechanical Engineers, pp. 225–234.

Stenbeck, T., 2008. Quantifying effects of incentives in a rail maintenance performance-based contract. Journal of construction engineering and management 134 (4), 265–272.

Stork, F., 2000. Branch-and-bound algorithms for stochastic resource-constrained project scheduling. Technical report, 702–2000.

Stukhart, G., Mar. 1984. Contractual Incentives. Journal of Construction Engineering and Management 110 (1), 34–42.

Talbi, E.-G., 2009. Metaheuristics: from design to implementation.

Tang, C. S., Zhang, K., Zhou, S. X., 2015a. Incentive contracts for managing a project with uncertain completion time. Production and Operations Management 24 (12), 1945–1954.

Tang, J., Brouste, A., Tsui, K. L., 2015b. Some improvements of wind speed markov chain modeling. Renewable Energy 81, 52–56.

Tang, W., Duffield, C., Young, D., 2006. Partnering mechanism in construction: an empirical study on the Chinese construction industry. Journal of Construction Engineering and Management 132 (March), 217–230.

Tang, W., Qiang, M., Duffield, C. F., Young, D. M., Lu, Y., 2008. Incentives in the chinese construction industry. Journal of Construction Engineering and Management 134 (7), 457–467.

Tareghian, H., Taheri, S., 2006. On the discrete time, cost and quality trade-off problem. Applied Mathematics and Computation 181, 1305–1312.

Tavakkoli-Moghaddam, R., Makui, A., Mazloomi, Z., 2010. A new integrated mathematical model for a bi-objective multi-depot location-routing problem solved by a multi-objective scatter search algorithm. Journal of Manufacturing Systems 29 (2), 111–119.

Tavana, M., Abtahi, A.-R., Khalili-Damghani, K., Aug. 2013. A new multi objective multi mode model for solving preemptive time cost quality trade off project scheduling problems. Expert Systems with Applications 41 (4), 1830–1846.

Tavares, L., 1999. Advanced models for project management. Kluwer Academic Publishers, Dordrecht, 1999.

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

Tavares, L., Ferreira, J., Coelho, J., 2002. A comparative morphologic analysis of benchmark sets of project networks. International Journal of Project Management 20, 475–485.

Tavner, P., 2012. Offshore Wind Turbines: Reliability, Availability and Maintenance. Energy Engineering Series. Institution of Engineering and Technology.
URL https://books.google.be/books?id=6CXTSgISVmIC

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning 4, 2.

Trietsch, D., Feb. 2006. Optimal feeding buffers for projects or batch supply chains by an exact generalization of the newsvendor result. International Journal of Production Research 44 (4), 627–637.
URL http://www.tandfonline.com/doi/abs/10.1080/00207540500371881

Tukey, J. W., 1949. Comparing individual means in the analysis of variance. Biometrics, 99–114.

Turner, J. R., Jan. 2004. Farsighted project contract management: incomplete in its entirety. Construction Management and Economics 22 (1), 75–83.

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

Van Bussel, G., Henderson, A., Morgan, C., Smith, B., Barthelmie, R., Argyriadis, K., Arena, A., Niklasson, G., Peltola, E., 2001. State of the art and technology trends for offshore wind energy: operation and maintenance issues. In: Offshore Wind Energy EWEA special topic conference.

Van Peteghem, V., Vanhoucke, M., 2014. An experimental investigation of metaheuristics for the multi-mode resource-constrained project scheduling problem on new dataset instances. European Journal of Operational Research 235 (1), 62–72.

Van Veldhuizen, D. A., 1999a. Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. Tech. rep., DTIC Document.

Van Veldhuizen, D. A., 1999b. Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. Tech. rep., DTIC Document.

Van Weele, A., van der Puil, J., 2013. International Contracting: Contract Management in Complex Construction Projects. World Scientific Publishing Company.

Vandenheede, L., Vanhoucke, M., Maenhout, B., 2015. A scatter search for the extended resource renting problem. International Journal of Production Research, 1–21.

Vanhoucke, M., 2006. Work continuity constraints in project scheduling. Journal of Construction Engineering and Management 132, 14–25.

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

Vanhoucke, M., 2010b. Measuring time: An earned value performance management study. The Measurable News 1, 10–14.

Vanhoucke, M., 2012a. Dynamic Scheduling: Integrating Schedule Risk Analysis with Earned Value Management. The Measurable News 2, 11–13.

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

Vanhoucke, M., 2013. Project Management using Dynamic Scheduling: Baseline scheduling, risk analysis and project control. The Measurable News 2, 45–50.

Vanhoucke, M., 2014. Integrated Project Management and Control: First comes the theory, then the practice. Springer.

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

Vanhoucke, M., Demeulemeester, E., Herroelen, W., 2001. Maximizing the net present value of a project with linear time-dependent cash flows. International Journal of Production Research 39, 3159–3181.

Veld, J. i. T., Peeters, W., Aug. 1989. Keeping large projects under control: the importance of contract type selection. International Journal of Project Management 7 (3), 155–162.

Walker, D. H., Hampson, K., Peters, R., 2002. Project alliancing vs project partnering: a case study of the australian national museum project. Supply Chain Management: An International Journal 7 (2), 83–91.

Wang, X., Zhu, Q., Cheng, T., 2015. Subcontracting price schemes for order acceptance and scheduling. Omega 54, 1–10.

Ward, S., Chapman, C., Feb. 1995. Evaluating fixed price incentive contracts. Omega 23 (1), 49–62.

Weitzman, M. L., 1980. Efficient incentive contracts. The Quarterly Journal of Economics, 719–730.

Wilks, D. S., Wilby, R. L., Jul. 1999. The weather generation game: a review of stochastic weather models. Progress in Physical Geography 23 (3), 329–357.
URL http://ppg.sagepub.com/cgi/doi/10.1177/030913339902300302

Willems, L. L., Vanhoucke, M., 2015. Classification of articles and journals on project control and earned value management. International Journal of Project Management 33 (7), 1610–1634.

William, I. C., Ashley, D., 1987. Impact of various construction contract clauses. Journal of Construction Engineering and Management 113 (3), 501–521.

Zhang, H., Xing, F., Dec. 2010. Fuzzy-multi-objective particle swarm optimization for time-cost-quality tradeoff in construction. Automation in Construction 19 (8), 1067–1075.

Zhang, J., Liu, G., Zhang, Q., Bai, Z., 2015. Coordinating a supply chain for deteriorating items with a revenue sharing and cooperative investment contract. Omega 56, 37–49.

Zhang, L., Wu, Q., Chen, C., Yue, Y., 2012. Solution for the Nonlinear Multi-Objective Model in Construction Projects Using Improved Particle Swarm Optimization. Research Journal of Applied Sciences, Engineering and Technology 4 (19), 3565–3573.

Zhou, A., Qu, B.-Y., Li, H., Zhao, S.-Z., Suganthan, P. N., Zhang, Q., 2011. Multiobjective evolutionary algorithms: A survey of the state of the art. Swarm and Evolutionary Computation 1 (1), 32–49.

Zitzler, E., Laumanns, M., Thiele, L., Zitzler, E., Zitzler, E., Thiele, L., Thiele, L., 2001. Spea2: Improving the strength pareto evolutionary algorithm.